

DEMAND FORECASTING WITH MACHINE LEARNING METHODS

Jose Martin Rožanec

Doctoral Dissertation
Jožef Stefan International Postgraduate School
Ljubljana, Slovenia

Supervisor: Prof. Dr. Dunja Mladenić, Jožef Stefan Institute, Ljubljana, Slovenia
Co-Supervisor: Dr. Blaž Fortuna, Qlector d.o.o., Ljubljana, Slovenia

Evaluation Board:

Prof. Dr. Gregor Papa, Chair, Jožef Stefan International Postgraduate School, Ljubljana, Slovenia

Prof. Dr. Zoran Bosnić, Member, University of Ljubljana, Faculty of Computer and Information Science, Ljubljana, Slovenia.

Prof. Dr. Dimitrios Kyritsis, Member, EPFL, Lausanne, Switzerland.

MEDNARODNA PODIPLOMSKA ŠOLA JOŽEFA STEFANA
JOŽEF STEFAN INTERNATIONAL POSTGRADUATE SCHOOL



Jose Martin Rožanec

DEMAND FORECASTING WITH MACHINE LEARNING METHODS

Doctoral Dissertation

NAPOVEDOVANJE POVPRAŠEVANJ Z METODAMI STRO- JNEGA UČENJA

Doktorska disertacija

Supervisor: Prof. Dr. Dunja Mladenić

Co-Supervisor: Dr. Blaž Fortuna

Ljubljana, Slovenia, June 2023

Družini, prijateljem, predhodnikom.

Acknowledgments

In the following lines, I try to express my gratitude to the many people who supported me on this journey. This work would not have been possible without the support of many people who helped me with my research, advised, prayed for, and encouraged me. My gratitude is, in many cases, beyond words.

First, I would like to thank my advisor Dunja Mladenić, for her patience, advice, and support. I remember the first days when I came to the Jožef Stefan Institute, the first presentations, and how, over the years, I grew in knowledge and many skills. She had always been of great support, enabled many growth opportunities, and provided feedback. Second, I would like to thank my co-advisor Blaž Fortuna. When developing my first machine learning models for demand forecasting, he observed the products we forecasted for had a sparse demand. That observation fueled my quest to find a solution to such a problem. I dug into the scientific literature reporting on approaches for similar demands. I learned about lumpy and intermittent demands from the scientific literature and what a frequent problem it was. There I found the research opportunities that lead to two relevant contributions of this thesis: a set of metrics to assess the quality of such models and a novel approach to forecast lumpy and intermittent demand. Third, I would like to thank Marko Grobelnik and Mitja Jermol for all the support, learning, and other opportunities they enabled and opened through these years.

Next, I would like to thank Prof. Dr. Dimitrios Kyritsis and Dr. Jinzhi Lu. Prof. Dr. Dimitrios Kyritsis helped me to dive deep into the world of ontologies and how these are useful in the context of digital twins, the manufacturing domain, and how they enable cognitive capabilities in digital twins. Dr. Jinzhi Lu helped me hone my writing skills, something I will always be grateful for.

While working on the Ph.D., I gained much research experience on two European H2020 projects: FACTLOG and STAR. I want to acknowledge some colleagues with whom we collaborated closely: Prof. Dr. George Arampatzis, Nikos Sarantinoudis, Kostas Kalaboukas, Prof. Dr. Christos Emmanouilidis, Dr. Ioannis Soldatos, Babis Ipektsidis, and so many others.

Furthermore, I am grateful to my colleagues and the many people who, through the years, in the Laboratory of Artificial Intelligence at the Jožef Stefan Institute helped me grow through conversations, feedback, and debates. I am also grateful for the many people they introduced me to. I must mention Klemen Kenda, Inna Novalija, Patrik Zajec, Erik Novak, Luka Bizjak, Miha Cimperman, Beno Šircelj, Alenka Guček, and Swati. I would also like to extend my gratitude to the colleagues from Qlector.

Education is a lifelong process. Achieving a Ph.D. would not be possible without much effort invested in prior education. Therefore I would like to remember the many people who invested in my education.

Finally, I would like to thank my family and friends. In particular, I would like to remember those who kindly supported me and have departed. Without the support of my family and friends, this milestone could not have been achieved.

Abstract

This thesis examines how machine learning can be applied in demand forecasting. In particular, it describes a novel approach toward lumpy and intermittent demand forecasting. It advocates using a two-fold model for forecasting lumpy (irregular demand occurrence, strong demand size variability) and intermittent (irregular demand occurrence, little demand size variability) demands when considering fixed forecasting horizons. The two-fold model comprises a classification model to predict demand occurrence and a regression model to predict the demand quantity. By dividing the forecasting problem into these two dimensions, insights into the reasons affecting the forecast quality are exposed by allowing us to understand to what extent the forecasting model fails to predict demand occurrence or demand quantity. Furthermore, a new set of metrics is proposed to assess the quality of the forecasts: ROC AUC to determine the quality of the classifier, variations of the MASE metric to measure the quality of the regression component, and the SPEC metric to determine the impact of the forecast on the stock costs.

While an accurate prediction of when the demand event occurs is of utmost importance, a prediction score alone does not provide an intuitive interpretation to the human. Furthermore, multiple models produce different predictive score distributions. Therefore, an additional model (a.k.a. calibrator) can be trained to map the original predictive scores to probabilities (in our case, the probability of demand occurrence). While much research has been devoted to measuring the quality of such calibrators, it is assumed that ground truth is required to measure and calibrate machine learning models. This thesis presents a novel approach to measuring the quality of the calibration, addressing the shortcomings of a widely adopted metric. The results show that the newly proposed metrics can estimate the calibrators' quality without using ground truth when no concept drift occurs. This approach could be helpful in at least two scenarios: (i) monitoring the performance of the calibrator over time without incurring labeling costs and (ii) training a calibrator without ground truth (human label on whether the event under consideration took place or not). While some experiments were performed on training a calibrator without ground truth, the first results were not promising, and more research must be invested in this direction.

Finally, the thesis highlights a use case of demand forecasting where explainable artificial intelligence was used to determine features that were relevant to a given forecast. Furthermore, such insights were enriched with expert knowledge from ontologies and insights obtained through text-mining techniques to build high-level explanations leveraging information retrieved from external sources.

Povzetek

V doktorski disertaciji smo preučili uporabo strojnega učenja za napoved povpraševanja. Predlagali smo novi pristop za napoved občasnega povpraševanja, ki ima redno manjše ali večje razlike v povpraševanih količinah. Pristop deli napoved občasnega povpraševanja v dva modela: klasifikacijski model za napoved samega dogodka povpraševanja in regresijski model za napoved povpraševane količine. Delitev napovedi povpraševanja na dva dela omogoča boljši vpogled v samo napoved in razumevanje vzrokov netočnosti: ali napoved ni točna zaradi slabe napovedi glede samega dogodka povpraševanja, ali napoved ni točna zaradi napovedi povpraševane količine. Za primerno ovrednotenje kakovosti napovedi predlagamo tri metrike: AUC ROC za ovrednotenje natančnosti klasifikacijskega modela, MASE za ovrednotenje natančnosti regresijskega modela ter SPEC za razumevanje vpliva napovedi na stroške zalog.

Napoved samega dogodka povpraševanja ne omogoča verjetnostne interpretacije in posledično nima intuitivne razlage za osebo, ki koristi napovedi. Treba je ustvariti dodaten model za preslikavo vrednosti originalnega modela strojnega učenja v verjetnostne napovedne rezultate. Mnoge raziskave so posvetile pozornost verjetnostni kalibraciji modelov strojnega učenja. V njih se predvideva, da obstajajo neki označeni podatki, na podlagi katerih lahko poteka kalibracija modela. V doktorski tezi predlagamo nove metrike za merjenje kakovosti kalibracijskih modelov. Predlagane metrike uspešno odpravijo nekatere pomanjkljivosti ene izmed pogosto uporabljenih metrik. Iz rezultatov je tudi razvidno, da predlagane metrike omogočajo ocenjevanje kakovosti kalibracijskega modela brez uporabe označenih podatkov, ko se distribucija vhodnih podatkov modela strojnega učenja ne spreminja. Predlagane metrike so lahko koristne za ocenjevanje (i) kakovosti kalibracijskega modela v času, ne da bi potrebovali označene podatke, ter (ii) treniranje kalibracijskega modela brez označenih podatkov (človeških oznak glede na dogodek, ki ga model napoveduje). Čeprav smo opravili nekaj eksperimentov v tej smeri, prvi rezultati niso kazali kompetitivnih rezultatov. Naše mnenje je, da je smer vredna raziskav, vendar je treba vložiti več časa vanjo.

Doktorsko disertacijo zaključimo z nekaj primeri uporabe razložljive umetne inteligence pri napovedi povpraševanja. Vpoglede razložljive umetne inteligence smo obogatili z ekspertnim znanjem iz ontologij in znanjem, pridobljenim preko besedilnega rudarjenja novic raznih informacijskih virov na internetu.

Contents

Abbreviations	xv
1 Introduction	1
1.1 General Introduction	1
1.2 Motivation	2
1.3 Demand Forecasting Approaches and Metrics	3
1.4 Probability Calibration of Machine Learning Models	5
1.5 Aims and Hypotheses	6
1.6 Scientific Contributions	7
1.7 Thesis Structure	8
2 Demand Forecasting	11
2.1 Automotive OEM Demand Forecasting: A Comparative Study of Forecasting Algorithms and Strategies	13
2.2 Reframing Demand Forecasting: a Two-Fold Approach for Lumpy and Intermittent Demand	33
3 Probability Calibration	55
3.1 Results amendment: Results and Evaluation of Probability Calibration Metrics and Techniques	79
3.2 Complementary Results: Spearman Correlations and Their Statistical Significance	81
4 Providing Explainability to Demand Forecasts	83
4.1 Explainable Demand Forecasting: A Data Mining Goldmine	85
4.2 Explaining Bad Forecasts in Global Time Series Models	113
5 Conclusions and Future Work	137
5.1 Contributions to Science	138
5.2 Future Work	139
References	141
Bibliography	147
Biography	149

Abbreviations

ACE ... Adaptive Calibration Error

AUC ... Area Under Curve of the Receiver Operating Characteristic
ROC

ECE ... Expected Calibration Error

MASE.. Mean Absolute Scaled Error

MCE ... Maximum Calibration Error

SPEC... Stock-keeping-oriented Prediction Error Cost

Chapter 1

Introduction

1.1 General Introduction

The world we live in has become increasingly digitalized. The development of new technologies and their increasing democratization have enabled the digitalization of every aspect of our lives. Information availability has enabled the development of machine learning models to address various needs. Digitalization of the Industry is no exception. The Industry 4.0 paradigm, first presented at the Hannover Industrial Fair in 2011 [1], [2], envisions using the latest technologies and paradigms to foster manufacturing innovation and address challenges related to reducing manufacturing times, enhancing quality, enabling predictive maintenance, mass customization, and smart product lifecycle management. Many consider the Industry 4.0 paradigm as a new industrial revolution and artificial intelligence as one of the key pillars of this revolution.

While humans have always been fascinated with foreseeing the future, we can think of this ability in two ways. First, it could be that the future is revealed to us (prophecies). Second, it could mean the capability to identify specific patterns that are not visible to the majority of the counterparts. This second ability would enable one to tell about the future with a certain degree of certainty, which may not be perfect but often realized. The latest is the case of many machine learning forecasting models. Machine learning models can learn from a wide range of data sources, learn correlations between the information contained among the features, and issue a prediction. One such case is demand forecasts. While the demand event occurrence can be forecasted with a prediction score, it is most intuitive to humans when such a score is probabilistic.

Developing the capability to identify patterns can be considered a learning process. Therefore, it is important to consider the factors that influence such a process. Some psychologists consider that the learning process in humans has three stages related to the learning and memory process: encoding (learning of information), storage (maintaining information over time), and retrieval (accessing the information when needed) [3]. Furthermore, multiple factors influence the learning process's speed (or success), such as the biological, psychometric, and social intelligence [4], and quality of instruction [5], [6]. In the broader sense, we could consider psychometric intelligence as the processing speed and the quality of instruction as the capability to correctly frame the problem at hand and follow an appropriate methodology (employ suitable algorithms). The last abovementioned elements do not only influence human learning but machine learning too. Furthermore, assessing such learning requires adequate metrics to understand whether key aspects were correctly learned and to what extent. Finally, the storage and ability to access the knowledge when needed relate to the machine learning models' predictive capabilities. This research will recurrently consider how to enhance learning and measure success. We address learning

and measure successful learning in the context of demand forecasting. Furthermore, we also consider probability calibration: how to turn predictive scores into real probabilities of occurrence for the event under consideration. In this direction, we do not develop novel calibration methods but rather address how the calibration quality can be better measured and how it can be estimated when no ground truth exists, given certain assumptions and constraints.

1.2 Motivation

Business originates from the fit of four interrelated components: people (who participate in the venture), context (elements influencing success or failure that the entrepreneur cannot control), deal (who satisfies the needs (something essential to human survival) or wants (something desired by the human person), to whom and when the deliveries and receipt take place), and opportunity (investment performed hoping a future return) [7]. The demand for such needs and wants is determined by the price of the good or service, the buyer's income, the prices of related goods or services (complementary or substitutes), the consumers' preferences, expectations regarding future prices, and the market size. To efficiently organize production, companies create demand forecasts, which guide multiple business activities (e.g., financial planning and risk assessment) and drive supply chain processes (e.g., raw material planning and purchases, inventory optimization). Such demand forecasts are created by examining past demand data and plan outlooks. Additional data sources can provide valuable information regarding factors affecting demand. Accurate demand forecasts reduce inefficiencies and can affect customer satisfaction. To enhance the accuracy of such forecasts, multiple improvements have been introduced. One is the Electronic Data Interchange software, whose implementation aimed to increase transparency between related parties by timely sharing manufacturing processes' data. The use of machine learning has been another step forward. Some benefits of machine learning models are that they are not prone to many biases the humans are, provide means for identifying patterns, and integrate a wide range of sources of information related to forecasting processes.

Multiple schemas to characterize demand have been proposed in the literature. Among them, we find a quadrant proposed by Syntetos et al. [8], which divides demand based on the frequency of demand events and the demand size. They name the demand patterns as *smooth* (regular demand occurrence, little demand size variability), *erratic* (regular demand occurrence, substantial demand size variability), *intermittent* (irregular demand occurrence, little demand size variability), and *lumpy* (irregular demand occurrence, strong demand size variability). In the literature, the four kinds of demand have been addressed as time series problems. Consequently, the models' performance has been measured through regression metrics. For cases where demand was sparse, much effort was invested into (a) mitigating such sparsity (e.g., through aggregation) or (b) devising metrics that would be less sensitive to such sparseness. This thesis agrees that demand forecasting with frequent demand events should be addressed as a time series problem. When demand is frequent, we assume demand events will always occur, and we must only learn to forecast the demand quantity. On the other hand, we do not consider it the right way to frame a demand forecasting problem when demand events are infrequent. In such cases, an additional learning goal must be considered: to learn to predict whether demand will occur. Consequently, the metrics required to measure learning success must be changed so that we understand to what extent both learning goals (determine whether demand will take place and the quantity of such demand) are satisfied.

Whenever we consider the occurrence of an event, we may be interested in thinking

about it in terms of probabilities. While machine learning models issue predictive scores, such scores can be calibrated into probability scores using a calibrator. The calibrator learns a mapping between the predictive scores and actual outcomes. A good calibrator will map the predictive scores to probability scores, ensuring that for the mean predicted probability for a class, the fraction of positive events for that class will be equal or close to the mean predicted probability value. Expected Calibration Error (ECE) is one of the most widely adopted metrics to measure calibrator quality. The metric is calculated as a weighted average of the accuracy and prediction error across the bins. It is weighted on the relative number of samples in each bin. Nevertheless, some shortcomings have been observed. Among them, we mention that only bins with some prediction are considered and that the metric does not inherently support multiclass scenarios. Furthermore, calibration techniques and metrics assume ground truth data is available to perform the calibration. In this thesis, we advocate that calibration measurements can be performed when no ground truth (human-labeled information regarding the occurrence of the event under consideration) is available. Therefore, we enable research on probability calibration when no ground truth is available.

In the following subsections, we introduce the problem of demand forecasting and probability calibration in greater detail.

1.3 Demand Forecasting Approaches and Metrics

Demand is a principle of economics that captures the consumer's desire to buy a certain good. Multiple demand categorizations have been proposed in the literature to determine material planning strategies, organize the production of such goods, and support stock management. Furthermore, such categorizations have been useful in deciding which forecasting technique to use in each case based on specific demand characteristics. Many authors proposed categorizing the demand types into three categories. Flores et al. [9] determined three categories based on the products' cost-volume share (ABC categorization): *A* for items with a large cost-volume share, *B* for items with a moderate cost-volume share, and *C* for items with a low cost-volume share. [10], on the other hand, proposed dividing the products according to their demand velocity from an inventory perspective (FSN categorization), establishing three categories: *F* (fast-moving), *S* (slow-moving), and *N* (non-moving). Scholz et al. [11] focused on the uncertainty of demand occurrence, and established the XYZ categorization. Such classification considers items with stable demand (*X*), items with volatile demand occurrence (*Z*), and items with a demand that falls in between *X* and *Z* demands (*Y*). Finally, Botter et al. [12] considered dividing products into three categories, based on how strong a disruption they could cause to the manufacturing process if not in stock (VED categorization): *V* (vital parts - expected to cause strong disruptions if not present in stock), *E* (essential parts - expected to cause moderate disruptions if not available in stock), and *D* (desirable parts - expected to cause minor disruptions if not present in stock). Some of the abovementioned categorizations were combined. For example, Scholz et al. [11] combined the ABC and XYZ analyses, while Nallusamy [13] combined the ABC and FSN analyses. Further categorizations were proposed considering the similar demand characteristics as considered by the XYZ analysis. Croston [14] proposed determining demand types based on demand size and inter-demand intervals. Williams [15] established five categories, considering the demand variance in the size and number of orders given a particular lead time. Eaves et al. [16] evolved this approach and suggested classifying demand considering the lead time variability, transaction rate variability, and demand size variability. Finally, Syntetos et al. [8] proposed classifying demand into four quadrants based on the frequency of the demand events and

the fluctuation of the demand quantities. They named the demand patterns as *smooth* (regular demand occurrence, little demand size variability), *erratic* (regular demand occurrence, substantial demand size variability), *intermittent* (irregular demand occurrence, little demand size variability), and *lumpy* (irregular demand occurrence, strong demand size variability).

While the research presented in this thesis involved machine learning models addressing the four types of demand as characterized by Syntetos et al. [8], the most important contributions relate to the approach addressing lumpy and intermittent demand. Multiple authors investigated the causes behind demand sparsity. Verganti [17] and Bartezzaghi [18] considered it a consequence of market characteristics (e.g., numerousness and heterogeneity of customers, the variety of customer requests, or the frequency at which customers place the orders). Furthermore, demand sparsity is related to the granularity level at which the demand is considered and mitigated by some authors by decreasing the granularity level (Nikolopoulos et al. [19]). Lumpy and intermittent demand is considered a relevant problem in demand forecasting, given a considerable proportion of any organization's stock value behaves according to such a demand (Babai et al. [20]). In particular, Johnston et al. [21] describe a use case where 75% of items had a lumpy demand and accounted for 40% of the company's revenue and 60% of stock investment. Furthermore, Amin-Naseri et al. [22] described multiple authors who found lumpy demand patterns are widespread, especially among the aviation, automotive, and telecommunication industries.

Both abovementioned demand types are considered among the most challenging demand forecasting problems given they both present sparse demand events and, therefore, represent an additional challenge to accurately estimating demand quantity (Amin-Naseri et al. [22] and Mukhopadhyay et al. [23]). Lumpy and intermittent demand have a characteristic in common: many periods have no demand. Therefore, many time series models get distorted, failing to provide an accurate prediction. Furthermore, such distortion is also seen in the regression metrics, making it difficult to determine the quality of the forecasts where zero forecasts dominate along the time series. This thesis presents a novel approach toward lumpy and intermittent demand forecasting. We advocate that for a fixed forecasting horizon, such demand is best forecasted using a two-fold model: a classification model to predict demand occurrence, and a regression model, to predict the demand quantity. By dividing the forecasting problem into these two dimensions, we also provide insights into the reasons affecting the forecast quality: whether we are failing to forecast demand occurrence or the demand quantity. Based on this approach, we propose a set of metrics to assess the quality of the forecast: ROC AUC to determine the quality of the classifier, variations of the MASE metric to measure the quality of the regression component, and the SPEC metric to determine the impact of the forecast on the stock costs.

To obtain superior results when forecasting demand, we trained global models. Global models are built with all experimental data available (in our case, with demand data corresponding to multiple products), in contrast to local models, which only consider data of a single product. While global models have been used in the past, there has recently been a renewed interest in them (Petropoulos et al. [24] and Makridakis et al. [25], [26]) following the results obtained by deep learning approaches and competition results (e.g., in M4 and M5 time series competitions). Furthermore, research has shown that global models can be trained for any dataset, and their performance should be equal to or better than that achieved by local models (Montero-Manso et al. [27]). In demand forecasting, global models can learn patterns shared across multiple products, enhancing performance. Furthermore, global time series models reduce the number of models to be deployed into production, therefore reducing efforts regarding human supervision, deployment, monitoring, and maintenance (Montero-Manso et al. [27]).

While an accurate prediction of when the demand event occurs is of utmost importance, a prediction score alone does not provide an intuitive interpretation to the human. Furthermore, multiple models produce different predictive score distributions. Therefore, an additional model (calibrator) can be trained to map the original predictive scores to probabilities (in our case, probability of demand occurrence for sparse demands). While much research has been devoted to measuring the quality of such calibrators, it is assumed that ground truth is required to measure and calibrate machine learning models. This thesis presents a novel approach to measuring calibration quality, addressing the shortcomings of some widely adopted metrics. Furthermore, we show that the newly proposed metrics can estimate the calibrators' quality without using ground truth. We consider this of utmost importance, given no additional data labeling is required to assess the calibrators' quality. We envision this finding could be helpful in at least two scenarios: (i) monitoring the performance of the calibrator over time without incurring labeling costs and (ii) training a calibrator without ground truth. While we performed some experiments on training a calibrator without ground truth, the first results were not promising, and more research must be invested in this direction.

1.4 Probability Calibration of Machine Learning Models

Probability calibration is the process of calibrating a model to return the true likelihood of an event. Such calibration is achieved by training a calibrator model that maps the classifier's predictive scores into probability scores (Song et al. [28]). By predicting probability scores, the decision rules created on top of the calibrator remain stable regardless of the underlying classification model. Such stability is achieved by abstracting the decision rules from the particular predictive scores of each classification model through a calibrator. The decision rules can account for different class priors or misclassification costs. While on the optimal decision boundary for hypothetical Bayes-optimal probabilities, the expected costs of true positives and true negatives are equal, the decision boundary can be altered based on specific circumstances (e.g., different costs of false positives and false negatives or class imbalance).

Miscalibration can occur due to underconfidence (the classifier better separates the classes than what would be expected from the probability scores) or overconfidence (the classifier does not separate the classes as well as it is expected from the probability scores). While many calibration methods have been developed, most do not inherently address the multi-class problem. Furthermore, at least three definitions exist on what is considered to be a multi-class calibrated model: (a) confidence calibration (considers the highest predicted probability only (Guo et al. [29])), (b) class-wise calibration (takes into account only the marginal probabilities (Zadrozny et al. [30])), and (c) multi-class calibration (takes into account all of the predicted probabilities in a vector).

For probability calibration, two sources of error are considered: (a) refinement error (likelihood of different classes receiving the same probability forecast) and (b) calibration error (deviation of given forecasts from the line denoting the expected outcomes for a perfectly calibrated model) (Song et al. [28]).

The calibration quality can be best assessed visually through calibration plots (a.k.a reliability diagrams). Nevertheless, multiple metrics have been devised to assess calibration quality. The Expected Calibration Error (ECE) is one of the most widespread calibration quality metrics (Possoco et al. [31]). While defined for a binary classification problem, many variations have been developed to address the multi-class scenarios. For binary classification, it is defined as the weighted average gap across all bins in a reliability diagram. The weighted average is computed considering the number of instances in each bin. For

multi-class scenarios, confidence-ECE is computed considering the highest predicted scores for each instance and then binarizing the ground truth by checking whether the highest score was correct regarding the target class. Classwise ECE, on the other hand, is defined as the average gap across all classwise reliability diagrams. Among the ECE shortcomings, we must mention its sensitivity to the number of bins considered, the fact that bins with zero predictions are not considered in the metric, and a truly multi-class ECE has not been proposed yet. Evaluating multi-class calibration is considered to still be an open research problem. Among other metrics, we find Adaptive Calibration Error (ACE), Maximum Calibration Error (MCE), and negative log-likelihood. The Negative Log-Likelihood is considered a standard measure of a calibrator’s quality and is minimized if and only if the prediction matches the ground truth conditional distribution. The ACE evolves from the ECE metric, and instead of fixed-sized intervals, it uses bin intervals of different sizes to ensure that each bin contains an equal number of samples (Nixon et al. [32]). Finally, the MCE aims to minimize the worst-case deviation between confidence and accuracy and is best suited where reliable confidence measures must be given (Naeini et al. [33]).

While much research was invested into probabilities calibration, to the extent of our knowledge, almost no research addressed the measurement of the calibration quality, given a lack of ground truth. We consider this approach to be feasible, at least under certain assumptions: (i) there is no concept drift, (ii) the feature values do not shift towards unseen regions of the input space, and (iii) the standard deviation of the noise observed around the predicted target depends on the input features. We propose a new set of calibration metrics, which better capture the quality of a calibrator when compared to the widespread ECE metric. The proposed metrics address some of the ECE metric shortcomings. Furthermore, we show they can be used to assess the quality of a calibrator in production environments considering the abovementioned constraints.

1.5 Aims and Hypotheses

Several studies have previously explored mechanisms for forecasting demand for lumpy and intermittent demand. Nevertheless, that research mainly considered lumpy and intermittent demand to be framed as a regular time series problem, focusing on attenuating the influence of zero forecasts on the forecasting models or the metrics used to assess the forecast performance. One of the primary goals of the research presented in this thesis was to reframe demand forecasting for lumpy and intermittent demands, showing that it could be more effectively addressed if considered a two-fold problem: demand occurrence prediction and demand quantity prediction. To address these two sub-problems, we develop a classification and regression model. Furthermore, we propose a set of metrics that assess the quality of each model and the forecast impact on inventory when joined into a single forecast. Another important goal of the research presented in this thesis concerns model probability calibration. We argue that calibration quality can be measured even when no ground truth is available. We develop a set of metrics and show a correlation between metrics measured with and without ground truth. Furthermore, we show that the proposed metrics solve shortcomings of widely adopted metrics used to measure calibration quality.

Hypothesis 1: *Using a two-fold model, predicting demand occurrence and demand quantity separately, could achieve better demand forecasts for lumpy and intermittent demand than current state-of-the-art models described in the scientific literature.*

Prior state of the art: It was known that a large number of zero-demand intervals affected the forecasting models and metrics used to assess their quality. Additionally, while features were devised to describe either demand occurrence or demand quantity, the

demand forecasting models were optimized for regression metrics. By doing so, the lack of predictive performance could not be attributed to whether it failed to predict demand occurrence or failed to predict demand quantity.

Aim: We have taken a novel perspective on lumpy and intermittent demand forecasting. We aimed to show that the lumpy and intermittent demand can be predicted using a two-fold approach, predicting demand occurrence and demand quantity separately. To test our approach, we perform experiments on real-world data and compare our approach against models described in the scientific literature across multiple metrics.

Hypothesis 2: *Forecasting models' performance for intermittent and lumpy demand is better assessed (when compared to current approaches) using three separate metrics: AUC ROC for event occurrence, Mean Absolute Scaled Error (MASE) to assess how accurate the demand quantities were predicted, and Stock-keeping-oriented Prediction Error Costs (SPEC) to assess how prediction errors impact stock over time.*

Prior state of the art: While many metrics have been devised to measure the quality of forecasts for lumpy and intermittent demands, they treated the demand forecasting problem as a regression problem. Therefore, metrics were developed to mitigate the influence of zero demand while achieving this to a greater or lesser extent. By using a single regression metric to assess the models' performance, it was not clear whether the models' lack of performance was related to a failure to predict the demand event occurrence or the demand size.

Aim: We have proposed a new set of metrics to evaluate the quality of demand forecasts. First, we measure how accurately the demand occurrence is predicted. Second, we measure how accurately the demand quantity is predicted. Finally, we measure the expected cost of a given forecast on inventories.

Hypothesis 3: *The quality of a probabilistic calibration on a classification model can be measured without ground truth.*

Prior state of the art: While many metrics have been proposed to measure the calibration quality, all assume ground truth exists.

Aim: We have proposed a new set of metrics that can determine the calibration quality when no ground truth exists. We do so by leveraging information regarding the predictions' distribution and the expected distribution for a perfect calibrator.

1.6 Scientific Contributions

The major scientific contributions of this thesis can be grouped based on their relation to the three hypotheses we have outlined in Section 1.5. These correspond to lumpy and intermittent demand forecasting (a novel forecasting approach, a set of metrics to measure performance) and probability calibration.

SC 1. Development of a novel approach to forecasting intermittent and lumpy demand. We did so in our work addressing demand forecasting for lumpy and intermittent demand (Rožanec et al. [34]).

We have shown that the abovementioned approach achieves state-of-the-art performance for lumpy and intermittent demand. It was determined that the proposed method improves primarily by enhancing the quality of forecasts regarding the demand event occurrence. This is a significant novelty, as previous works considered features related to demand occurrence but did not address the forecasting problem as two separate problems: forecasting demand occurrence and demand size. We achieve state-of-the-art performance by creating a classification model for forecasting demand occurrence and a regression model for forecasting demand size when demand occurs.

SC 2. Definition of a novel set of metrics for assessing the quality of forecasting intermittent and lumpy demands. We did so in our work addressing demand forecasting for lumpy and intermittent demand (Rožanec et al. [34]).

The quality of demand forecasts for lumpy and intermittent demands can be better assessed if the following dimensions are measured: (a) how accurately the demand events are predicted, (b) how accurately the demand sizes are predicted, and (c) the cost of demand forecasts on inventory. The results show that the proposed set of metrics has shown that most demand forecasting models for lumpy and intermittent demand fail to predict demand occurrence (they do so at a chance level - the classifier is not able to rank a random positive sample higher than a random negative one: the model cannot distinguish between positive and negative classes). Furthermore, it was shown that (in)ability to predict demand occurrence was key to the final forecast performance.

SC 3. Development of novel metrics to assess the quality of probability calibration of classification models without using ground truth. We did so in our work addressing model calibration in the context of automated visual defect inspection (Rožanec et al. [35]).

The results show that the metrics could better assess the calibration quality than a widely adopted calibration quality metric. Such assessment was performed by comparing metric scores and calibration plots when ground truth was available. The results show that the proposed metrics can determine the quality of calibration even when no ground truth is available when certain preconditions are met.

Most of the listed contributions represent steps in a new way of conceiving lumpy and intermittent demand forecasting. The experimentally observed improvements suggest that reframing demand forecasting of lumpy and intermittent demand into two sub-problems (forecast of demand occurrence and quantity) provides tangible benefits and is a promising direction for further research. Furthermore, these could be the first steps towards reframing sparse time series forecasting.

Our contributions regarding models' probabilistic calibration represent a new way of conceiving probabilities calibration. While scientific literature considers ground truth is required to perform such calibrations, we have defined a set of metrics that can assess the calibration quality when no ground truth is available. This contribution reduces costs associated with data labeling and opens the door to new calibration approaches.

1.7 Thesis Structure

The thesis starts with a brief outline of the general background and motivation, describing the context of the problems related to demand forecasting. We focus on lumpy and intermittent demand forecasting, given that one of this thesis's most important contributions is a novel approach to forecasting such demands. Second, we briefly describe problems related to the calibration of machine learning models. The importance of such calibration in this thesis is two-fold. First, machine learning model calibration is required to transform any classification model predicting demand occurrence into a probabilistic score. By doing so, we provide an interpretable score that can be understood by anyone leveraging the demand forecast. Second, the calibration metrics described in this thesis address shortcomings of widely adopted calibration metrics and enable measuring calibration quality even when no ground truth is available. This last characteristic is of utmost importance, given it reduces costs associated with data labeling. Furthermore, it opens new research opportunities. Such information can be exploited to enhance the calibration quality or develop new approaches.

Chapter 2 describes how we tackled demand forecasting with machine learning. We

focused on point forecasts for different types of demand. The results regarding erratic and smooth demand are presented in the paper *Automotive OEM Demand Forecasting: A Comparative Study of Forecasting Algorithms and Strategies* (Rožanec et al. [36]). In particular, we frame demand forecasting as a regression problem and use multiple features to capture time series characteristics and relevant contextual data. The research shows how global models outperform local ones. Furthermore, multiple data pooling strategies are researched to constrain the magnitude of error of the global model. The results show that pooling product data based on the past demand magnitude successfully constrain the forecasting error magnitude. A different approach is followed to forecast lumpy and intermittent demand (SC 1). We describe a two-fold demand forecasting approach in *Reframing demand forecasting: a two-fold approach for lumpy and intermittent demand* (Rožanec et al. [34]). Furthermore, we describe and advocate for a set of metrics that measure the quality of such forecasts more precisely (SC 2).

Chapter 3 describes a novel approach to measuring the probabilities calibration quality. We propose a set of metrics that do not require ground truth to determine the calibration quality. The research was performed in the context of automated visual defect recognition. Machine learning model predictions were calibrated using multiple probability calibration techniques. Their quality was measured with the Expected Calibration Error (ECE) metric, a set of novel metrics proposed in our research *Active learning and novel model calibration measurements for automated visual inspection in manufacturing* (Rožanec et al. [35]), and through visual evaluation by analyzing the calibration plots. The research shows the proposed metrics successfully capture the calibration quality, outperforming the Expected Calibration Error when ground truth is available (SC 3). Furthermore, the results show the proposed metrics can capture meaningful information regarding the calibration quality when no ground truth is available.

Chapter 4 highlights how explanations can be provided for demand forecasting models. We use well-known Explainable Artificial Intelligence techniques to extract information on meaningful features and enrich such information with complementary insights and data.

Chapter 5 concludes the thesis by looking back at the scientific contributions and stating objectives and directions for future work regarding demand forecasting and probability calibration.

Chapter 2

Demand Forecasting

Demand forecasting is a critical supply chain component. Accurate demand forecasting can mitigate the bullwhip effect, which results from the inefficient propagation of information across multiple players.

This chapter comprises two journal papers devoted to the demand forecasting problem. The first one is titled *Automotive OEM Demand Forecasting: A Comparative Study of Forecasting Algorithms and Strategies* and was published in MDPI Applied Sciences in 2022 (Q2 - IF 2.838) [36]. The paper compares local and global machine learning models' performance when forecasting smooth and erratic demand. It demonstrates that global models' errors can be constrained by pooling products based on the magnitude of their past demand. Nevertheless, given global models can learn patterns that deceive forecasts for particular instances, some research was performed to explain them. We present such research in Section 4 in the journal paper *Explaining Bad Forecasts in Global Time Series Models* [37]. Explainability of machine learning models is of utmost relevance, given it can help to debug machine learning models and make them more accurate over time.

The second journal paper considered in this chapter is titled *Reframing demand forecasting: a two-fold approach for lumpy and intermittent demand* and published in MDPI Sustainability in 2022 (Q1 - IF 3.889) [34]. The paper introduces a novel demand forecasting approach for lumpy and intermittent demands. Given that the abovementioned demands are characterized by sparse demand occurrence, demand forecasts must accurately predict demand occurrence and quantity. This was not the case for most scientific research for these particular demand types. Most authors considered forecasting lumpy and intermittent demand a time series problem (Pinçe et al. [38]), framing the problem similarly to when predicting smooth and erratic demands. Specific models and metrics were developed to mitigate the sparseness of demand events. This thesis proposes addressing lumpy and intermittent demand forecasting with two models: (a) a classification model to forecast demand occurrence and (b) a regression model to forecast demand size. The results show that the proposed model achieves state-of-the-art performance. A preprint of the paper was first published on arxiv on March 23th, 2021. Since then, it has been cited by several authors (Sigwarth [39], Dijkman et al. [40], Zhuang et al. [41], Sarlo et al. [42], and Chanada et al. [43]). Furthermore, Zhuang et al. [41] implemented variations of the approach we described and adopted two metrics we proposed for evaluating such models (ROC AUC and MASE, the authors did not analyze the stock keeping costs). They trained the models on a real-world dataset corresponding to intermittent demand of 3,089 SKUs of equipment manufacturing and maintenance replacement of several car types. While the dataset had a daily granularity between 2018-08-27 and 2020-12-28, it was aggregated to a weekly level to reduce the extreme sparsity. The authors reported results comparable to the ones we achieved on our dataset.

The analysis of results in the abovementioned paper has shown that the forecasting horizon had little influence on the classification performance. This is a feasible outcome if the features used to predict demand occurrence adequately capture information regarding demand occurrence and if such information has little variability across time. This last factor is perfectly possible in scenarios where demand events are infrequent: likely, the conditions leading to a product demand did not change in the time between the two time horizons. Finally, if the shipment day from the manufacturing factory is regarded as the demand date, rules related to shipments could provide solid hints on demand occurrence. E.g., shipping agreements that products to a client are shipped regularly on a particular weekday. Such agreements and rules would vary only exceptionally and remain invariant to the time horizons.

2.1 Automotive OEM Demand Forecasting: A Comparative Study of Forecasting Algorithms and Strategies

Demand forecasting is a crucial component of demand management. It directly impacts manufacturing companies' planning, revenues, and actors along the supply chain. Therefore, the demand forecasts are expected to be accurate so that they can be relied upon. Furthermore, a certain degree of explainability could be desired to enable responsible decision-making. In *Automotive OEM Demand Forecasting: A Comparative Study of Forecasting Algorithms and Strategies* [36], we compare twenty-one baseline, statistical, and machine learning algorithms to forecast smooth and erratic demand. We do so on a real-world use case scenario related to a European original equipment manufacturer targeting the global automotive industry market. In particular, we explore how global models can supersede the performance of local ones and study data pooling strategies that could lead to more accurate predictions while constraining prediction errors. Our intuition behind the experiments is that providing more data and a set of features capturing invariants of the time series under consideration could help the machine learning algorithms learn meaningful patterns and lead to better results.

In this paper, among the selected features, we considered invariant features related to the economic context (e.g., unemployment rate or gross domestic product) and not the product demands. Nevertheless, when learning on demand data from multiple products, the model may learn patterns and values not observed for the product under consideration. We analyzed two pooling strategies: (a) median magnitude of past demand and (b) demand type. Our intuition behind grouping data of products with the same median magnitude of past demand was that a model forecasting values out of the usual magnitude observed for a particular product could significantly degrade the model's overall performance. By constraining the data in the training set to products with the same magnitude of past demand, forecasts would be mostly constrained to a particular magnitude and avoid costly errors. On the other hand, our intuition behind pooling data by demand type was that similar demand distributions should reduce noise when learning the behavior of a particular demand, leading to better overall forecast results. Our experiments show that global models achieved the best performance and that pooling data by the median magnitude of past demand effectively reduced the overall forecasting error.

Looking in retrospect, we consider the feature set of the machine learning models could be enriched with a set of features invariant to the demand time series magnitudes. By doing so, we could better capture time series patterns, such as trends and seasonality, own to the particular demand type on top of the context considered with external datasets used for data enrichment. Furthermore, we could have used other feature selection techniques to filter out redundant features and probably improve the models' overall performance. The future work envisioned in the paper was mainly addressed with the research presented in Section 4.

Article

Automotive OEM Demand Forecasting: A Comparative Study of Forecasting Algorithms and Strategies

Jože M. Rožanec ^{1,2,3,*} , Blaž Kažič ^{1,2,3}, Maja Škrjanc ^{1,2,3}, Blaž Fortuna ^{1,2}  and Dunja Mladenec ¹ 

¹ Jožef Stefan Institute, Jamova 39, 1000 Ljubljana, Slovenia; blaz.kazic@qlector.com (B.K.); maja.skrjanc@qlector.com (M.Š.); blaz.fortuna@qlector.com (B.F.); dunja.mladenec@ijs.si (D.M.)

² Qlector d.o.o., Rovšnikova 7, 1000 Ljubljana, Slovenia

³ Jožef Stefan International Postgraduate School, Jamova 39, 1000 Ljubljana, Slovenia

* Correspondence: joze.rozanec@ijs.si

Featured Application: The outcomes of this work can be applied to B2B discrete demand forecasting in the automotive industry and probably generalized to other demand forecasting domains.

Abstract: Demand forecasting is a crucial component of demand management, directly impacting manufacturing companies' planning, revenues, and actors through the supply chain. We evaluate 21 baseline, statistical, and machine learning algorithms to forecast smooth and erratic demand on a real-world use case scenario. The products' data were obtained from a European original equipment manufacturer targeting the global automotive industry market. Our research shows that global machine learning models achieve superior performance than local models. We show that forecast errors from global models can be constrained by pooling product data based on the past demand magnitude. We also propose a set of metrics and criteria for a comprehensive understanding of demand forecasting models' performance.

Keywords: demand forecasting; smart manufacturing; artificial intelligence; supply chain agility; digital twin



Citation: Rožanec, J.M.; Kažič, B.; Škrjanc, M.; Fortuna, B.; Mladenec, D. Automotive OEM Demand Forecasting: A Comparative Study of Forecasting Algorithms and Strategies. *Appl. Sci.* **2021**, *11*, 6787. <https://doi.org/10.3390/app11156787>

Academic Editor: Grzegorz Dudek

Received: 30 June 2021

Accepted: 21 July 2021

Published: 23 July 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Supply (the “amount of something ready to be used” [1]) and demand (“the fact of customers buying goods... and the amount that they buy” [1] at a given point in time) are two key elements continually interacting in the market. The ability to accurately forecast future demand enables manufacturers to make operational and strategic decisions on resources (allocation and scheduling of raw material and tooling), workers (scheduling, training, promotions, or hiring), manufactured products (market share increase and production diversification), and logistics for deliveries [2]. Accurate demand forecasts reduce inefficiencies, such as high stocks or stock shortages, which have a direct impact on the supply chain (e.g., reducing the bullwhip effect [3,4]), and prevent a loss of reputation [5].

There is consensus that greater transparency between related parties helps to mitigate the issues mentioned above [6,7]. Such transparency can be achieved through automation and digitalization (e.g., implementing Electronic Data Interchange software), by sharing manufacturing processes' data, and making it timely available to internal stakeholders and relevant external parties where appropriate [8]. In addition, the ability to apply intelligence to multiple stages across the supply chain can improve its performance [9]. Such ability and the capability to get up-to-date information regarding any aspect of the manufacturing plant enable the creation of up-to-date forecasts and provide valuable insights for decision-making.

Multiple authors found that machine learning outperforms statistical methods for demand forecasting [10,11]. Machine learning methods can be used to train a single model

per target demand or a global model to address them all. When designing the models, it is crucial to consider which data is potentially relevant to such forecasts. Multiple factors may affect the product's demand. First, it is necessary to understand if the products can be considered inelastic (their demand is not sensitive to price fluctuations), complementary, or be substituted for alternative products. Second, intrinsic product qualities, such as being a perishable or luxury item, or their expected lifetime, may be relevant to demand. Finally, the manufacturer must also consider the kind of market it operates on and the customer expectations. When dealing with demand forecasting in the automotive industry, most authors do not use only demand data but also incorporate data regarding exogenous factors that influence demand.

Demand forecasting has direct consequences on decision-making. As such, forecasts are expected to be accurate so that they can be relied upon. When training machine learning models, more significant amounts of good quality data can help the model better learn patterns and provide more accurate forecasts. This intuition is considered when building global models. On the other side, while for local models, the forecast error is constrained to past data of a single time series, in global models, the forecasting error is influenced by patterns and values observed in other time series, which can lead to greater errors as well. In this work, we explore a strategy to constrain such forecasting errors in global models. Furthermore, providing a greater amount of data should be considered regarding products' demand and its context. To that end, we enrich the demand data with data from complementary data sources, such as world Gross Domestic Product (GDP), unemployment rates, and fuel prices.

This work compares 21 statistical and machine learning algorithms, building local and global forecasting models. We propose two data pooling strategies to develop global time series models. One of them successfully constrains the global time series models' forecasting errors. We also propose a set of metrics and criteria for the evaluation of demand forecasts for smooth and erratic demands [12]. The error bounding data pooling strategy enables us to gain the benefits of training machine learning models on larger amounts of data (increased forecast accuracy) while avoiding anomalous forecasts by constraining the magnitude of maximum forecasting errors. The metrics and evaluation criteria aim to characterize the given forecasts and provide better insight when deciding on the best-performing model. We expect the outcomes of this work to provide valuable insights for the development and assessment of demand forecasting models related to the automotive industry, introducing forecasting models and evaluation strategies previously not found in the scientific literature.

To evaluate the performance of our models, we consider the mean absolute scaled error (MASE) [13] and the R^2 -adjusted (R^2_{adj}) metrics. We compute the uncertainty ranges for each forecast and compare if differences between forecasts are statistically significant by performing a Wilcoxon paired rank test [14]. Finally, we analyze the proportion of products with forecasting errors below certain thresholds and the proportion of forecasts that result in under-estimates.

The rest of this paper is structured as follows. Section 2 presents related work. Section 3 describes the methodology we followed to gather and prepare data, create features, and build and evaluate the demand forecasting models. Section 4 details the experiments performed and the results obtained. Finally, Section 5 presents the conclusions and an outline of future work.

2. Related Work

Products' demand forecasting is a broad topic addressed by many authors in the scientific literature. Different demand patterns require specific approaches to address their characteristics. Multiple authors proposed demand classification schemas to understand which techniques are appropriate for a particular demand type. For example, the work in [15] focused on demand variance during lead times, while the work in [16] introduced the concept of average demand interval (ADI), which was later widely adopted.

$$ADI = \frac{\text{Total Periods}}{\text{Total Demand Buckets}} \quad (1)$$

The work in [12] complemented this view of demand introducing the coefficient of variation (CV). Both concepts allow us to divide demand types into quadrants, classifying them as intermittent, lumpy, smooth, and erratic demands (see Figure 1). Smooth and intermittent demand have little variability in demand quantities. Smooth demand has little variability regarding demand intervals over time, while intermittent demand displays a higher demand interval variability. Erratic and lumpy demands have higher variability in demand quantities, which comprehends an additional forecasting model challenge. Erratic demand has little variability regarding demand intervals over time. In lumpy ones, this is an essential factor to be considered. Following demand types proposed in [12], in this work, we focus on smooth and erratic demands.

$$CV = \frac{\text{Demand Standard Deviation}}{\text{Demand Mean}} \quad (2)$$

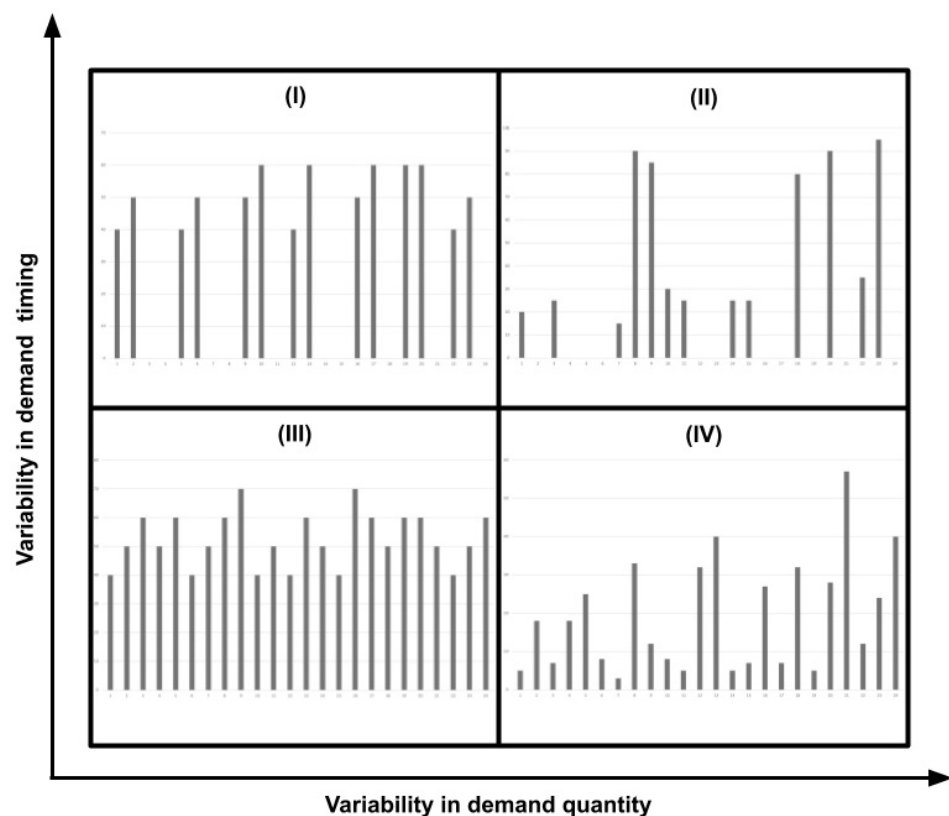


Figure 1. Demand types classification by Syntetos et al. [12]. Quadrants correspond to (I) intermittent, (II) lumpy, (III) smooth, and (IV) erratic demand types.

Planners, who regularly create demand forecasts, must understand the products they sell, the market they target, the economic context, and customer expectations. Over time they learn how buyers behave, the vast array of factors that can influence product demand, and create their estimates. Each planner can weigh different factors and have distinct ways to ponder them. Most of this information can be collected and fed to machine learning (ML) models, which learn how demand behaves over time to provide a forecast. In the scientific literature addressing demand forecasting in the automotive industry, most authors do not use only demand data, but also incorporate data regarding exogenous factors that influence demand, such as the effect of personal income on car ownership [17,18], or the effect of the GDP [5,17,19,20], inflation rate [18,19], unemployment rate [5], population density [20], and fuel prices [5,18,20–22] on vehicles demand.

Demand distributions can also be considered as another source of information for demand forecasting: research performed by many authors confirmed a relationship exists between demand types and demand distributions [23–26].

A wide range of models was explored in the literature addressing car, and car components demand forecasting. Ref. [5] developed a custom additive forecasting model with seasonal, trend, and calendar components. The authors used a phase average to compute the seasonal component, experimented with Multiple Linear Regression (MLR) and Support Vector Machine (SVM) for trend estimation, and used a Linear Regression to estimate the number of working days within a single forecasting period (calendar component). The models were built with car sales data from Germany, obtaining the best results when estimating trends with an SVM model and providing forecasts quarterly. Models' performance was measured using Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE) metrics. Ref. [18] compared three models: an adaptive network-based fuzzy inference system (ANFIS), an autoregressive integrated moving average model (ARIMA), and an artificial neural network (ANN). The forecasting models were built considering new automobile monthly sales in Taiwan, obtaining the best results with the ANFIS model. Ref. [21] developed an ANN model considering the inflation rate, pricing changes in crude oil, and past sales. They trained and evaluated the model on data from the Maruti Suzuki Ltd. company from India, measuring the models' performance with the Mean Squared Error (MSE) metric. Ref. [27] compared three models: ANFIS, ANN, and a Linear Regression. They are trained on car sales data from the Maruti car Industry in India and compared with the root mean squared error (RMSE) metric. The authors report that the best performance was obtained with the ANFIS model. Ref. [28] compared the ARIMA and the Holt–Winters models using demand data regarding remanufactured alternators and starters manufactured by an independent auto parts remanufacturer. They measured performance using the MAPE and average cumulative absolute percentage errors (CAPE) metrics, obtaining the best results for the Holt–Winters models. Ref. [29] developed three models (ANN, Linear Regression, and Exponential Regression) based on data from the Kia and Hyundai corporations in the US and Canada. Results were measured with the MSE metric, obtaining the best one with the ANN model. Ref. [19] analyzed the usage of genetic algorithms to tune the parameters from ANFIS models built with data from the Saipa group, a leading automobile manufacturer from Iran. Measuring RMSE and R^2 , they achieved the best results with ANFIS models tuned with genetic algorithms compared to ANFIS and ANN models without any tuning. Ref. [30] compared custom deep learning models trained on real-world products' data provided by a worldwide automotive original equipment manufacturer (OEM). Ref. [31] developed a long short-term memory (LSTM) model based on car parts sales data in Norway and compared it against Simple Exponential Smoothing, Croston, Syntetos-Boylan Approximation (SBA), Teunter-Syntetos-Babai, and Modified SBA. Best results were obtained with the LSTM model when comparing models' mean error (ME) and MSE. Ref. [22] developed tree models (autoregressive moving average (ARMA), Vector Autoregression (VAR) model, and the Vector Error Correction Model (VECM)) to forecast automobile sales in China. The models were compared based on their performance measured with RMSE and MAPE metrics, finding the best results with the VECM model. The VECM model was also applied by [20], when forecasting cars demand for the state of Sarawak in Malaysia. Finally, Ref. [32] compared forecasts obtained from different moving average (MA) algorithms (simple MA, weighted MA, and exponential MA) when applied to production and sales data from the Gabungan Industri Kendaraan Bermotor Indonesia. Considering the Mean Absolute Deviation, the best forecasts were obtained with the Exponential Moving Average.

Additional insights regarding demand forecasting can be found in research related to time series forecasting in other domains. Refs. [33,34] described the importance of time series preprocessing regarding trend and seasonality, though [35,36] found the ANN models could learn seasonality. The use of local and global forecasting models for time series forecasting was researched in detail by [35]. Local forecasting models model each

time series individually as separate regression problems. In contrast, global forecasting models assume there is enough similarity across the time series to build a single model to forecast them all. Researchers explored the use of global models either clustering time series [36,37], or creating a single model for time series that cannot be considered related to each other [38]. They achieved good performance in both cases.

3. Methodology

To address the demand forecasting problem, we followed a hybrid of the agile and cross-industry standard process for data mining (CRISP-DM) methodologies [39]. From the CRISP-DM methodology, we took the proposed steps: focus first on understanding the business and the available data, later tackle the data preparation and modeling, and, finally, evaluate the results. We did not follow these steps sequentially, but rather moved several times through them forward and backward, based on our understanding and feedback from end users, as is done in agile methodologies. We describe the work performed in each phase in the following subsections.

3.1. Business Understanding

The automotive industry accounts for one of the largest economies in the world, by revenue [40]. It is also considered a strong employment multiplier, a characteristic that is expected to grow stronger with the incorporation of complex digital technologies and the fusion with the digital industry [41]. Environmental concerns have prompted multiple policies and agreements, which foster the development of more environment friendly vehicles and rethinking of current mobility paradigms [42–44]. Nevertheless, global vehicle sales and automotive revenue are expected to continue to grow in the future [45,46].

Demand forecasting is a critical component to supply chain management as its outcomes directly affect the supply chain and manufacturing plant organization. In our specific case, demand forecasts for the automotive industry engine components worldwide were required on a monthly level, six weeks in advance. In Section 2, we highlighted related work, data, and techniques used by authors in the automotive industry. On top of data sources suggested in the literature for deriving machine learning features (past demand data, GDP, unemployment rates, and oil price), we incorporated three additional data sources based on experts' experience: Purchasing Managers' Index (PMI), copper prices, and sales plans.

PMI is a diffusion index obtained from monthly surveys sent to purchasing managers from multiple manufacturing companies. It summarizes expectations regarding whether the market will expand, contract, or stay the same and how strong the growth or contraction will be.

Prices of the products we forecast are tied to copper price variations used to manufacture them. Therefore, we consider the price of this metal and create derivative features to capture how it influences the products' price and how it may influence it in the future.

The strategic sales department creates sales plans on a yearly and quarterly basis. Experts consider projected sales to be a good proxy of future demand as they inform buyers' purchase intentions. We found research that backs their claim (see, e.g., in [47]), showing that purchase intentions contribute to the forecast's accuracy. The research done in [48] shows that purchase intentions are good predictors of future demand for durable products and that this accuracy is higher for short time horizons. Research also shows that the purchase intention bias can be adjusted with past sales data.

Much research was performed on the effect of aggregation on time series [34,49–51], showing that a higher aggregation improves forecast results. Though research shows optimal demand aggregation levels exist [52], we considered forecasts at a monthly level to reflect business requirements specific to our use case.

To understand demand forecasting models' desired behavior and performance, we consulted industry experts. They agreed that one-third of demand forecasts produced by planners have up to 30% error, and up to 20% forecasts may have more than 90% error.

They also pointed out that 40% of all forecasts result in under-estimates. When issuing a forecast, it is more desirable to have over-estimates than under-estimates. We consider these facts to assess the forecast results.

We address demand analysis in detail in Section 3.2.

3.2. Data Understanding

We make use of several data sources when forming features for machine learning, described in Table 1. We distinguish between internal data sources (non publicly available data regarding a manufacturing company provided by that same company) and external data sources used for the data enrichment process.

Table 1. Data sources. In the first and second columns, we indicate the kind of data we retrieve and its source. The third column provides information on how frequently new data is available. In contrast, the last column describes the aggregation level at which the data is published. Periodicity and aggregation levels can be at a yearly, quarterly, monthly, or daily level and are denoted by “Y”, “Q”, “M”, or “D”, respectively. The London Metal Exchange published copper prices for weekdays.

Data	Source	Periodicity	Aggregation Level
History of deliveries	Internal	D	D
Sales Plan	Internal	Y,Q	M
Gross Domestic Product (GDP)	World Bank	Y	Y
Unemployment rate	World Bank	Y	Y
Crude Oil price	World Bank	M	M
Purchasing Managers’ Index (PMI)	Institute of Supply Chain Management	M	M
Copper price	London Metal Exchange	D	D
Car sales	International Organization of Motor Vehicle Manufacturers	Y	Y

When performing preliminary data analysis over the seven years of data, we found that GDP, crude oil prices, PMI values, and demand (see Figure 2) show a different pattern before and after the *year 4* of our dataset. When searching for possible root causes, we observed that in *year 4* some significant economic and political events took place affecting the economy worldwide. Among them, we found a stock market crash of a relevant country, a decrease in crude oil production, and several political events that affected the market prospects.

We consider demand quantity as the executed orders of a given product leaving the manufacturing plant on a specific date. Even though demand data is available daily, we aggregate them monthly, satisfying business requirements and providing smoother curves and ease of forecasting. We also consider that months have different working days (due to weekdays and holidays). Thus, we computed the average demand per working day for each month. Future demand can be estimated using the average demand per working day, multiplying by the number of working days in the target month.

Based on the demand classification in [12], we analyzed how many products correspond to erratic and smooth demands. We create features to capture this behavior. We present the products demand segmentation in Table 2. From the works in [23,24,26,53], we understand that demands of a given type follow a certain distribution. Thus, most manufacturing companies’ products may have a slightly different demand behavior but share enough characteristics that would reflect common patterns. We observed that demand values for each product follow a geometric distribution.

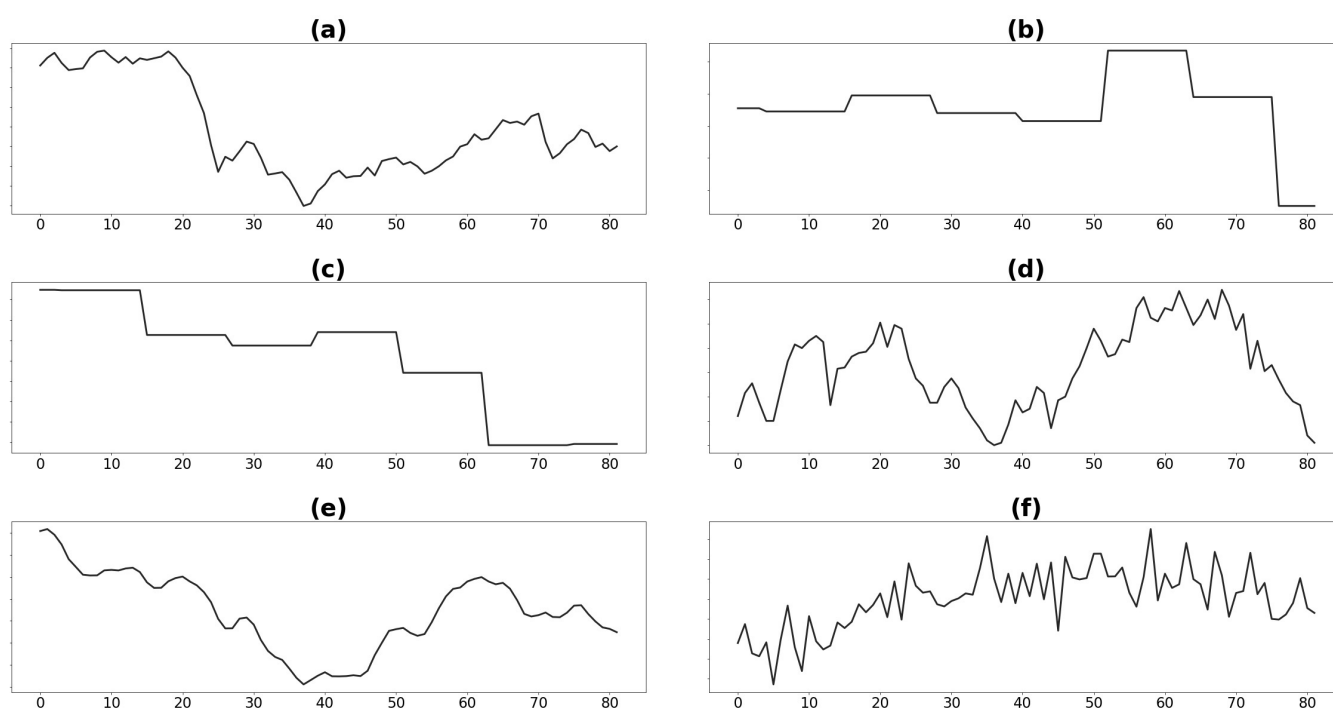


Figure 2. Median values for (a) crude oil price, (b) GDP, (c) unemployment rate worldwide, (d) PMI, (e) copper price (last three months average), and (f) demand.

Table 2. Demand segmentations, by demand type as per [12], and by demand magnitude, considering demanded quantities per month.

Years	By Demand Type		By Demand Magnitude				
	Smooth	Erratic	10	100	1 K	10 K	100 K
All years	13	43	13	2	5	26	10
Last 3 years	19	37	10	1	4	28	13

To discover potential patterns, we made use of different visualizations.

Demand seasonality was assessed with correlograms (see Figure 3), which show what lags in time most frequently display a statistically significant correlation (with a p -value = 0.05). Considering all data available, we found the strongest correlations for products at three, four, five, eight, and eleven months before the target month. However, we observed a different pattern in the last three years of data: the strongest correlations occurred at eight, ten, and eleven months before the target. Therefore, we choose only those statistically significant when analyzing correlation values, considering a confidence interval of 95%.

Plotting products' monthly demand for every year, as shown in Figure 4, we found that most products were likely to behave similarly over the years for a given month.

When assessing demand data sparsity, we analyzed how many non-zero demand data points we have for each product and the demand magnitudes we observe in each case. We present the data in Table 2. Higher aggregation levels regarding the time dimension allow reducing variability in time. However, aggregate data at a higher than monthly level are not applicable in our case.

3.3. Data Preparation

The first step we followed for data preparation was to remove records that would fall into the black period for any given point in time and thus avoid provide our models any indication about the future (except for the target we aim to predict). In our case, we

consider a forecasting horizon of six weeks until the beginning of the target month, as depicted in Figure 5.

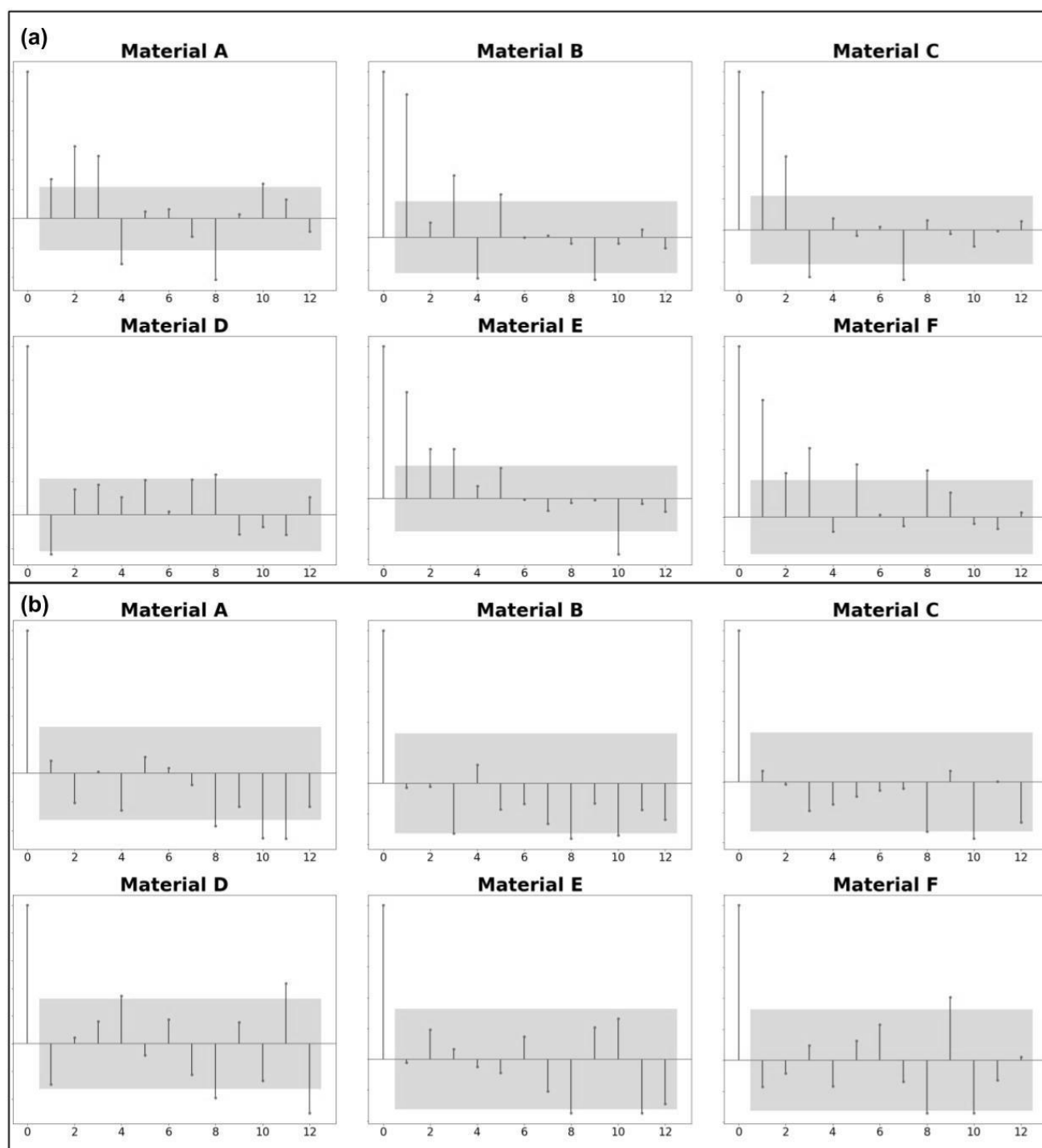


Figure 3. Sample demand correlograms, indicating seasonality patterns. The correlogram in panel (a) is computed over the seven years of data, while correlogram in (b) is computed over last three years.

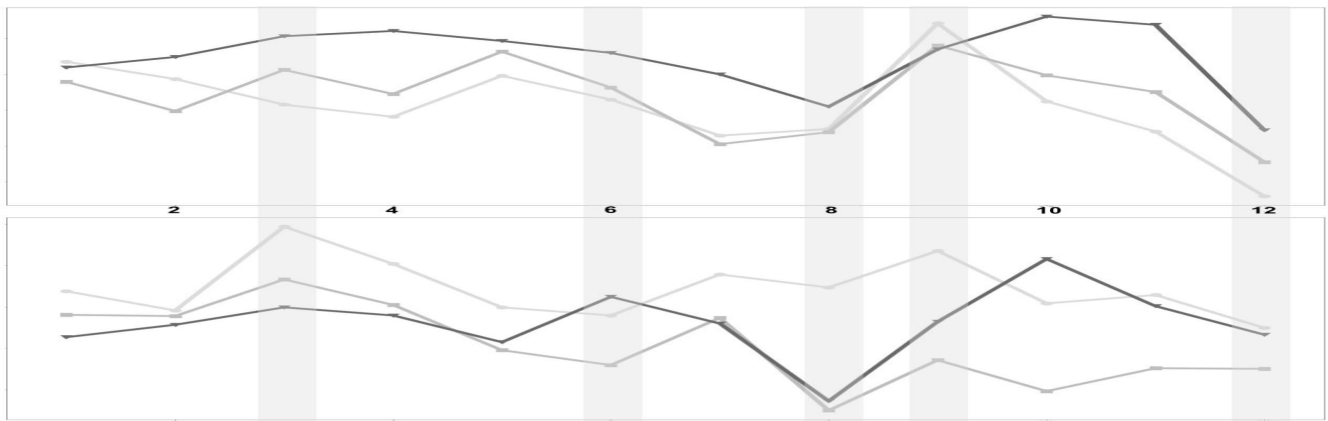


Figure 4. Monthly demand over the years of selected products. We compare the last three years of data.

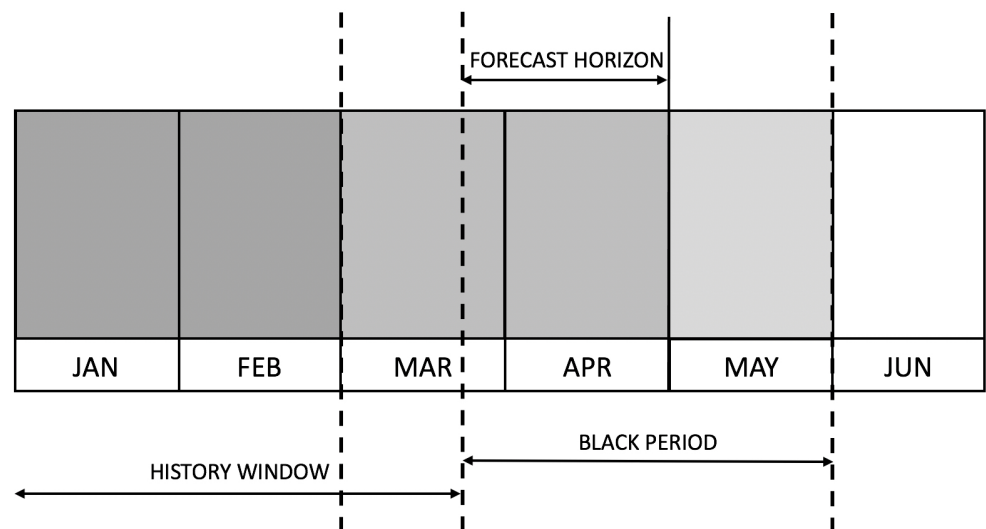


Figure 5. Relevant points in time we considered for forecasting purposes. There is a six-week slot between the moment we issue the forecast and the month we predict. The day of the month considered issuing the prediction is fixed.

For sales plans, we performed data fusion, merging annual and quarterly plans into a single one. In each case, we considered black periods, dates on which each plan becomes available, and the granularity level at which those estimates are provided.

PMI values are informed by the Institute of Supply Chain Management at the beginning of each month, based on the previous month's survey results. For crude oil prices, we considered the ones provided by the World Bank on a monthly level. We considered the same source for worldwide unemployment rates and GDP values, published yearly and available from March onwards. Every year in early March, the International Organization of Motor Vehicle Manufacturers publishes statistics regarding yearly worldwide car sales, which we used as well. Statistics regarding vehicle production worldwide are incomplete, and thus we did not consider them in this research. For copper prices, we took the London Metal Exchange value for each weekday of the year. We then computed the average price over the last three months. Finally, we computed features based on price adjustments applied to final products based on copper price fluctuations.

All sources of data are merged into a single dataset and aggregated at a product and monthly level.

3.3.1. Feature Creation

We can characterize demand data as a time series per product, each of which may present some level, trend, and seasonality. Therefore, we require a proper assessment of these aspects to create good statistical and ML models. While statistical models use only data regarding past demand to predict the future, ML models can leverage a more extensive set of features. These features provide insights into different factors affecting product demand and are taken into account to make predictions. We created a total of 708 features, some of which we describe below.

We computed rolling summary statistics such as average, maximum, and minimum demand over the last three months to address the time series level. In addition, we computed the same features for a weighted average and minimum and maximum values regarding the target month's past demand.

Trends help us understand demand growth or contraction and must be considered in forecasting models. However, different approaches may be helpful for statistical and ML models. The statistical models assume that the forecasted time series is stationary. To fulfill this assumption, we applied first-level differencing, which is suitable to address stochastic trends. For ML models, we created features to describe trends and capture monthly or interannual growth or contraction of GDP, unemployment rates, car sales, crude oil, copper prices, and demand. We also created features such as relations between observed demand and sales plans to capture common distortions that may take place on sales plans. Finally, we created derivative features that indicate growth or contraction for given months and more extended periods and detect and tag time series peaks from trend data.

We assessed seasonality using correlograms (see Figure 3). In addition, we incorporated demand at the lag values described before as proxies of potential demand.

We created naïve features to capture time characteristics to represent the month, a quarter, and workdays for given months. We used information regarding weekdays, national and collective holidays to compute the average demand per workday.

Values such as lagged demands, average, maximum, or minimum value of the last n observed months, and values from sales plans for target month, the weighted average of past demands for target month, and given product could act as reasonable approximations of demand values. Average demand per workday can be used to project expected demand on the target month, multiplying it by the number of workdays in the manufacturing plants. These demand approximations can be further adjusted using trend information.

We created two features to signal demand event occurrence: one based on product data sales plan, and the second one considering values from a probability density function on lagged demand values.

When building a single model for multiple products, it can be helpful to have some features convey information regarding demand similarity. Among others, we provide one-hot encoded features indicating demand-type as described in [12], considering demand behavior ever since we have data about the product and the last twelve and six months of the point in time we consider. To identify a similar context in which demand occurs, we binned GDP, unemployment rate, crude oil prices, car sales, and demand data into four bins of equal length for each case. Such features may also help identify specific cases, such as *year 4*, when context differs from most observed history.

3.3.2. Feature Selection

Feature selection reduces the number of features used to build a model, producing a succinct one that is quick to train, analyze, and understand.

We performed feature selection by combining the manual addition of common sense features with those suggested by a Gradient Boosted Regression Trees (GBRT) model, which is not sensitive to data distribution and allows us to rank features based on how much they reduce variance concerning target values. We only use the data that is later used to train the models and ensure that the test data remains unseen. We performed feature selection to extract the most relevant features in all experiments, considering all products

rather than for each product separately. In every case, we selected K features, obtaining K from $\sqrt{N}$, where N is the number of instances in the training subset, as suggested in [54], and empirically verified these features within our setting. Some of the best features are presented in Table 3.

Table 3. Top 15 features selected by the GBRT model considering the last three years of data. We did not remove correlated features in this case.

Feature	Brief Description
wdp_{3m}	Estimate of target demand based on average demand per working day on third month before predicted month, and amount of working days on target month.
$sp \cdot \frac{demand_{pastwavg}}{sp_{pastwavg}}$	Planned sales for target month adjusted with ratio of weighted averages of past demand and past planned sales for given month.
$demand_{lag4m} \cdot \frac{UE_{3m}}{UE_{15m}}$	Lagged demand (4 months before target month), adjusted by the ratio of unemployment rates three, and fifteen months before the month we aim to predict.
sp_{lag12m}	Planned sales for last year, same month we aim to predict.
$sp \cdot \frac{UE_{3m}}{UE_{15m}}$	Planned sales, adjusted by the ratio of unemployment rates three and fifteen months before the month we aim to predict.
sp	Planned sales for target month
$demand_{lag3m} \cdot \frac{GDP_{3m}}{GDP_{15m}}$	Lagged demand (3 months before target month), adjusted by the ratio of GDP three and fifteen months before the month we aim to predict.
$sp \cdot \frac{GDP_{3m}}{GDP_{15m}}$	Planned sales for target month, adjusted by the ratio of GDP three and fifteen months before the month we aim to predict.
$demand_{lag3m}$	Lagged demand (3 months before target month)
$wdp_{12} \cdot \frac{sp}{sp_{pastwavg}}$	Estimate of target demand based on average demand per working day a year before the predicted month and amount of working days on target month. Adjusted by the the ratio between planned sales for target month and the weighted average of planned sales for the same month over past years.
wdp_{8m}	Estimate of target demand based on average demand per working day on eighth month before predicted month and amount of working days on target month.
$wdp_{5m} \cdot \frac{UE_{3m}}{UE_{15m}}$	An estimate of target demand based on average demand per working day on the fifth month before predicted month and amount of working days on target month. Adjusted by the ratio of unemployment rates three and fifteen months before the month we aim to predict.
$wdp_{12m} \cdot \frac{PMI_{13m}}{PMI_{14m}}$	An estimate of target demand based on average demand per working day a year before predicted month and the amount of working days on target month. Adjusted by the ratio between PMI values 13, and 14 months before the target month.
$demand_{lag3m_{scaled}}$	Lagged demand (3 months before target month) - scaled between 0–1, considering products past demand values.
$wdp_{3m} \cdot \frac{GDP_{3m}}{GDP_{15m}}$	Estimate of target demand based on average demand per working day on third month before predicted month, and amount of working days on target month. Adjusted by the ratio of GDP three and fifteen months before the month we aim to predict.

3.4. Modeling

3.4.1. Feature Analysis and Prediction Techniques

ML algorithms may have different requirements regarding data preprocessing in order to ensure the best learning conditions. We thus analyzed data distributions and identified which steps were required in each case to satisfy those requirements.

For ML algorithms, we standardized (3) the features so that they would have zero mean and unit variance Equation (3), except for the case of the Multiple Linear Perceptron Regressor (MLPR), where we scaled the values of the features between zero and one Equation (4). Standardization enhances the model's numerical stability, makes some

algorithms consider all features equally important, and shortens ML models' training times [55].

$$x_{scaled} = \frac{X - \bar{X}}{\sigma} \quad (3)$$

$$x_{scaled} = \frac{X - \min(X)}{\max(X) - \min(X)} \quad (4)$$

We took into account 21 forecasting techniques. We considered the naïve forecast (last observed value as prediction) as the baseline method. We train twelve different batch ML models: on top of MLR, support vector regressor (SVR) [56], and multilayer perceptron regressor (MLPR) [57], which we found were used in automotive demand forecasting literature, we also evaluate Ridge [58]; Lasso [59]; Elastic Net [60]; K-nearest-neighbor regressor (KNNR) [61]; tree-based regressors (decision tree regressor (DTR), random forest regressor (RFR), and GBRT); a voting ensemble created using the most promising and diverse algorithms (KNNR, SVR, and RFR); as well as a stacked regression [62] considering KNNR, SVR, and RidgeCV as underlying estimators; and a GBRT model as final regressor. We also take into account four streaming ML algorithms: Adaptive Random Forest Regressor (ARFR) [63], Hoeffding Tree Regressor (HTR) [64], and Hoeffding Adaptive Tree Regressor (HATR) [65]. Additionally, we also consider forecasts obtained as the average demand for the last three months (MA(3)) and the ones obtained from statistical forecasting methods (exponential smoothing, random walk, ARIMA(1,1,0), and ARIMA(2,1,0)). We did not create deep learning models since we consider that not enough data was available to train them.

When training the models, we used MSE as the loss function where possible. We choose MSE because it has the desired property of penalizing higher errors more, thus reducing substantial discrepancies in predicted values.

3.5. Evaluation

From the literature review in Section 2 we observed that authors mostly used ME, MSE, RMSE, CAPE, MAPE, and R^2 metrics to measure the performance of the demand forecasting models related to the automotive industry. While ME, MSE, and RMSE are widely adopted, they all depend on the magnitude of the predicted and observed demands and thus cannot be used to compare groups of products with a different demand magnitude. This issue can be overcome with MAPE or CAPE metrics, though MAPE puts a heavier penalty on negative errors, preferring low forecasts—an undesired property in demand forecasting. Though R^2 is magnitude agnostic, it has been noticed that its value can increase when new features are added to the model [66].

To evaluate the performance of our models, we consider two metrics: MASE and R^2_{adj} . MASE informs the ratio between the MAE of the forecast values against the MAE of the naïve forecast, is magnitude agnostic, and not prone to distortions. R^2_{adj} informs how well predictions adjust to target values. In addition, it weights the number of features used to make the prediction, preferring succinct models that use fewer features for the same forecasting performance.

We compute an uncertainty range for each forecast, which illustrates possible bounds in which future demand values may be found. We also perform the Wilcoxon paired rank test [14] to assess if forecasts of a given model are significantly better than others.

Summary metrics may not be enough to understand the goodness of fit of a particular model [67,68]. Therefore, based on experts' opinions described in Section 3, and available demand characterizations, we analyzed the proportion of products with forecasting errors below certain thresholds (5%, 10%, 20%, and 30%), and the proportion of forecasts that resulted in under-estimates.

Though some research highlighted the importance of measuring forecast utility related to inventory performance (see, e.g., in [69,70]), this remains out of the scope of this work.

4. Experiments and Results

In this section, we describe the experiments we conducted (summarized in Table 4) and assess their results with metrics and criteria we described in Section 3.5. We summarize the outcomes in Table 5, to understand if a particular model performs significantly better than others. To evaluate the models, we used nested cross-validation [71], which is frequently used to evaluate time series models. To ensure conditions on ML streaming models were comparable to ML batch models, we implemented the nested cross-validation evaluation strategy. By doing so, we ensured the streaming model did not see new events until the required month was predicted. In order to test the models, we set apart the last six months of data. We published the nested cross-validation implementation for streaming models in the following repository: <https://github.com/JozefStefanInstitute/scikit-multiflow> (accessed date 21 July 2021).

Table 4. Description of experiments performed. Regarding the feature selection procedure, we consider two cases: (I) top features ranked by a GBRT model and curated by a researcher, and (II) top features ranked by a GBRT model, removing those with strong collinearity, curated by a researcher as well. N in the “Number of features” column refers to the number of instances in a given dataset.

Years of Data	Experiment	Feature Selection	Number of Features
All years available	Experiment 1	I	6
	Experiment 2	II	6
Last three years	Experiment 3	I	6
	Experiment 4	II	6
	Experiment 5	II	6
	Experiment 6	II	6
	Experiment 7	II	$\sqrt{N}$
	Experiment 8	II	$\sqrt{N}$
	Experiment 9	II	$\sqrt{N}$
	Experiment 10	II	$\sqrt{N}$
	Experiment 11	Only past demand	1

Table 5. Median of results obtained for each ML experiment. We abbreviate *under-estimates* as UE. In Experiments 9–10, streaming models based on Hoeffding bound show poor performance, resulting in negative R^2_{adj} values. We highlight the best results in bold.

Experiment	R^2_{adj}	MASE	5% Error	10% Error	20% error	30% Error	UE	90%+ Error
Experiment 1	0.8584	1.1450	0.0670	0.1086	0.2039	0.3051	0.3854	0.4077
Experiment 2	0.8447	1.1450	0.0655	0.1101	0.1920	0.2887	0.4182	0.3928
Experiment 3	0.9067	0.9150	0.0655	0.1280	0.2351	0.3095	0.4256	0.3928
Experiment 4	0.8998	0.9750	0.0655	0.1176	0.2143	0.3051	0.4152	0.4018
Experiment 5	0.8757	0.3900	0.0536	0.1116	0.2173	0.3140	0.4762	0.3497
Experiment 6	0.8679	0.3350	0.0565	0.1012	0.1875	0.2768	0.4851	0.3601
Experiment 7	0.8903	0.3550	0.0521	0.1131	0.2247	0.3155	0.4851	0.3408
Experiment 8	0.8786	0.3100	0.0506	0.0938	0.1890	0.2813	0.4658	0.3497
Experiment 9	−0.1611	0.8100	0.0357	0.0714	0.1428	0.2143	0.7321	0.3601
Experiment 10	−1.5344	0.5300	0.0178	0.0536	0.1250	0.2143	0.7143	0.4613

In Experiments 1–4, we assessed how events in year 4 affected model learning and if they significantly degraded forecasts. We also compared two different sets of features, resulting from two different procedures to obtain them. We obtained the best performance with local models trained over the last three years of data. Removing features with high collinearity did not enhance the median of R^2_{adj} and MASE. Therefore, we consider Experiment 3 performed best, having the best MASE and R^2_{adj} values. In contrast, the rest of the evaluation criteria values were acceptable.

Next, we analyzed if grouping products by specific criteria would enhance the quality of the predictions. We trained these global models over the last three years of data, considering insights obtained from Experiments 1–4. Following the *ceteris paribus* principle, we considered the same features as for Experiment 3. We experimented with grouping products based on the median magnitude of past demand (Experiment 5) and demand-type (Experiment 6). We observed that even though the median of R^2_{adj} was lower, and the under-estimates ratio higher, compared to results in previous experiments, the median MASE values decreased by more than 40%. Models based on the median of past demand had the best results in most aspects, including the proportion of forecasts with more than 90% error. Encouraged by these results, we conducted Experiments 7–8, preserving the grouping criteria but adapting the number of features considered according to the amount of data available in each sub-group. In Experiment 7, we grouped them based on the magnitude of the median of past demand. In contrast, in Experiment 8, we grouped products based on demand type. In both cases, we observed that R^2_{adj} values and under-estimates ratios improved, and MASE values remained low. We consider the best results were obtained in Experiment 7, which achieved the best values in all evaluation criteria, except for MASE. We ranked models of these two experiments by R^2_{adj} , and took the top three. We obtained SVR, voting, and stacking models for Experiment 7 and SVR, voting, and RFR models for Experiment 8. The models from Experiment 8 exhibited lower MASE in all cases, a better ratio of under-estimates, and a better proportion of forecasts with an error ratio higher than 90%. Top 3 models from Experiment 8 remained competitive regarding R^2_{adj} and proportion of forecasts with error ratio bounded to 30% or less error.

We assessed the statistical significance of both groups' models in all the performance aspects mentioned above, at a p -value = 0.05. The models had no significant difference in the same group regarding R^2_{adj} and MASE. However, the difference was significant between voting models in both groups for these two metrics. The difference was also significant between the voting model from Experiment 7 and the RFR model from Experiment 8 for the MASE metric. Considering the proportion of forecasts with errors lower than 30%, we observed no differences between both groups' models. However, differences between SVR and voting models in Experiment 8 were significant. Finally, differences regarding the number of under-estimates were statistically significant between all top three models from Experiment 7 against SVR and RFR models of Experiment 8. For this particular performance aspect, the stacking model from Experiment 7 only achieves significance against the voting model from Experiment 8.

Having explored a wide range of batch ML models, we conducted Experiments 9–10 with streaming ML models, following the same conditions as Experiment 7–8, but creating a global streaming model for each magnitude of the median of past demand demand-type. This experiment aimed to understand the performance of streaming ML models against the widely used ML batch models and confirm if they behaved the same regarding error bounds as models in Experiments 7–8. We found that streaming models based on Hoeffding inequality did not learn well. On the other side, the Adaptive Random Forest Regressor displayed a better performance. While its R^2_{adj} was lower than the top 3 models from Experiment 8, it achieved the best MASE in Experiment 10. It also had among best proportion predictions with less than 5%, 10%, 20%, and 30% error or more than 90% error. However, the proportion of under-estimates, a parameter of crucial importance in our use case, hindered these performance results. ML streaming models had among the highest proportions of under-estimates of all created forecasting models. The highest proportion of under-estimates was obtained in ML streaming models based on the Hoeffding inequality, reaching a median of underestimates above 70%.

In Experiments 5–10, we consistently observed global models created considering the magnitude of the median of past demand outperformed those created based on demand-type when considering the proportion of forecasts with an error higher to 90%. On the other side, global models based on demand-type scored better on MASE. However, these

differences did not prove statistically significant in most cases when comparing top-ranking models of both groups.

Having explored different ML models, we then trained statistical models for each product considering demand data available for the last three years (Experiment 11) and contrasted results obtained with the top three models from Experiment 8 (see Table 6). When preparing demand data for the statistical models, we applied differencing to remove stochastic trends. We observed that the ML models outperformed the statistical ones in almost every aspect. R^2_{adj} was consistently low for statistical models, and though their MASE was better compared to the baseline models, ML models performed better. When assessing the ratio of forecasts with less than 30% error, ML models displayed a better performance. We observed the same when analyzing the under-estimates ratio. Even though the random walk had a low under-estimates ratio, the rest of the metrics indicate the random walk model provides poor forecasts. We consider the best overall performers are the SVR, RFR, and GBRT models, which achieved near-human performance in almost every aspect considered in this research. Even though differences regarding R^2_{adj} , MASE, and the ratio of forecasts with less than 30% error are not statistically significant between them in most cases, they display statistically significant differences when analyzing under-estimates.

Table 6. Results we obtained for the top 3 performing models from Experiment 8 (ML batch models), best result for experiments 9–10 (ML streaming models), and baseline and statistical models. We abbreviate *under-estimates* as *UE*.

Algorithm Type	Algorithm	R^2_{adj}	MASE	5% Error	10% Error	20% Error	30% Error	UE	90+% Error
ML batch	SVR	0.9212	0.2600	0.0774	0.1101	0.2321	0.3333	0.4077	0.3304
	Voting	0.9059	0.2800	0.0625	0.0923	0.1786	0.2798	0.4792	0.3393
	RFR	0.8953	0.2900	0.0417	0.1012	0.2173	0.3244	0.3423	0.3482
ML streaming	ARFR (Experiment 9)	0.8728	0.3300	0.0744	0.1339	0.2500	0.3274	0.5387	0.3452
	ARFR (Experiment 10)	0.8205	0.2200	0.0744	0.1280	0.2232	0.3274	0.5268	0.3423
Baseline	MA(3)	0.8938	0.8800	0.1190	0.1667	0.2530	0.3482	0.3571	0.3065
	Naïve	0.8519	1.0000	0.2024	0.2411	0.3423	0.4137	0.4137	0.3214
Statistical	ARIMA(2.1.0)	0.3846	0.4500	0.0476	0.0774	0.1429	0.1875	0.5536	0.5208
	Exponential smoothing	0.3258	0.3600	0.0506	0.1161	0.1905	0.2738	0.5923	0.4434
	ARIMA(1.1.0)	0.2840	0.5200	0.0387	0.0744	0.1012	0.1726	0.5119	0.6071
	Random walk	−0.6705	0.9000	0.0327	0.0387	0.0655	0.0923	0.3780	0.7678

5. Conclusions

This research compares 21 forecasting techniques (baseline, statistical, and ML algorithms) to provide future demand estimates for an automotive OEM company located in Europe. We use various internal and external data sources that describe the economic context and provide insights on future demand. We considered multiple metrics and criteria to assess forecasting models' performance (R^2_{adj} , MASE, the ratio of forecasts with less than 30% error, and the ratio of forecasts with under-estimates)—all of them magnitude-agnostic. These metrics and criteria allow us to characterize results to be comparable regardless of the underlying data. We also assess the statistical significance of results, something we missed in most related literature.

The obtained results show that grouping products according to their demand patterns or past demand magnitude enhances the performance of ML models. We observed that the best MASE performance was obtained on models created for a group of products with the same demand type. Furthermore, when training global models based on the median of past demand, models usually achieved a better R^2_{adj} and a better bound on high forecast errors. However, these values were not always statistically significant.

Our experimental evaluation indicates that the best performing models are SVR, voting ensemble, and RFR trained over product data of the same demand type. The SVR and RFR models achieved near-human performance for the ratio of forecasts under 30% error, and the RFR model scored close to human performance regarding under-estimates. However, none of the models achieved close to human performance on the proportion of forecasts with a high error (more than 90%). How to efficiently detect and bound such cases remains a subject of future research.

When comparing batch and streaming ML models' performance, we observed that ML batch models displayed a more robust performance. From the streaming algorithms, the ARFR achieved competitive results, except for a high ratio of under-estimates. This critical aspect must not be overlooked. Models based on the Hoeffding inequality did not learn well and had poor performance, and further research is required to understand the reasons hindering these models' learning process.

Building a single demand forecasting model for multiple products not only drives better performance, but has engineering implications: fewer models need to be trained and deployed into production. The need for regular deployments can be further reduced by using ML streaming models. This advantage gains importance when considering ever shorter forecasting horizons as it avoids the overhead regular model re-trainings and model deployments. We consider timely access to real data and the ability to regularly update machine learning models as factors that enable digital twins' creation. Such digital twins not only provide accurate forecasts but allow estimating different what-if scenarios of interest.

We envision at least two directions for future work. First, further research is required to develop effective error bounding strategies for demand forecasts. We want to explore the usage of ML anomaly detection methods to identify anomalous forecasts issued by global models and develop strategies to address such anomalies. Second, research is required to provide explanations that inform the context considered by the ML model and models' forecasted values and uncertainty. We understand that accurate forecasts are a precondition to building users' trust in a demand forecasting software. Nevertheless, accurate forecasts alone are not enough. ML models explainability is required to help the user understand the reasons behind a forecast, decide if it can be trusted, and gain more profound domain knowledge.

Author Contributions: Conceptualization, J.M.R., B.K., M.Š., B.F. and D.M.; methodology, J.M.R., B.K., M.Š., B.F. and D.M.; software, J.M.R. and B.K.; validation, J.R., B.K., M.Š. and B.F.; formal analysis, J.M.R., B.K., M.Š. and B.F.; investigation, J.M.R., M.Š. and B.F.; resources, J.M.R., B.K., M.Š. and B.F.; data curation, J.M.R., B.K., M.Š. and B.F.; writing—original draft preparation, J.M.R.; writing—review and editing, J.M.R., M.Š., B.F. and D.M.; visualization, J.M.R.; supervision, B.F. and D.M.; project administration, M.Š., B.F. and D.M.; funding acquisition, B.F. and D.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Slovenian Research Agency and European Union's Horizon 2020 program project FACTLOG under grant agreement number H2020-869951.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ADI	Average Demand Interval
ANFIS	Adaptive Network-based Fuzzy Inference System
ANN	Artificial Neural Network

ARFR	Adaptive Random Forest Regressor
ARIMA	autoregressive integrated moving average model
ARMA	Autoregressive Moving Average
CAPE	Cumulative Absolute Percentage Errors
CRISP-DM	CRoss-Industry Standard Process for Data Mining
CV	Coefficient of Variation
DTR	Decision Tree Regressor
GBTR	Gradient Boosted Regression Trees
GDP	Gross Domestic Product
HATR	Hoeffding Adaptive Tree Regressor
HTR	Hoeffding Tree Regressor
KNNR	K-Nearest-Neighbor Regressor
MA	Moving Average
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MASE	Mean Absolute Scaled Error
ME	Mean Error
ML	Machine Learning
MLPR	Multiple Linear Perceptron Regressor
MLR	Multiple Linear Regression
MSE	Mean Squared Error
OEM	Original Equipment Manufacturer
PMI	Purchasing Managers' Index
R ²	Coefficient of determination
R ² adj	Coefficient of determination - adjusted
RFR	Random Forest Regressor
RMSE	Root Mean Squared Error
SBA	Syntetos–Boylan Approximation
SVM	Support Vector Machine
SVR	Support Vector Regressor
UE	Under-estimates
VAR	Vector Autoregression
VECM	Vector Error Correction Model

References

- Cambridge University Press. *Cambridge Learner's Dictionary with CD-ROM*; Cambridge University Press: Cambridge, UK, 2007.
- Wei, W.; Guimarães, L.; Amorim, P.; Almada-Lobo, B. Tactical production and distribution planning with dependency issues on the production process. *Omega* **2017**, *67*, 99–114. [\[CrossRef\]](#)
- Lee, H.L.; Padmanabhan, V.; Whang, S. The bullwhip effect in supply chains. *Sloan Manag. Rev.* **1997**, *38*, 93–102. [\[CrossRef\]](#)
- Bhattacharya, R.; Bandyopadhyay, S. A review of the causes of bullwhip effect in a supply chain. *Int. J. Adv. Manuf. Technol.* **2011**, *54*, 1245–1261. [\[CrossRef\]](#)
- Brühl, B.; Hülsmann, M.; Borscheid, D.; Friedrich, C.M.; Reith, D. A sales forecast model for the german automobile market based on time series analysis and data mining methods. In *Industrial Conference on Data Mining*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 146–160.
- de Almeida, M.M.K.; Marins, F.A.S.; Salgado, A.M.P.; Santos, F.C.A.; da Silva, S.L. Mitigation of the bullwhip effect considering trust and collaboration in supply chain management: a literature review. *Int. J. Adv. Manuf. Technol.* **2015**, *77*, 495–513. [\[CrossRef\]](#)
- Dwaikat, N.Y.; Money, A.H.; Behashti, H.M.; Salehi-Sangari, E. How does information sharing affect first-tier suppliers' flexibility? Evidence from the automotive industry in Sweden. *Prod. Plan. Control.* **2018**, *29*, 289–300. [\[CrossRef\]](#)
- Martinsson, T.; Sjöqvist, E. Causes and Effects of Poor Demand Forecast Accuracy A Case Study in the Swedish Automotive Industry. Master's Thesis, Chalmers University of Technology/Department of Technology Management and Economics, Gothenburg, Sweden, 2019.
- Ramanathan, U.; Ramanathan, R. *Sustainable Supply Chains: Strategies, Issues, and Models*; Springer: New York, NY, USA, 2020.
- Gutierrez, R.S.; Solis, A.O.; Mukhopadhyay, S. Lumpy demand forecasting using neural networks. *Int. J. Prod. Econ.* **2008**, *111*, 409–420. [\[CrossRef\]](#)
- Lolli, F.; Gamberini, R.; Regattieri, A.; Balugani, E.; Gatos, T.; Gucci, S. Single-hidden layer neural networks for forecasting intermittent demand. *Int. J. Prod. Econ.* **2017**, *183*, 116–128. [\[CrossRef\]](#)
- Syntetos, A.A.; Boylan, J.E.; Croston, J. On the categorization of demand patterns. *J. Oper. Res. Soc.* **2005**, *56*, 495–503. [\[CrossRef\]](#)
- Hyndman, R.J.; others. Another look at forecast-accuracy metrics for intermittent demand. *Foresight Int. J. Appl. Forecast.* **2006**, *4*, 43–46.

14. Wilcoxon, F. Individual comparisons by ranking methods. In *Breakthroughs in Statistics*; Springer: New York, NY, USA, 1992; pp. 196–202.
15. Williams, T. Stock control with sporadic and slow-moving demand. *J. Oper. Res. Soc.* **1984**, *35*, 939–948. [\[CrossRef\]](#)
16. Johnston, F.; Boylan, J.E. Forecasting for items with intermittent demand. *J. Oper. Res. Soc.* **1996**, *47*, 113–121. [\[CrossRef\]](#)
17. Dargay, J.; Gately, D. Income's effect on car and vehicle ownership, worldwide: 1960–2015. *Transp. Res. Part Policy Pract.* **1999**, *33*, 101–138. [\[CrossRef\]](#)
18. Wang, F.K.; Chang, K.K.; Tzeng, C.W. Using adaptive network-based fuzzy inference system to forecast automobile sales. *Expert Syst. Appl.* **2011**, *38*, 10587–10593. [\[CrossRef\]](#)
19. Vahabi, A.; Hosseininia, S.S.; Alborzi, M. A Sales Forecasting Model in Automotive Industry using Adaptive Neuro-Fuzzy Inference System (Anfis) and Genetic Algorithm (GA). *Management* **2016**, *1*, 2. [\[CrossRef\]](#)
20. Ubaidillah, N.Z. A study of car demand and its interdependency in sarawak. *Int. J. Bus. Soc.* **2020**, *21*, 997–1011. [\[CrossRef\]](#)
21. Sharma, R.; Sinha, A.K. Sales forecast of an automobile industry. *Int. J. Comput. Appl.* **2012**, *53*, 25–28. [\[CrossRef\]](#)
22. Gao, J.; Xie, Y.; Cui, X.; Yu, H.; Gu, F. Chinese automobile sales forecasting using economic indicators and typical domestic brand automobile sales data: A method based on econometric model. *Adv. Mech. Eng.* **2018**, *10*, 1687814017749325. [\[CrossRef\]](#)
23. Kwan, H.W. On the Demand Distributions of Slow-Moving Items. Ph.D. Thesis, University of Lancaster, Lancaster, UK, 1991
24. Eaves, A.H.C. Forecasting for the Ordering and Stock-Holding of Consumable Spare Parts. Ph.D. Thesis, Lancaster University, Lancaster, UK, 2002
25. Syntetos, A.A.; Babai, M.Z.; Altay, N. On the demand distributions of spare parts. *Int. J. Prod. Res.* **2012**, *50*, 2101–2117. [\[CrossRef\]](#)
26. Lengu, D.; Syntetos, A.A.; Babai, M.Z. Spare parts management: Linking distributional assumptions to demand classification. *Eur. J. Oper. Res.* **2014**, *235*, 624–635. [\[CrossRef\]](#)
27. Dwivedi, A.; Niranjana, M.; Sahu, K. A business intelligence technique for forecasting the automobile sales using Adaptive Intelligent Systems (ANFIS and ANN). *Int. J. Comput. Appl.* **2013**, *74*, 975–8887. [\[CrossRef\]](#)
28. Matsumoto, M.; Komatsu, S. Demand forecasting for production planning in remanufacturing. *Int. J. Adv. Manuf. Technol.* **2015**, *79*, 161–175. [\[CrossRef\]](#)
29. Farahani, D.S.; Momeni, M.; Amiri, N.S. Car sales forecasting using artificial neural networks and analytical hierarchy process. In Proceedings of the Fifth International Conference on Data Analytics: DATA ANALYTICS 2016, Venice, Italy, 9–13 October 2016; p. 69
30. Henkelmann, R. A Deep Learning based Approach for Automotive Spare Part Demand Forecasting. Master Thesis, Otto von Guericke Universität Magdeburg, Magdeburg, Germany, 2018
31. Chandriah, K.K.; Naraganahalli, R.V. RNN/LSTM with modified Adam optimizer in deep learning approach for automobile spare parts demand forecasting. *Multimed. Tools Appl.* **2021**, 1–15. [\[CrossRef\]](#)
32. Hanggara, F.D. Forecasting Car Demand in Indonesia with Moving Average Method. *J. Eng. Sci. Technol. Manag.* **2021**, *1*, 1–6.
33. Zhang, G.P.; Qi, M. Neural network forecasting for seasonal and trend time series. *Eur. J. Oper. Res.* **2005**, *160*, 501–514. [\[CrossRef\]](#)
34. Athanasopoulos, G.; Hyndman, R.J.; Song, H.; Wu, D.C. The tourism forecasting competition. *Int. J. Forecast.* **2011**, *27*, 822–844. [\[CrossRef\]](#)
35. Montero-Manso, P.; Hyndman, R.J. Principles and algorithms for forecasting groups of time series: Locality and globality. *arXiv* **2020**, arXiv:2008.00444.
36. Salinas, D.; Flunkert, V.; Gasthaus, J.; Januschowski, T. DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *Int. J. Forecast.* **2020**, *36*, 1181–1191. [\[CrossRef\]](#)
37. Bandara, K.; Bergmeir, C.; Smyl, S. Forecasting across time series databases using recurrent neural networks on groups of similar series: A clustering approach. *Expert Syst. Appl.* **2020**, *140*, 112896. [\[CrossRef\]](#)
38. Laptev, N.; Yosinski, J.; Li, L.E.; Smyl, S. Time-series extreme event forecasting with neural networks at uber. In Proceedings of the International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; Volume 34, pp. 1–5
39. Wirth, R.; Hipp, J. CRISP-DM: Towards a standard process model for data mining. In *Proceedings of the 4th International Conference on the Practical Applications of Knowledge Discovery and Data Mining*; Springer: London, UK, 2000; pp. 29–39.
40. Wang, C.N.; Tibo, H.; Nguyen, H.A. Malmquist productivity analysis of top global automobile manufacturers. *Mathematics* **2020**, *8*, 580. [\[CrossRef\]](#)
41. Tubaro, P.; Casilli, A.A. Micro-work, artificial intelligence and the automotive industry. *J. Ind. Bus. Econ.* **2019**, *46*, 333–345. [\[CrossRef\]](#)
42. Ryu, H.; Basu, M.; Saito, O. What and how are we sharing? A systematic review of the sharing paradigm and practices. *Sustain. Sci.* **2019**, *14*, 515–527. [\[CrossRef\]](#)
43. Li, M.; Zeng, Z.; Wang, Y. An innovative car sharing technological paradigm towards sustainable mobility. *J. Clean. Prod.* **2021**, *288*, 125626. [\[CrossRef\]](#)
44. Svennevik, E.M.; Julsrud, T.E.; Farstad, E. From novelty to normality: reproducing car-sharing practices in transitions to sustainable mobility. *Sustain. Sci. Pract. Policy* **2020**, *16*, 169–183. [\[CrossRef\]](#)
45. Heineke, K.; Möller, T.; Padhi, A.; Tschiesner, A. *The Automotive Revolution is Speeding Up*; McKinsey and Co.: New York, NY, USA, 2017.

46. Verevka, T.V.; Gutman, S.S.; Shmatko, A. Prospects for Innovative Development of World Automotive Market in Digital Economy. In Proceedings of the 2019 International SPBPU Scientific Conference on Innovations in Digital Economy, Saint Petersburg, Russian, 14–15 October 2019; pp. 1–6
47. Armstrong, J.S.; Morwitz, V.G.; Kumar, V. Sales forecasts for existing consumer products and services: Do purchase intentions contribute to accuracy? *Int. J. Forecast.* **2000**, *16*, 383–397. [[CrossRef](#)]
48. Morwitz, V.G.; Steckel, J.H.; Gupta, A. When do purchase intentions predict sales? *Int. J. Forecast.* **2007**, *23*, 347–364. [[CrossRef](#)]
49. Hotta, L.; Neto, J.C. The effect of aggregation on prediction in autoregressive integrated moving-average models. *J. Time Ser. Anal.* **1993**, *14*, 261–269. [[CrossRef](#)]
50. Souza, L.R.; Smith, J. Effects of temporal aggregation on estimates and forecasts of fractionally integrated processes: A Monte-Carlo study. *Int. J. Forecast.* **2004**, *20*, 487–502. [[CrossRef](#)]
51. Rostami-Tabar, B.; Babai, M.Z.; Syntetos, A.; Ducq, Y. Demand forecasting by temporal aggregation. *Nav. Res. Logist. (NRL)* **2013**, *60*, 479–498. [[CrossRef](#)]
52. Nikolopoulos, K.; Syntetos, A.A.; Boylan, J.E.; Petropoulos, F.; Assimakopoulos, V. An aggregate–disaggregate intermittent demand approach (ADIDA) to forecasting: an empirical proposition and analysis. *J. Oper. Res. Soc.* **2011**, *62*, 544–554. [[CrossRef](#)]
53. Syntetos, A.; Babai, M.; Altay, N. Modelling spare parts’ demand: An empirical investigation. In Proceedings of the 8th International Conference of Modeling and Simulation MOSIM, Hammamet, Tunisia, 10–12 May 2010; Citeseer: Forest Grove, OR, USA, 2010; Volume 10
54. Hua, J.; Xiong, Z.; Lowey, J.; Suh, E.; Dougherty, E.R. Optimal number of features as a function of sample size for various classification rules. *Bioinformatics* **2005**, *21*, 1509–1515. [[CrossRef](#)]
55. Varma, S.; Simon, R. Bias in error estimation when using cross-validation for model selection. *BMC Bioinform.* **2006**, *7*, 91. [[CrossRef](#)]
56. Drucker, H.; Burges, C.J.; Kaufman, L.; Smola, A.J.; Vapnik, V. Support vector regression machines. In Proceedings of the Advances in Neural Information Processing Systems, Denver, CO, USA, 2–5 December 1997; pp. 155–161
57. Rosenblatt, F. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychol. Rev.* **1958**, *65*, 386. [[CrossRef](#)]
58. Hoerl, A.E.; Kennard, R.W. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics* **1970**, *12*, 55–67. [[CrossRef](#)]
59. Tibshirani, R. Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B* **1996**, *58*, 267–288. [[CrossRef](#)]
60. Zou, H.; Hastie, T. Regularization and variable selection via the elastic net. *J. R. Stat. Soc. Ser. B* **2005**, *67*, 301–320. [[CrossRef](#)]
61. Altman, N.S. An introduction to kernel and nearest-neighbor nonparametric regression. *Am. Stat.* **1992**, *46*, 175–185.
62. Wolpert, D.H. Stacked generalization. *Neural Netw.* **1992**, *5*, 241–259. [[CrossRef](#)]
63. Gomes, H.M.; Barddal, J.P.; Ferreira, L.E.B.; Bifet, A. Adaptive random forests for data stream regression. In Proceedings of the European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN), Bruges, Belgium, 2–4 October 2018
64. Domingos, P.; Hulten, G. Mining high-speed data streams. In Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Boston, MA, USA, 20–23 August 2000; pp. 71–80
65. Bifet, A.; Gavaldà, R. Adaptive learning from evolving data streams. In *International Symposium on Intelligent Data Analysis*; Springer: New York, NY, USA, 2009; pp. 249–260
66. Ferligoj, A.; Kramberger, A. Some Properties of R^2 in Ordinary Least Squares Regression. 1995
67. Armstrong, J.S. Illusions in regression analysis. *Int. J. Forecast.* **2011**, *28*, 689–694. [[CrossRef](#)]
68. Tufte, E.R. *The Visual Display of Quantitative Information*; Graphics Press: Cheshire, CT, USA, 2001; Volume 2.
69. Ali, M.M.; Boylan, J.E.; Syntetos, A.A. Forecast errors and inventory performance under forecast information sharing. *Int. J. Forecast.* **2012**, *28*, 830–841. [[CrossRef](#)]
70. Bruzda, J. Demand forecasting under fill rate constraints—The case of re-order points. *Int. J. Forecast.* **2020**, *36*, 1342–1361. [[CrossRef](#)]
71. Stone, M. Cross-validatory choice and assessment of statistical predictions. *J. R. Stat. Soc. Ser. B* **1974**, *36*, 111–133. [[CrossRef](#)]

2.2 Reframing Demand Forecasting: a Two-Fold Approach for Lumpy and Intermittent Demand

Forecasting lumpy and intermittent demand is considered among the most complex demand forecasting challenges, given the distortions that no-demand intervals introduce to the machine learning models. To mitigate data sparseness, different approaches have been considered. When considering fixed forecasting time horizons, authors explored including features that could signal the absence of demand or consider higher data aggregation levels, ensuring some level of demand always exists. When no fixed forecasting time horizon is required, methods such as forecasting the next demand occurrence period and the expected demand were considered. Our research argues that the approaches considered so far always considered the demand forecasting problem a regression problem, a statement grounded in the evaluation metrics observed in the related work. By doing so in a fixed forecasting horizon problem, they could not accurately predict demand occurrence, leading to poor forecasting results. Furthermore, no satisfactory solution was provided to the underlying sparseness problem. Our paper proposes a two-fold machine learning model for a fixed forecasting horizon to forecast lumpy and intermittent demand. In particular, we propose a machine learning classifier to predict demand occurrence and a separate model to predict demand quantity. Both models can have separate sets of features and optimize for different goals. Furthermore, the regression models consider only data where demand occurred, ensuring no distortions occur due to the demand sparseness. Moreover, we propose a set of metrics to assess the different model objectives and overall forecast impact on inventory. To determine how accurate are the demand occurrence predictions, we evaluate the classifier models with the AUC ROC metric. Conversely, we use the MASE metric to determine how precise the regression models are. The MASE metric provides a magnitude-agnostic assessment of the model's regression quality and therefore enables comparing multiple models, regardless of the magnitude of the demand considered in each case. Finally, we compute the SPEC metric to assess the impact of the two-fold model forecasts on inventory. By considering separate metrics to evaluate how accurately the models predict demand occurrence, expected demand quantity, and forecast impact on inventory, we allow for accurate diagnoses of model forecasting failures. Such diagnoses can be used to refine the predictive models further, leading to improved forecasts.

Through various experiments, we show that popular and state-of-the-art demand forecasting models for lumpy and intermittent demand display poor performance by failing to predict demand occurrence correctly. Conversely, we succeed in predicting demand occurrence accurately, achieving state-of-the-art performance. Furthermore, we propose a set of metrics to evaluate forecasting models for lumpy and intermittent demand.

Article

Reframing Demand Forecasting: A Two-Fold Approach for Lumpy and Intermittent Demand

Jože Martin Rožanec ^{1,2,3,*} , Blaž Fortuna ^{1,2}  and Dunja Mladenec ¹ 

¹ Jožef Stefan Institute, Jamova 39, 1000 Ljubljana, Slovenia; blaz.fortuna@qlector.com (B.F.); dunja.mladenec@ijs.si (D.M.)

² Qlector d.o.o., Rovšnikova 7, 1000 Ljubljana, Slovenia

³ Jožef Stefan International Postgraduate School, Jamova 39, 1000 Ljubljana, Slovenia

* Correspondence: joze.rozanec@ijs.si

Abstract: Demand forecasting is a crucial component of demand management. While shortening the forecasting horizon allows for more recent data and less uncertainty, this frequently means lower data aggregation levels and a more significant data sparsity. Furthermore, sparse demand data usually result in lumpy or intermittent demand patterns with irregular demand intervals. The usual statistical and machine learning models fail to provide good forecasts in such scenarios. Our research confirms that competitive demand forecasts can be obtained through two models: predicting the demand occurrence and estimating the demand size. We analyze the usage of local and global machine learning models for both cases and compare the results against baseline methods. Finally, we propose a novel evaluation criterion for the performance of lumpy and intermittent demand forecasting models. Our research shows that global classification models are the best choice when predicting demand event occurrence. We achieved the best results using the simple exponential smoothing forecast to predict demand sizes. We tested our approach on real-world data made up of 516 time series corresponding to the daily demand, over three years, of a European original automotive equipment manufacturer.

Keywords: demand forecasting; lumpy demand; supply chain agility; smart responsive manufacturing; artificial intelligence; machine learning; industry 4.0; CatBoost



Citation: Rožanec, J.M.; Fortuna, B.; Mladenec, D. Reframing Demand Forecasting: A Two-Fold Approach for Lumpy and Intermittent Demand. *Sustainability* **2022**, *14*, 9295. <https://doi.org/10.3390/su14159295>

Academic Editor: Masood Fathi

Received: 7 June 2022

Accepted: 26 July 2022

Published: 29 July 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Demand forecasting is a critical component of supply chain management, directly affecting production planning and order fulfillment. Accurate forecasts have an impact across the whole supply chain and affect manufacturing plant organization: operational and strategic decisions are made regarding resources (the allocation and scheduling of raw material and tooling), workers (scheduling, training, promotions, and hiring), manufactured products (market share increase and production diversification), and logistics for deliveries.

To issue accurate forecasts, we have to consider demand characteristics (Kim et al. [1] and Moon et al. [2]). Multiple demand characterizations have been proposed (Williams [3] and Johnston et al. [4]), and one of the most influential is the characterization proposed by Syntetos et al. [5], which divides demand patterns into four quadrants based on the inter-demand interval and the coefficient of variation. The four demand types are smooth (regular demand occurrence and low demand quantity variation), erratic (regular demand occurrence and high demand quantity variation), intermittent (irregular demand occurrence and low demand quantity variation), and lumpy (irregular demand occurrence and high demand quantity variation). Intermittent and lumpy demand problems are considered among the most challenging demand forecasting problems (Amin-Naseri et al. [6] and Mukhopadhyay et al. [7]). Both present infrequent demand arrivals with many zero-demand periods; these factors pose an additional challenge to accurate demand quantity estimation. Demand quantity estimation is harder for lumpy demand since it also presents variable demand sizes (Petropoulos et al. [8] and Bartezzaghi et al. [9]). Nevertheless, intermittent

demand items account for considerable proportions of any organization's stock value (Babai et al. [10]). Furthermore, for a use case described by Johnston et al. [11], it was found that 75% of items had a lumpy demand and accounted for 40% of the company's revenue and 60% of stock investment. Along this line, Amin-Naseri et al. [6] cite multiple authors who observe that lumpy patterns are widespread, especially in organizations that hold many spare parts, such as the process, aviation, and automotive industries, as well as companies dealing with telecommunication systems, large compressors, and other examples.

Increasing industry automation, digitalization, and information sharing (e.g., through electronic data interchange software), fomented by national and regional initiatives (Davies [12], Glaser [13], and Yang et al. [14]), accelerates the data and information flow within an organization, enabling greater agility. Therefore, it is critical to develop demand forecasting models capable of providing forecasts at a low granularity level to achieve greater agility in supply chain management. Such models enable short forecasting horizons and provide insights at a significant level of detail, allowing organizations to foresee and react to changes quickly. However, while these forecasting models benefit from the most recent data available (which helps enhance the forecast's accuracy), the low granularity level frequently requires dealing with irregular demand patterns (Syntetos et al. [15]).

Given the variety of demand types, researchers have proposed multiple approaches to providing accurate demand forecasts. While smooth and erratic demand patterns achieve good results using regression models, intermittent and lumpy demand require specialized models that consider demand occurrence. Statistical, machine learning, and hybrid models have been developed to that end. The increase in industry digitalization enables the timely collection of data relevant to demand forecasts. Data availability is critical for developing machine learning models, which sometimes achieve the best results.

To deal with intermittent demand, Croston [16] proposed a forecasting model that provides separate estimates for demand occurrence and demand quantity. Since then, much work has followed this direction. The measurement of intermittent and lumpy model performance has also been the subject of extensive research. Many authors agree that we require regression accuracy and inventory metrics. Furthermore, there is increasing agreement that regression metrics alone, used for smooth and erratic demand, do not help measure intermittent and lumpy demand since they fail to weigh zero-demand periods. Inventory metrics suffer the same bias while providing a perspective on how much time products stay in stock.

Croston's intuition in separating demand occurrence from demand sizes was valuable. While many authors followed this intuition, we found that in the reviewed literature, no author fully considered demand forecasting as a compound problem that required separate models and metrics. We propose reframing demand forecasting as a two-phase problem that requires (i) a classification model to predict demand occurrence and (ii) a regression model to predict demand sizes. Classification can be omitted for smooth and erratic demand patterns since demand (almost) always occurs. In those cases, using only a regression model provides good demand forecasts (Brühl et al. [17], Wang et al. [18], Sharma et al. [19], Gao et al. [20], Salinas et al. [21], and Bandara et al. [22]). For intermittent and erratic demand patterns, using separate models for classification and regression provides at least two benefits. First, separate models allow optimization for different objectives. Second, each problem has adequate metrics, and the cause of performance or under-performance can be clearly understood and addressed.

In this research, we propose:

1. Decoupling the demand forecasting problem into two separate problems: classification (demand occurrence) and regression (demand quantity estimation);
2. Using four measurements to assess demand forecast performance: (i) the area under the receiver operating characteristic curve (AUC ROC) (Bradley [23]) to assess demand occurrence, (ii) two variations of the mean absolute scaled error (MASE) (Hyndman et al. [24]) to assess demand quantity forecasts, and (iii) stock-keeping-

oriented prediction error cost (SPEC), proposed by Martin et al. [25] as an inventory metric;

3. A new demand classification schema based on the existing literature and our research findings.

We compared the statistical methods proposed by Croston [16], Syntetos et al. [26], and Teunter et al. [27]; the hybrid models developed by Nasiri Pour et al. [28] and Willemain et al. [29]; the ADIDA forecasting method introduced by Nikolopoulos et al. [30]; an extreme learning machine (ELM) model proposed by Lolli et al. [31]; and the VZ_{adj} model described by Hasni et al. [32]. We measured their performance using classification, regression, and inventory metrics. We also developed a compound model that outperforms the ones listed above.

We performed our research on a dataset consisting of 516 time series of intermittent and lumpy demand at a daily aggregation level, corresponding to the demand of European manufacturing companies related to the automotive industry.

The rest of this paper is structured as follows: Section 2 presents related work, Section 3 describes our approach to demand forecasting, with a particular focus on intermittent and lumpy demand, Section 4 describes the features we created for each forecasting model, as well as how we built and evaluated them, and Section 5 describes the experiments we performed and the results obtained. In Section 6, we provide our conclusions and outline future work.

2. Related Work

2.1. Demand Characterization

Many authors have tried to characterize demand to support stock management, determine material planning strategies, and provide cues to decide which forecasting model is most appropriate for each case. A common practice is to classify products into three categories (ABC) according to their cost–volume share: A usually includes items with a large cost–volume share, B is for items with a moderate cost–volume share, and C is for items with a low cost–volume share (Flores et al. [33]). Another three categories are proposed by the FSN approach, which divides products according to their demand velocity into the fast-moving, slow-moving, and non-moving categories (Mittra et al. [34]). XYZ categorization divides items according to their fluctuations in consumption: X for items with almost constant consumption, Y for items with consumption fluctuations (due to trends or seasonality), and Z for items with irregular demand (Scholz-Reiter et al. [35]). Botter et al. [36] mention the VED approach, where products are divided into a different set of categories: vital parts (that cause high losses if not present in stock), essential parts (that cause moderate losses if not available in stock), and desirable parts (that cause minor disruptions to the manufacturing process if not present in stock). ABC categorization has been used jointly with several of the aforementioned categorization schemes. For example, Nallusamy et al. [37] reported using a mixed approach combining the ABC and FSN analyses, while Scholtz et al. [35] combined the ABC and XYZ analyses. Croston [16] proposed another approach that assessed demand based on demand size and inter-demand intervals. Williams [3] considered the variance in the number and size of orders given a particular lead time, classifying items into five categories based on high/low demand sporadicity and size. Based on the author’s empirical findings regarding demand intermittence, a particular category was created for products with a highly sporadic demand occurrence and high demand size variance. Eaves et al. [38] found that the classification schema proposed by Williams did not provide the means to distinguish continuous demand solely based on transaction variability. They proposed dividing demand into five categories based on lead time variability, transaction rate variability, and demand size variability.

$$ADI = \frac{\text{Total Periods}}{\text{Total Demand Buckets}} \quad (1)$$

Equation (1): ADI stands for average demand interval.

$$CV = \frac{\text{Demand Standard Deviation}}{\text{Demand Mean}} \quad (2)$$

Equation (2): CV stands for coefficient of variation. It is computed over non-zero demand occurrences (Syntetos et al. [39]).

Johnston et al. [4] introduced the concept of the average demand interval (ADI, see Equation (1)), which was complemented by Syntetos et al. [5], who introduced the coefficient of variation (CV, see Equation (2)). Both concepts allow us to divide demand into quadrants, i.e., the smooth, erratic, intermittent, and lumpy demand types (see Figure 1). Smooth and erratic demand present regular demand; smooth demand has little variability in demand sizes, while this variability is strong for erratic demands. Intermittent and lumpy demand present irregular demand intervals over time. Intermittent demand has little variability in demand sizes, unlike lumpy demand, which has a greater demand size variability. Thresholds were set based on empirical findings regarding where the methods proposed by Croston [16] and Syntetos et al. [39,40] performed best.

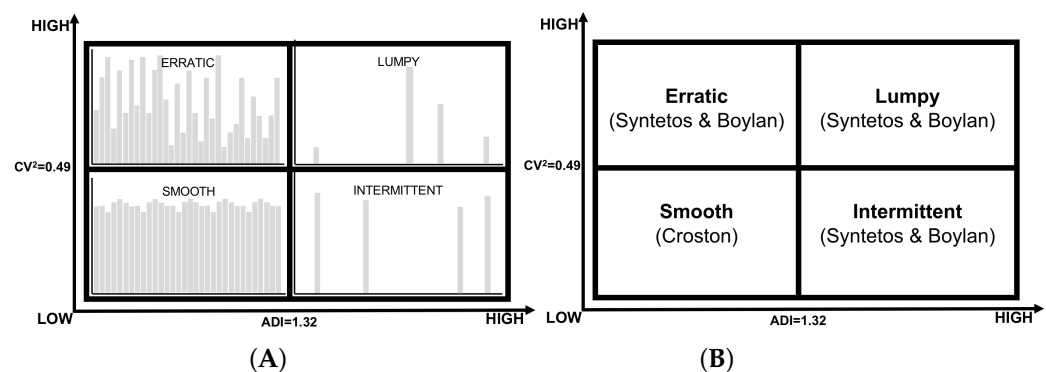


Figure 1. Demand pattern classification. (A) depicts different demand patterns, while (B) shows the classification proposed by Syntetos et al. [5] based on empirical findings.

In this paper, we use the term *continuous demand* to refer to smooth and erratic demand (both of which display regular demand occurrence) and *irregular demand* or *sparse demand* for intermittent and lumpy demand (both of which display irregular or infrequent demand occurrence).

2.2. Forecasting Sparse Demand

In order to mitigate lumpiness, four traditional strategies are frequently used in manufacturing: (a) the use of fixed additional capacity to handle peaks, (b) additional inventories, (c) order rejection during high-demand periods, and (d) temporary increases in manufacturing capacity (e.g., using subcontractors, overtime, or introducing additional shifts) (Arzi et al. [41]). However, accurate forecasts can reduce the need for such approaches, protect the firm's reputation (e.g., by not rejecting consumer orders), and reduce costs (e.g., ensuring enough planned capacity to meet the expected demand).

Forecasting irregular demand is considered a challenging task since it requires considering irregular demand occurrence in addition to the demand size forecast. Box–Jenkins approaches, frequently used for regular time series forecasting, are considered useless in the context of irregular demand (Wallström et al. [42]) since it is challenging to estimate trends and seasonality given the high proportion of zeros. Therefore, researchers developed specialized models to tackle this particular type of demand.

2.3. Demand Forecasting Models

Forecasting irregular demand is considered a challenging task since, in addition to the demand size forecast, it requires taking into account irregular demand occurrence. A seminal work regarding intermittent demand forecasting was developed by Croston [16], who identified exponential smoothing as inadequate for estimating demand when the

mean demand interval between two transactions is greater than two time periods. He proposed a method to estimate the expected interval between transactions and the expected demand size (Equation (3)). Assuming that successive demand intervals and sizes are independent, the inter-demand intervals follow a geometric distribution, and the demand sizes follow a normal distribution. Shenstone et al. [43] showed that Croston's method was not consistent with intermittent demand properties, but its results still outperformed conventional methods. Many researchers followed Croston's approach, either by enhancing this method or proposing similar ones.

$$\text{if } d_t > 0 \begin{cases} a_{t+1} = \alpha \cdot d_t + (1 - \alpha) \cdot a_t \\ p_{t+1} = \alpha \cdot q \cdot d_t + (1 - \alpha) \cdot p_t \\ f_{t+1} = \frac{a_t}{p_t} \end{cases} \quad \text{if } d_t = 0 \begin{cases} a_{t+1} = a_t \\ p_{t+1} = p_t \\ f_{t+1} = f_t \end{cases} \quad (3)$$

Equation (3): Croston's formula [16] for irregular demand estimation, where a is the demand level, p is periodicity, d refers to demand observations, q is previous demand occurrence, and α represents a smoothing constant.

Syntetos et al. [26] proposed a slight modification to Croston's method, known as the Syntetos–Boylan approximation (SBA), in order to avoid a positive correlation between the forecasted demand size and the smoothing constant (Equation (4)). Levén et al. [44] suggested computing a new demand rate every time demand takes place, considering a maximum of one time per time bucket. Teunter et al. [45] considered computing a demand probability for each period and updating the demand quantity forecast only when demand takes place (Equation (5)). In a similar line of research, Vasumathi et al. [46] adapted Croston's method by considering the average of the last two demands. Prestwich et al. [47] proposed a hybrid of Croston's method and Bayesian inference to consider items' obsolescence. Türkmen et al. [48] considered intermittent demand forecasting models as instances of renewal processes and extended Croston's method for a probabilistic forecast. Furthermore, they envisioned that Croston-type methods could be replaced by recurrent neural networks. Chua et al. [49] developed an algorithm that estimated future demand occurrence based on three time series: non-zero-demand periods, inter-arrival periods between demand occurrences, and periods spanning between two demand occurrences. They estimated demand size with a simple moving average.

$$Y_t = \left(1 - \frac{\alpha}{2}\right) \cdot \text{pred}_{\text{Croston}} \quad (4)$$

Equation (4): Syntetos et al. [26] proposed the *Syntetos–Boylan Approximation* as an adjusted version of Croston's [16] forecast formula.

$$\text{if } d_t > 0 \begin{cases} a_{t+1} = \alpha \cdot d_t + (1 - \alpha) \cdot a_t \\ p_{t+1} = \beta \cdot q \cdot d_t + (1 - \beta) \cdot p_t \\ f_{t+1} = a_{t+1} \cdot p_{t+1} \end{cases} \quad \text{if } d_t = 0 \begin{cases} a_{t+1} = a_t \\ p_{t+1} = (1 - \beta) \cdot p_t \\ f_{t+1} = a_{t+1} \cdot p_{t+1} \end{cases} \quad (5)$$

Equation (5): The Teunter, Syntetos, and Babai formula [45] for irregular demand estimation, where a is the demand level, p is the probability of demand occurrence, d refers to demand observations, q is previous demand occurrence, and α represents a smoothing constant.

Wright [50] developed linear exponential smoothing, an adaptation of Holt's double exponential smoothing model that considers variable reporting frequency and irregularities in time spacing to compute and update a trend line with exponential smoothing. Altay et al. [51] demonstrated that this method helps forecast intermittent demand where the trend is present. Sani et al. [52] and Ghobbar et al. [53] found that averaging methods can provide acceptable performance in some cases, despite demand intermittency. Chatfield et al. [54] suggested using a zero-demand model for highly lumpy demand, where the holding cost is much higher than the shortage cost. Gutierrez et al. [55] proposed forecasting lumpy demand with a three-layer multilayer perceptron (MLP) (Rosenblatt [56]),

considering only two inputs: the demand of the immediately preceding period and the number of periods separating the last two non-zero-demand transactions. Along the same line, Lolli et al. [31] considered using ELM (Huang et al. [57]) and training the model based on three input features: (a) the demand size for the previous time period, (b) the number of periods separating the last two non-zero-demand transactions considering the immediately preceding period, and (c) the cumulative number of successive periods with zero demand.

Some researchers proposed performing data aggregation to achieve smooth time series to cope with event sparsity. Much research was performed on the effect of aggregation on regular time series (Hotta et al. [58], Souza et al. [59], Athanasopoulos et al. [60], Rostami-Tabar et al. [61,62], Petropoulos et al. [8], and Kourentzes et al. [63]), showing that higher aggregation improves forecast results. One such approach is the aggregate–disaggregate intermittent demand approach (ADIDA) (Nikolopoulos et al. [30]), which is a three-stage process: (i) perform time-series aggregation (either overlapping or non-overlapping aggregation), (ii) forecast the next time series value over the aggregated time series, and (iii) disaggregate the forecasted value to the original aggregation level.

Following Croston’s intuition [16], some researchers developed separate models to forecast demand occurrence and demand size. Willemain et al. [29] proposed to model demand occurrence as a Markov process and forecast demand size by randomly sampling past demand sizes and eventually jittering them to account for not-yet-seen values. Hua et al. [64] followed a similar approach, attributing demand occurrence to autocorrelation or explanatory variables. If they attributed demand occurrence to autocorrelation, they predicted it based on Markov processes. Otherwise, they used a logistic regression model. Nasiri Pour et al. [28] developed a hybrid approach, forecasting demand occurrence with a neural network (see Figure 2) while they estimated demand size with exponential smoothing. The neural network they proposed considered four input variables: (a) the demand size at the end of the preceding period, (b) the number of periods between the last two demand occurrences, (c) the number of periods between the target period and the last demand occurrence, and (d) the number of periods between target period and first immediately preceding zero-demand period. The authors considered demand to not occur if the network forecasted a zero value and considered demand to take place if the predicted value was greater than zero.

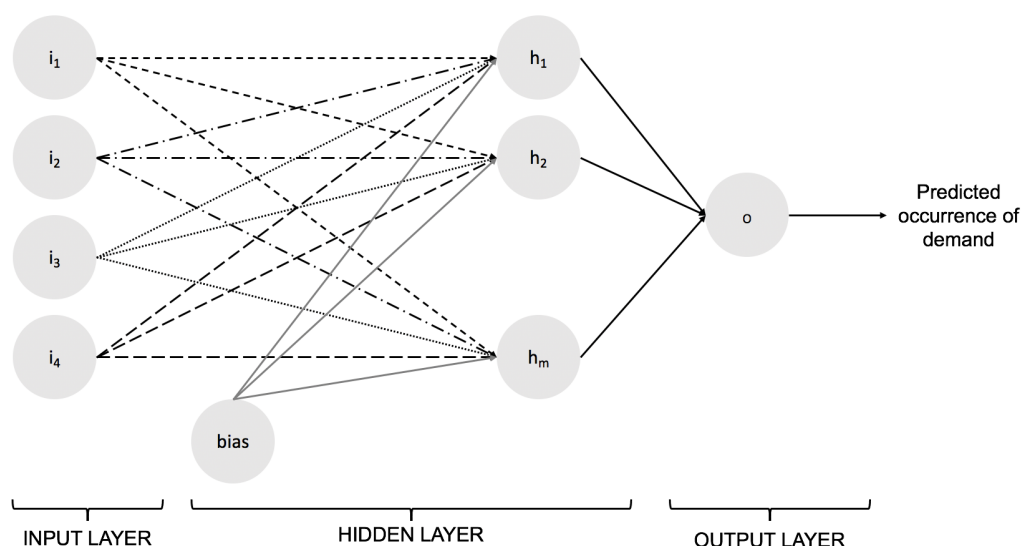


Figure 2. MLP for the hybrid approach proposed by Nasiri Pour et al. [28]. The inputs to the model are the demand size at the end of the preceding period, the number of periods between the last two demand occurrences, the number of periods between the target period and the last demand occurrence, and the number of periods between the target period and the first immediately preceding zero-demand period.

Finally, Petropoulos et al. [65] developed an alternative perspective to the ADIDA framework (Nikolopoulos et al. [30]), aggregating time series in such a way that each time bucket contained a single demand occurrence. As a result, the transformed time series no longer presented intermittency, and it forecasted the time-varying number of periods when such demand would occur. Then, based on the mean values for the inter-demand interval and the coefficient of variation, demand size was estimated using Croston's method, SBA, or simple exponential smoothing. A similar approach was described by Hasni et al. [32]. The authors proposed forecasting demand by repeatedly random sampling from the frequency histograms of past inter-demand intervals and demand sizes.

Forecasting Features

Bartezzaghi et al. [66] considered demand lumpiness as a consequence of different market characteristics, such as the numerosness and heterogeneity of customers, the frequency at which customers place the orders, the variety of customer requests (e.g., high customization in make-to-order settings (Verganti [67])), and the correlations between customer behavior). Lumpiness is also related to the granularity level at which demand is considered (e.g., visualizing demand at a client and product level vs. only at a product level or visualizing daily demand vs. monthly). Higher aggregation levels usually reduce the number of periods without demand, changing the demand pattern classification.

In the scientific literature related to the demand forecasting of intermittent and lumpy demand patterns, authors describe multiple characteristics and features relevant to demand occurrence forecasting. Among them, we find the average inter-demand interval size (Levén et al. [44]), the previous demand event occurrence (Gutierrez et al. [55]), the distribution of inter-demand interval sizes (Croston [16]), demand size (Nasiri Pour et al. [28]), demand shape distribution (Bartezzaghi et al. [68] and Zotteri [69]), the usage of early information generated by customers during the purchasing process (Verganti [67]), the presence of paydays, billing cycles, or holidays (Hyndman et al. [24]), demand event autocorrelation (Willemain et al. [70]), demand event correlation (products being complementary or alternate) (Arzi et al. [41]), and whether items may be purchased by the same supplier or shipped using the same transportation mode (Syntetos [71]).

We found that two techniques were applied to estimate demand size across all cases. The first one was exponential smoothing (and its variants), and it was applied across previous non-zero demand sizes (Nasiri Pour et al. [28]). The second one was the use of jittering on top of randomly sampled past demand sizes to account for yet-unseen values (Willemain et al. [29] and Hua et al. [64]). Altay et al. [51] described means to adjust demand size values based on the presence of trends in data.

On top of the above-mentioned approaches, two approaches used to reduce uncertainty regarding lumpiness can help create features: early sales and order overplanning (Verganti [67] and Bartezzaghi et al. [66]). The early sales approach considers information regarding actual orders received for future delivery, where future demand can be estimated given that some degree of correlation exists between the unknown and known portions of the demand. On the other hand, the order overplanning approach focuses on every customer instead of the overall demand. It, therefore, enables collecting and using information specific to each customer and their future needs.

2.4. Metrics

The measurement of forecasting models' performance for lumpy and intermittent demand has been a subject of extensive research (Anderson [72]). Syntetos et al. [71] compared the performance of the Mean Signed Error, Wilcoxon Rank Sum Statistic, Mean Square Forecast Error, Relative Geometric Root Mean Square Error (RGRMSE), and Percentage of times Better metrics (PB). They concluded that the RGRMSE behaves well in the context of irregular demand. Teunter et al. [73] pointed out that the RGRMSE cannot be applied on a single item for zero or moving average forecasts (it would result in zero error). Hemeimat et al. [74] suggested using the tracking signal metric, which is calculated by dividing the most re-

cent sum of forecast errors by the most recent estimate of the mean absolute deviation (MAD). Among many metrics, Hyndman et al. [24] suggested using the MASE for lumpy and intermittent demand to provide a scale-free assessment of the accuracy of demand size forecasts. Although traditional per-period forecasting metrics, such as the root mean squared error (RMSE), the mean squared error (MSE), the MAD, and the mean absolute percentage error (MAPE), were widely used in the literature regarding irregular demand, Teunter et al. [73] and Kourentzes [75] showed they were not adequate due to the high proportion of zeros. Prestwich et al. [76] proposed computing a modified version of the error measures that considered the mean of the underlying stochastic process instead of the point demand for each point in time. Finally, it is relevant to point out that two metrics were used in the M5 competition (Makridakis et al. [77]) to assess time series with respect to irregular sales: the root mean squared scaled error (RMSSE, introduced by Hyndman et al. [24]) and the weighted RMSSE. With these metrics, using a score that considers squared errors, result measurements optimize towards the mean. The weighted RMSSE variant allows for penalizing each time series error based on various criteria (e.g., item price). However, both metrics unevenly penalize products sold during the whole time period in contrast to those that are not.

Syntetos et al. [78] noted that regardless of the metrics used to estimate how accurate demand forecasts are, it is crucial to measure the impact of forecasts on stock-holding and service levels. Along this line, Wallström et al. [42] proposed two complementary metrics. The first one was the number of shortages, counting how many times the cumulated forecast error was over zero during the time interval of interest. The second one was periods in stock (PIS), defined as the number of periods the forecasted items spent in fictitious stock (or how many stock-out periods existed). More recently, Martin et al. [25] proposed the stock-keeping-oriented prediction error cost (SPEC), which considers, for each time step, whether demand forecasts translate into costs of opportunity or stock-keeping costs; the result is never both at the same time. We summarize the metrics adopted in relevant related works in Table 1.

Table 1. Metrics identified in main related works we reviewed on the topic of lumpy and intermittent demand.

[illegible]

3. Reframing Demand Forecasting

3.1. A Classification of Existing Demand Forecasting Models for Lumpy and Intermittent Demand

From the models we presented in Section 2, we observe that certain patterns arose in how the demand forecasting problem for lumpy and intermittent demand was framed. We, therefore, classify the forecasting approaches into four categories:

- **Type I:** uses a single model to predict the expected demand size for a given time step.
- **Type II:** uses aggregation to remove demand intermittency and benefit from regular time series models to forecast demand.
- **Type III:** uses separate models to estimate whether demand will take place at a given point in time and the expected demand size.
- **Type IV:** uses separate models to estimate the demand interval and demand size.

We classify the models analyzed in the related works according to these four categories in Table 2.

Table 2. Model types identified in main related works we reviewed on the topic of lumpy and intermittent demand.

Model Type	Related Work
I	[16,26,31,44–46,49–55]
II	[8,30,58–61,63]
III	[28,29,64]
IV	[32,65]

3.2. Demand Characterization and Forecasting Models

Croston [16] developed the idea to consider two components for intermittent demand forecasts: demand occurrence and demand sizes. Syntetos et al. [5] considered these two components and developed an influential system of demand categorization, dividing demand into four types: smooth, erratic, intermittent, and lumpy, based on the coefficient of variation and the average demand interval. The need to separately address these two dimensions was further recognized by Nikolopoulos et al. [84].

Many authors followed Croston's lead, developing separate models to estimate demand occurrence and demand size (e.g., Willemain et al. [29], Hua et al. [64], and Nasiri Pour et al. [28]), though none of them measured the performance of the demand occurrence component. We thus propose decoupling the demand forecasting problem into two sub-problems, each of which requires a separate model with separate features and metrics: (i) demand occurrence, addressed as a classification problem, and (ii) demand size estimation, addressed as a regression problem. Following the original work by Syntetos et al. [5] and the division mentioned above, we propose an alternative demand categorization schema. Considering demand occurrence and demand quantity forecasting as two different problems, we can divide demand into two types (see Figure 3). The first demand type is 'R', which refers to demand with regular demand event occurrence. Since demand (almost) always occurs in this case, estimating demand occurrence is rendered irrelevant and does not pose a challenge when developing a regression model to estimate demand size. The second demand type is 'C+R', which refers to demand with an irregular occurrence. This second case benefits from models that consider both demand occurrence and demand size (see Section 2.3).

In Figure 4, we present a demand forecasting model architecture and a flowchart describing how to build it and issue demand forecasts. Since the classification and regression models address different problems, we expect them to use different features to help achieve their goals. We described the aspects relevant to demand occurrence and demand size forecasting presented in the literature in Section 2.3. These can be used as the features of the classification and regression models.

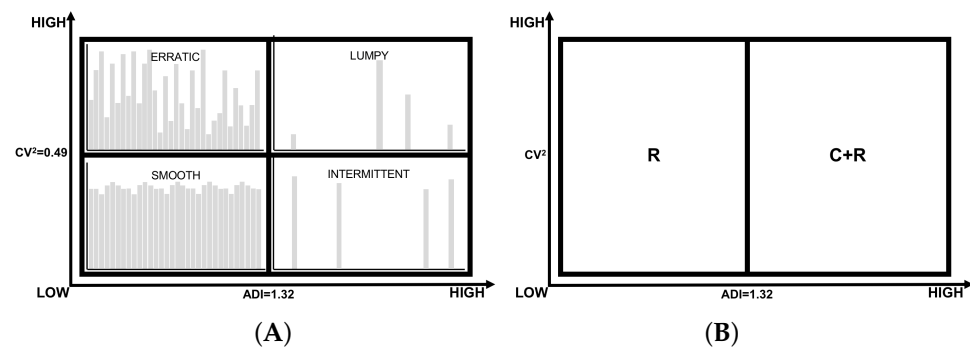


Figure 3. Demand categorization schemas. On the right, (A) corresponds to the influential categorization developed by Syntetos et al. [5]. On the left, (B) proposes a new schema that only considers demand occurrence, dividing demand into two groups. ‘R’ denotes regular demand occurrence, where demand size can be predicted with a regression model. ‘C+R’ denotes irregular demand occurrence that requires a model to predict demand occurrence and a model to predict demand size.

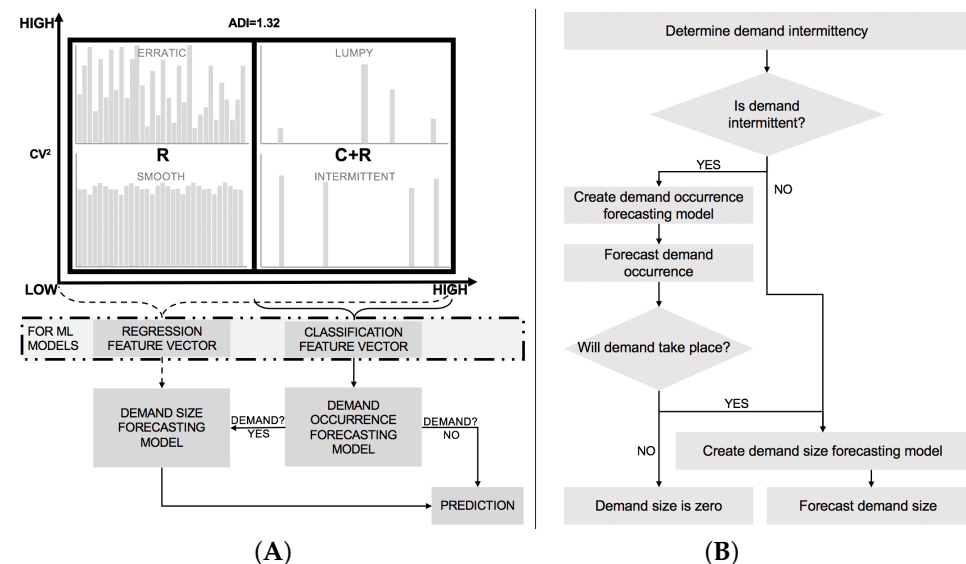


Figure 4. Two-fold machine learning approach to demand forecasting. (A) shows a basic architecture for demand forecasting when reframing demand forecasting as classification and regression problems. (B) shows a flowchart with steps followed to create the demand forecasting models and issue demand forecasts.

An increasing body of research suggests that global machine learning time series models (models built with multiple time series) provide better results than local ones (models considering time series corresponding to a single product) (Bandara et al. [22] and Salinas et al. [21]). Furthermore, increased performance is observed even when training models with disparate time series that have different magnitudes or may not be related to each other (Laptev et al. [85] and Rožanec et al. [86]), although how to bound the maximum possible error in such models remains a topic of research. Given this insight, we conclude that dividing demand based on the coefficient of variation provides limited value and is no longer relevant to demand categorization.

We keep the cut-off value of $ADI = 1.32$ proposed by Syntetos et al. [5] as a reference. This cut-off value, jointly with the CV^2 , is accepted as a measure of whether a collection of observed demand is smooth, lumpy, intermittent, or erratic (Lowas III et al. [87]); it remains relevant for statistical methods. Nevertheless, we consider its relevance to blur with respect to different machine learning models. Its importance may be rendered irrelevant for global machine learning classification models that predict demand occurrence. By considering multiple items simultaneously, global models perceive a higher density of demand

events and less irregularity than models developed with data regarding a single demand item. Simultaneously, the model can learn underlying patterns, which may be related to specific behaviors (e.g., deliveries that take place only on certain days). It is important to note that event scarcity usually results in imbalanced classification datasets, posing an additional challenge.

We present suggested metrics to assess each model, and the overall demand forecasting performance, in Section 3.3.

3.3. Metrics

Though several authors (e.g., [16,39,83]) considered separating demand occurrence and demand size when forecasting irregular demand, in the scientific literature we reviewed (see Table 1), we found that researchers measured them separately. We thus conclude that they did not consider demand occurrence and demand size forecast as entirely different problems.

Considering irregular demand forecasting only as a regression problem led to much research and discussion (presented in Section 2.4) on how to mitigate and integrate zero-demand occurrence to measure demand forecasting models' performance adequately. In our research, we provide a different perspective. We adopt four metrics to assess the performance of demand forecasting models: (i) AUC ROC to measure how accurately the model forecasts demand occurrence, (ii) two variants of MASE to measure how accurately the model forecasts demand size, and (iii) SPEC to measure how the forecast impacts inventory. When measuring SPEC, we consider $\alpha_1 = \alpha_2 = 0.5$ since we have no empirical data that would support weighting α asymmetrically.

$$\text{SPEC}_{\alpha_1, \alpha_2} = \frac{1}{n} \sum_{t=1}^n \sum_{i=1}^t \left(\max \left[0; \min \left[y_i; \sum_{k=1}^i y_k - \sum_{j=1}^t f_j \right] \cdot \alpha_1; \min \left[f_i; \sum_{k=1}^i f_k - \sum_{j=1}^t y_j \right] \cdot \alpha_2 \right] \cdot (t - i + 1) \right) \quad (6)$$

Equation (6): Stock-keeping-oriented Prediction Error Cost (SPEC).

AUC ROC is widely adopted as a classification metric and has many desirable properties, such as being threshold independent and invariant to a priori class probabilities. MASE has the desirable property of being scale-invariant. We consider two variants (namely MASE_I and MASE_{II}). Following the criteria in Wallström et al. [42], we compute MASE_I for the time series that results from ignoring zero-demand values. By doing so, we assess how well the regression model performs against a naïve forecast, assuming a perfect demand occurrence prediction. On the other hand, MASE_{II} is computed for the time series that considers all points where demand either took place or is predicted according to the classification model. By doing so, we measure the impact of demand event occurrence misclassification on the demand size forecast. When the model predicting demand occurrence has perfect performance, (i) should equal (ii). Finally, we compute SPEC for the whole time series (considering zero and non-zero demand occurrences). This way, the metric measures the overall forecast impact on inventory, weighting stock-keeping, and opportunity costs.

4. Methodology

4.1. Business Understanding

Demand forecasting is critical to supply chain management since its outcomes directly affect the supply chain and manufacturing plant organizations. This research focuses on demand forecasting for a European original equipment manufacturer in the automotive industry. We explored providing demand forecasts for each material and client daily. Such forecasts enable highly detailed planning. From the forecasting perspective, accurate forecasts at a daily level can leverage the most recent information, which is lost at higher aggregation levels for non-overlapping aggregations. They also avoid imprecisions that result from higher-level forecast disaggregations. We tackled the demand forecasting problem as a modeling approach. In particular, we developed machine learning models that learn

from past data to issue forecasts. In the following subsections, we provide a detailed insight into the steps we followed to create such predictive models.

4.2. Data Understanding

For this research, we used a dataset with three years of demand data extracted from enterprise resource planning software. We considered records that accounted for products shipped from manufacturing locations. For demand data, demand was registered on the day products shipped. The dataset comprised 516 time series corresponding to 279 materials and 149 clients. When categorized according to the schema proposed by Syntetos et al. [5], we found that 49 corresponded to a lumpy demand pattern and 467 to an intermittent demand pattern (see Figure 5). In Tables 3 and 4, we provide summary statistics for the time series corresponding to each demand pattern. We found that demand occurrence for both sets of time series was highly infrequent, having a mean of one demand event in almost two months or more.

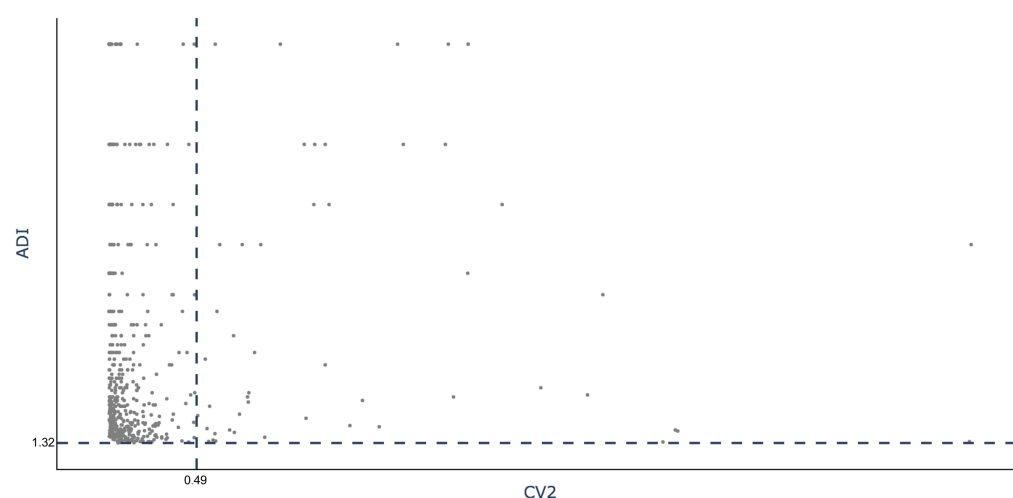


Figure 5. The ADI- CV^2 categorizations for the time series based on the classification proposed by Syntetos et al. [5].

Table 3. Summary statistics for 49 lumpy demand time series.

Metric	Mean	Std	Min	25%	50%	75%	Max
ADI	86.00	87.26	1.97	11.86	37.29	156.60	261.00
CV^2	1.44	1.04	0.50	0.70	1.10	1.90	4.83

Table 4. Summary statistics for 467 intermittent demand time series.

Metric	Mean	Std	Min	25%	50%	75%	Max
ADI	56.72	70.58	1.41	9.79	25.26	71.18	261.00
CV^2	0.09	0.11	0.00	0.02	0.05	0.13	0.48

4.3. Data Preparation, Feature Creation, and Modeling

We forecasted irregular demand with two separate models: a classification model to predict demand occurrence and a regression model to predict demand size. Though source data were the same for both, different features were required to address each model's goals. Based on knowledge distilled from the literature, we created the features presented in Section 2.3 and some features of our own.

To model demand occurrence, we considered weekdays since last demand, the day of the week of the last demand occurrence, the day of the week for the target date, the mean of the inter-demand intervals, the mean of the last inter-demand intervals (across all products), and the skew and kurtosis of the demand size distributions, among other factors.

To model demand size, we considered, for each product, the size of the last demand, the average of the last three demand occurrences, the median value of past occurrences, the most frequent demand size value, and the exponential smoothing of past values, among other factors.

When computing feature values, we considered two forecasting horizons, namely fourteen and fifty-six days (Figure 6), to understand how the forecasting horizon size affects forecasts. The forecasting horizons were selected based on a business use case.

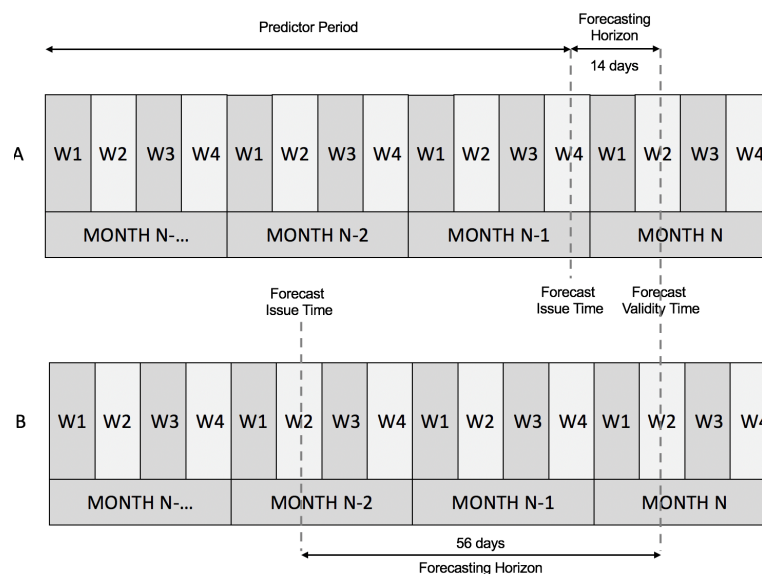


Figure 6. We computed predictions for two forecasting horizons, i.e., 14 (A) and 56 (B) days, to test the sensitivity of predictions regarding demand occurrence and demand size to the forecasting horizon.

To forecast irregular demand, we compared nine methods described in the literature:

- **Croston:** Croston's method [16] (see Equation (3)).
- **SBA** [26] (see Equation (4)).
- **TSB** [27] (see Equation (5)).
- **MC+RAND:** a hybrid model proposed by Willemain et al. [29]. Demand occurrence is estimated as a Markov process, while demand sizes are randomly sampled from previous occurrences.
- **NN+SES:** a hybrid model proposed by Nasiri Pour et al. [28]. Considers a NN model (see Figure 2) to forecast demand occurrence; demand size is computed by exponential smoothing over non-zero demand quantities in past periods. We used the following parameters for the NN: a maximum of 300 iterations, a constant learning rate of 0.01, and a hyperbolic tangent activation. Given that no description was given on whether scaling was applied to the dataset prior to training the network, we explored two models: without feature scaling ($NN_{NS}+SES$) and with feature scaling ($NN_{WS}+SES$).
- **ADIDA** forecasting method, proposed by Nikolopoulos et al. [30], which removes intermittence through aggregation and then disaggregates the forecast back to the original aggregation level.
- **ELM:** an ELM model as proposed by Lolli et al. [31]. We initialized the model with the following parameters: 15 hidden units, ReLU activation, a regularization factor of 0.1, and normal weight initialization. We trained two models: ELM(C1) (two models, trained per demand type) and ELM(C2) (global model, considering all the demand types).
- **VZ_{adj}:** a method proposed by Hasni et al. [32], considering only positive demands when the predicted lead-time demand was equal to the forecasting horizon considered.

We also developed models of our own. We created a CatBoost model (Prokhorenkova et al. [88]) to forecast demand occurrence and compare six models to forecast demand size: naïve, most frequent value (MFV), moving average over last three demand periods

(MA(3)), simple exponential smoothing (SES), random sampling from past values with jittering (RAND), and LightGBM regressor (ML). We used the LightGBM regression algorithm (Ke et al. [89]) because four of the top five time series forecasting models in the M5 competition were based on this algorithm (Makridakis et al. [77]).

CatBoost is an implementation of gradient-boosted decision trees. During training time, it sequentially builds a set of symmetric binary decision trees, ensuring that each new tree built reduces the loss compared to previous trees. The algorithm avoids one-hot-encoding categorical features by computing the frequency of the occurrence of particular values, reducing sparsity while enhancing computation times. CatBoost uses gradient descent to minimize a cost function, which informs how successful it is at meeting the classification goal. Since the dataset regarding demand occurrence was heavily imbalanced (less than 6% of instances corresponded to demand occurrence), we chose to optimize the model training with focal loss (Lin et al. [90]). Focal loss has the desirable property of asymmetric penalization of training samples, focusing on misclassified ones to improve the overall classification.

Our CatBoost classification model was built with a maximum depth of 2, 150 iterations, focal loss, and the AUC ROC evaluation metric. The LightGBM regressor was built with a maximum depth of 2, the RMSE objective, gradient-boosting decision tree boosting, 100 estimators, a learning rate of 0.1, and 31 tree leaves for base learners.

5. Experiments and Results

This section describes the experiments we conducted and assesses their results with the metrics described in Section 3.3. For the SPEC metric, we considered α_1 and α_2 equal to 0.5. We summarize our experiments in Table 5. In Table 6, we summarize the results obtained for our own models, while in Table 7, we compare the best-performing of our models against the models described in the scientific literature and described above (in Section 4.3).

We adopted two forecasting horizons (fourteen and fifty-six days) to understand how sensitive the existing approaches were to forecast lead time. To evaluate the models, we used nested cross-validation (Stone [91]), which is frequently used to evaluate time-sensitive models. We tested our models by having them make predictions at the weekday level for six months of data. For the classification models, we measured AUC ROC, with prediction scores cut at a threshold of 0.5. The only exception to this was the model by Nasiri Pour et al. [28] since the authors explicitly stated that they considered any prediction above zero as an indication of demand occurrence.

Table 5. Description of the reference models we evaluated for demand occurrence forecasting and demand size estimation.

Model Task	Model Type	ID	Models	Data
Classification	Global, per demand type (lumpy or intermittent)	C1	CatBoost	Demand occurrence features
	Global, over all instances (lumpy and intermittent)	C2	CatBoost	Demand occurrence features
Regression	Local, one per each time series	R1	Naive SES MA(3) MFV RAND	Past non-zero demand sizes
	Global, per demand type (lumpy or intermittent)	R2	LightGBM	Demand size features
	Global, over all instances (lumpy and intermittent)	R3	LightGBM	Demand size features

Table 6. Overall results obtained with the models we proposed, for both forecasting horizons. Best results are bolded, second-best results are displayed in italics.

Model	Regression	Forecasting Horizon: 14 Days				Forecasting Horizon: 56 Days			
		AUC ROC ↑	MASE _I ↓	MASE _{II} ↓	SPEC _{median} ↓	AUC ROC ↑	MASE _I ↓	MASE _{II} ↓	SPEC _{median} ↓
C1R1	Naive	<i>0.9408</i>	0.5764	1.1084	121.9809	<i>0.9409</i>	0.4861	1.1084	121.9809
	MA(3)	<i>0.9408</i>	0.5482	1.0544	281.2004	<i>0.9409</i>	0.4624	1.0544	281.2004
	SES	<i>0.9408</i>	0.5229	1.0058	1210.7634	<i>0.9409</i>	0.4410	1.0058	1210.7634
	MFV	<i>0.9408</i>	0.5437	1.0457	141.4351	<i>0.9409</i>	0.4585	1.0457	141.4351
	RAND	<i>0.9408</i>	0.7552	1.4524	2011.5744	<i>0.9409</i>	0.6353	1.4485	2675.2844
C1R2	ML	<i>0.9408</i>	0.5813	1.1183	46,422.3206	<i>0.9409</i>	0.4917	1.1215	46,162.9523
C1R3	ML	<i>0.9408</i>	0.5758	1.1250	48,807.4847	<i>0.9409</i>	0.4613	1.1274	48,279.0973
C2R1	Naive	0.9700	0.5271	1.0460	<i>110.3435</i>	0.9700	0.4445	1.0460	<i>110.3435</i>
	MA(3)	0.9700	0.4906	0.9736	245.1851	0.9700	0.4137	0.9736	245.1851
	SES	0.9700	0.4611	0.9152	1343.2328	0.9700	0.3888	0.9152	1343.2328
	MFV	0.9700	<i>0.4760</i>	<i>0.9448</i>	94.8092	0.9700	<i>0.4014</i>	<i>0.9448</i>	94.8092
	RAND	0.9700	0.7267	1.4422	2034.5172	0.9700	0.5975	1.4059	2938.0534
C2R2	ML	0.9700	0.5194	1.0310	44,255.2309	0.9700	0.4398	1.0352	44,101.6088
C2R3	ML	0.9700	0.5295	1.0510	39,918.1584	0.9700	0.4479	1.0542	40219.3378

Table 7. Comparison of methods found in related works and three of the best models we created: C2R1-SES, C2R1-MFV, and C2R1-Naïve. Our three models achieved the highest AUC ROC and showed competitive MASE_I and MASE_{II} values. C2R1-MFV achieved the best overall performance on the SPEC metric, while C2R1-Naïve achieved the second-best result. Best results are bolded, and second-best results are presented in italics.

Model	Forecasting Horizon: 14 days				Forecasting Horizon: 56 days			
	AUC ROC ↑	MASE _I ↓	MASE _{II} ↓	SPEC _{median} ↓	AUC ROC ↑	MASE _I ↓	MASE _{II} ↓	SPEC _{median} ↓
Croston [16]	0.5000	1.5997	96.2732	182,590,225.2842	0.5000	1.4769	1.4769	173,846,999.3033
SBA [26]	0.5000	1.5196	91.4593	173,457,612.1803	0.5000	1.4047	88.2762	165,151,599.2596
TSB [27]	0.5337	1.0340	42.8525	82,848,841.2377	0.5448	0.7491	11.9425	2,0024,649.5191
MC+RAND [29]	0.5000	0.8616	1.3786	199.3702	0.5000	0.8619	1.3786	199.3702
NN _{WS} +SES [28]	0.5000	0.3872	<i>0.8588</i>	18,804,861.1954	0.5000	0.3825	<i>0.8588</i>	18,240,504.9768
NN _{NS} +SES [28]	0.5000	0.3834	<i>0.8588</i>	18,804,861.1954	0.5000	0.3825	<i>0.8588</i>	18,240,504.9768
ELM(C1) [31]	<i>0.6931</i>	1.1020	0.0004	1593.4317	0.7024	<i>0.3087</i>	0.0004	1566.1803
ELM(C2) [31]	<i>0.6955</i>	0.1034	0.0004	1605.4686	0.6967	0.0317	0.0004	1581.5055
VZ _{adj} [32]	0.5000	0.9998	1.9864	23,883.8074	0.5000	0.9922	1.9864	23,883.8074
C2R1-Naïve	0.9700	0.5271	1.0460	<i>110.3435</i>	0.9700	0.4445	1.0460	<i>110.3435</i>
C2R1-SES	0.9700	0.4611	0.9152	1343.2328	0.9700	0.3888	0.9152	1343.2328
C2R1-MFV	0.9700	0.4760	0.9448	94.8092	0.9700	0.4014	0.9448	94.8092

We observed that although the approaches in the literature provided different means of estimating demand occurrence, their performance results were all close to 0.5 AUC ROC. Differences in performance were mainly driven by the method used to estimate demand size.

All the models we developed in our experiments strongly outperformed the models replicated from the literature. When considering AUC ROC, our models achieved scores of at least 0.94, almost doubling every model described in the literature. When considering regression metrics, our models displayed three to four times better MASE_I and MASE_{II} values and even larger differences regarding the SPEC metric for most models. The only exceptions were the ELM(C1) and ELM(C2) models, which achieved better MASE metrics but not competitive AUC ROC or SPEC values. These results confirm the importance of considering the demand forecasting of irregular demand patterns as two separate problems (demand occurrence and demand size), each with its features and optimized against its own set of metrics. The improvements in classification scores substantially impacted demand size and inventory metrics.

We obtained the best results with the CatBoost classifier trained over all instances, making no distinction between lumpy and intermittent demand. The model achieved a

high classification AUC ROC score, reaching 0.97 for both the fourteen-day and fifty-six-day horizons. Among the regression models used to estimate demand size, we achieved the best results with SES, MFV, and Näive. SES outperformed every other model with respect to $MASE_I$ and $MASE_{II}$, while MFV had the best median SPEC score and remained competitive on $MASE_I$ and $MASE_{II}$ values. The Näive variant achieved the second-best SPEC score while remaining competitive on MASE values.

When building the global classification model, we were interested in how much better it performed than models built using only lumpy or intermittent demand. We found that the model built only on lumpy or intermittent demand achieved an AUC ROC of at least 0.7368 and 0.9666, respectively, in each subset of products, whereas the global model built on all of the time series increased the performance to 0.9097 and 0.9776, respectively (see Table 8).

Table 8. Comparison of AUC ROC for lumpy and intermittent demand, obtained from the C1 and C2 models. Using all data for a single classification model to predict demand occurrence showed improvements for both groups and time horizons. The largest improvement was observed for lumpy demand, with an improvement greater than 0.17.

Model	Forecasting Horizon: 14 Days		Forecasting Horizon: 56 Days	
	AUC ROC _{lumpy} ↑	AUC ROC _{intermittent} ↑	AUC ROC _{lumpy} ↑	AUC ROC _{intermittent} ↑
C1	0.7368	0.9666	0.7379	0.9666
C2	0.9097	0.9776	0.9097	0.9776

The results show that the forecasting horizon has little influence on classification performance. We attribute this difference to the fact that demand occurrence is scarce; thus, changes in demand behavior are likely to be slow. This fact has engineering implications since there is no need to retrain and deploy the classification model frequently. Achieving the best demand size forecasts with SES and MFV also means that demand forecasting does not require expensive computations or much maintenance in production.

Finally, we compared our C2R1-SES model against the ADIDA approach. We selected 64 products that had continuous demand over the three years we considered and used SES to provide forecasts two months in advance; we measured MASE for the last six months using cross-validation. The C2R1-SES model with a fifty-six-day forecasting horizon showed strong performance, achieving a MASE of 0.0052; for ADIDA-SES, we measured a MASE of 1.6640. Our model outperformed the state-of-the-art self-improving mechanism, ADIDA, at an aggregate level. In the light of the results obtained, we conclude that our approach has at least two significant advantages over aggregate forecasts. First, we avoid issues related to forecasting disaggregation, provided the classification model is good enough. Second, our models benefit from the information that is lost with aggregation, which may help to issue better forecasts.

Our results demonstrate that the proposed approach achieved state-of-the-art performance for lumpy and intermittent demand. We attribute the strong performance not only to the accurate demand size estimation but also to the ability to predict the demand event occurrence accurately, which is an aspect that has been overlooked so far. Our two-fold approach provides additional insights to decision-makers: the classification model can provide accurate forecasts regarding demand event occurrence. Furthermore, probability calibration can be used to ensure that the model outputs the probability of a demand event occurrence for a more informative and intuitive score interpretation (Song et al. [92]). The ability to accurately predict demand occurrence for lumpy and intermittent demand (which account for 75% of manufactured products, as per Johnston et al. [11]) will likely have a strong impact on the supply chain, given that it will enable better order and production planning and lower stock costs due to reduced uncertainty.

We believe that the approach we propose for demand forecasting could be used to forecast sparse time series. Furthermore, AUC ROC and MASE metrics could be used to assess the quality of forecasts for any sparse time series. AUC ROC measures how well the

model predicts the event occurrence, while MASE assesses how close the forecasted values correspond to the ones observed in the time series.

6. Conclusions

This research proposes a new look at the demand forecasting problem for infrequent demand patterns. Breaking it down into two prediction problems (classification for demand occurrence and regression for demand size), we (i) enabled accurate model diagnostics and (ii) optimized each model for its specific task. Our results show that such decomposition enhances overall forecasting performance. Analyzing models proposed in the literature, we found that most of them underperformed by first failing to predict demand occurrence accurately. In addition to the decomposition of the forecasting problem, we proposed a set of four metrics (AUC ROC for classification, $MASE_I$ and $MASE_{II}$ for regression, and SPEC to assess the impact on inventory). We also developed a novel model that outperformed not only seven models described in the literature for lumpy and intermittent demand but also the state-of-the-art self-improving mechanism known as ADIDA. Considering the problem separation mentioned above, we propose a new demand classification schema based on the approach described herein that provides a good demand forecast. We consider two types of demands: 'R' for demands where only regression is required and 'C+R' where classification and regression models estimate demand occurrence and size, respectively.

We envision several directions for future research. First, we would like to replicate these experiments on some widely cited datasets, such as SKUs of the automotive industry (initially used by Syntetos et al. [26]) or SKUs of the Royal Air Force (used initially by Teunter et al. [73]). Second, we enable new approaches to dealing with forecasts involving item obsolescence, stationarity, and trends in irregular demands by decoupling demand occurrence from demand size. Third, we hypothesize that this approach can be extended to the forecasting of irregular and infrequent sales, significantly impacting retail. Finally, we believe this approach can be applied in other domains with time series displaying scarce event occurrences.

Author Contributions: Conceptualization, J.M.R.; methodology, J.M.R.; software, J.M.R.; validation, J.M.R.; formal analysis, J.M.R.; investigation, J.M.R.; resources, B.F. and D.M.; data curation, J.M.R. and B.F.; writing—original draft preparation, J.M.R.; writing—review and editing, J.M.R. and D.M.; visualization, J.M.R.; supervision, B.F. and D.M.; project administration, B.F. and D.M.; funding acquisition, B.F. and D.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Slovenian Research Agency and the European Union's Horizon 2020 program project FACTLOG under grant agreement number H2020-869951.

Data Availability Statement: Data not available due to restrictions.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

A-MAPE	Alternative mean absolute percentage error
ADI	Average demand interval
ADIDA	Aggregate-disaggregate intermittent demand approach
AUC ROC	Area under the curve of the receiver operating characteristic
CV	Coefficient of variation
ELM	Extreme learning machine
FSN	Fast-slow-non moving
GMAE	Geometric mean absolute error
GMMAE	Geometric mean (across series) of the arithmetic mean (across time) of the absolute errors
GMRAE	Geometric mean relative absolute error
MAD	Mean absolute deviation
MADn	Mean absolute deviation over non-zero occurrences
MAE	Mean absolute error
MAPE	Mean absolute percentage error

MAR	Mean absolute error
MASE	Mean absolute scaled error
MdAE	Median absolute error
MdRAE	Median relative absolute error
ME	Mean error
MFV	Most frequent value
mGMRAE	Mean-based geometric mean relative absolute error
MLP	Multilayer perceptron
mMAE	Mean-based mean absolute error
mMAPE	Mean-based mean absolute percentage error
mMdAE	Mean-based median absolute error
mMSE	Mean-based mean squared error
mPB	Mean-based percentage of times better
MSE	Mean squared error
MSEn	Mean squared error over non-zero occurrences
MSR	Mean squared rate
NN	Neural network
PB	Percentage of times better
PIS	Periods in stock
RGRMSE	Relative geometric root mean squared error
RMSE	Root mean squared error
RMSSE	Root mean squared scaled error
sAPIS	Scaled absolute periods in stock
SBA	Syntetos–Boylan approximation
SES	Simple exponential smoothing
SKU	Stock-keeping unit
sMAPE	Symmetric mean absolute percentage error
SPEC	Stock-keeping-oriented prediction error cost
TSB	Teunter, Syntetos, and Babai
VED	Vital–essential–desirable
WRMSSE	Weighted root mean squared scaled error

References

- Kim, M.; Jeong, J.; Bae, S. Demand forecasting based on machine learning for mass customization in smart manufacturing. In Proceedings of the 2019 International Conference on Data Mining and Machine Learning, Hong Kong, China, 28–30 April 2019; pp. 6–11.
- Moon, S. Predicting the Performance of Forecasting Strategies for Naval Spare Parts Demand: A Machine Learning Approach. *Manag. Sci. Financ. Eng.* **2013**, *19*, 1–10. [\[CrossRef\]](#)
- Williams, T. Stock control with sporadic and slow-moving demand. *J. Oper. Res. Soc.* **1984**, *35*, 939–948. [\[CrossRef\]](#)
- Johnston, F.; Boylan, J.E. Forecasting for items with intermittent demand. *J. Oper. Res. Soc.* **1996**, *47*, 113–121. [\[CrossRef\]](#)
- Syntetos, A.A.; Boylan, J.E.; Croston, J. On the categorization of demand patterns. *J. Oper. Res. Soc.* **2005**, *56*, 495–503. [\[CrossRef\]](#)
- Amin-Naseri, M.R.; Tabar, B.R. Neural network approach to lumpy demand forecasting for spare parts in process industries. In Proceedings of the 2008 International Conference on Computer and Communication Engineering, Kuala Lumpur, Malaysia, 13–15 May 2008; IEEE: New York, NY, USA, 2008; pp. 1378–1382.
- Mukhopadhyay, S.; Solis, A.O.; Gutierrez, R.S. The accuracy of non-traditional versus traditional methods of forecasting lumpy demand. *J. Forecast.* **2012**, *31*, 721–735. [\[CrossRef\]](#)
- Petropoulos, F.; Kourentzes, N. Forecast combinations for intermittent demand. *J. Oper. Res. Soc.* **2015**, *66*, 914–924. [\[CrossRef\]](#)
- Bartezzaghi, E.; Kalchschmidt, M. The Impact of Aggregation Level on Lumpy Demand Management. In *Service Parts Management: Demand Forecasting and Inventory Control*; Springer: London, UK, 2011; pp. 89–104. [\[CrossRef\]](#)
- Babai, M.Z.; Syntetos, A.; Teunter, R. Intermittent demand forecasting: An empirical study on accuracy and the risk of obsolescence. *Int. J. Prod. Econ.* **2014**, *157*, 212–219. [\[CrossRef\]](#)
- Johnston, F.; Boylan, J.E.; Shale, E.A. An examination of the size of orders from customers, their characterisation and the implications for inventory control of slow moving items. *J. Oper. Res. Soc.* **2003**, *54*, 833–837. [\[CrossRef\]](#)
- Davies, R. Briefing Industry 4.0 Digitalisation for productivity and growth. In *EPRS | European Parliamentary Research Service*; European Parliament: Strasbourg, France, 2015.
- Glaser, B.S. *Made in China 2025 and the Future of American Industry*; Center for Strategic International Studies: Washington, DC, USA, 2019.
- Yang, F.; Gu, S. Industry 4.0, a revolution that requires technology and national strategies. *Complex Intell. Syst.* **2021**, *7*, 1311–1325. [\[CrossRef\]](#)
- Syntetos, A.A.; Babai, Z.; Boylan, J.E.; Kolassa, S.; Nikolopoulos, K. Supply chain forecasting: Theory, practice, their gap and the future. *Eur. J. Oper. Res.* **2016**, *252*, 1–26. [\[CrossRef\]](#)
- Croston, J.D. Forecasting and stock control for intermittent demands. *J. Oper. Res. Soc.* **1972**, *23*, 289–303. [\[CrossRef\]](#)

17. Brühl, B.; Hülsmann, M.; Borscheid, D.; Friedrich, C.M.; Reith, D. A sales forecast model for the german automobile market based on time series analysis and data mining methods. In Proceedings of the Industrial Conference on Data Mining, Leipzig, Germany, 20–22 July 2009; Springer: Berlin/Heidelberg, Germany, 2009; pp. 146–160.
18. Wang, F.K.; Chang, K.K.; Tzeng, C.W. Using adaptive network-based fuzzy inference system to forecast automobile sales. *Expert Syst. Appl.* **2011**, *38*, 10587–10593. [\[CrossRef\]](#)
19. Sharma, R.; Sinha, A.K. Sales forecast of an automobile industry. *Int. J. Comput. Appl.* **2012**, *53*, 25–28. [\[CrossRef\]](#)
20. Gao, J.; Xie, Y.; Cui, X.; Yu, H.; Gu, F. Chinese automobile sales forecasting using economic indicators and typical domestic brand automobile sales data: A method based on econometric model. *Adv. Mech. Eng.* **2018**, *10*, 1687814017749325. [\[CrossRef\]](#)
21. Salinas, D.; Flunkert, V.; Gasthaus, J.; Januschowski, T. DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *Int. J. Forecast.* **2020**, *36*, 1181–1191. [\[CrossRef\]](#)
22. Bandara, K.; Bergmeir, C.; Smyl, S. Forecasting across time series databases using recurrent neural networks on groups of similar series: A clustering approach. *Expert Syst. Appl.* **2020**, *140*, 112896. [\[CrossRef\]](#)
23. Bradley, A.P. The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognit.* **1997**, *30*, 1145–1159. [\[CrossRef\]](#)
24. Hyndman, R.J. Another look at forecast-accuracy metrics for intermittent demand. *Foresight Int. J. Appl. Forecast.* **2006**, *4*, 43–46.
25. Martin, D.; Spitzer, P.; Kühl, N. A New Metric for Lumpy and Intermittent Demand Forecasts: Stock-keeping-oriented Prediction Error Costs. *arXiv* **2020**, arXiv:2004.10537.
26. Syntetos, A.A.; Boylan, J.E. The accuracy of intermittent demand estimates. *Int. J. Forecast.* **2005**, *21*, 303–314. [\[CrossRef\]](#)
27. Teunter, R.; Sani, B. On the bias of Croston’s forecasting method. *Eur. J. Oper. Res.* **2009**, *194*, 177–183. [\[CrossRef\]](#)
28. Nasiri Pour, A.; Rostami-Tabar, B.; Rahimzadeh, A. *A Hybrid Neural Network and Traditional Approach for Forecasting Lumpy Demand*; World Academy of Science, Engineering and Technology: Paris, France, 2008; Volume 2.
29. Willemain, T.R.; Smart, C.N.; Schwarz, H.F. A new approach to forecasting intermittent demand for service parts inventories. *Int. J. Forecast.* **2004**, *20*, 375–387. [\[CrossRef\]](#)
30. Nikolopoulos, K.; Syntetos, A.A.; Boylan, J.E.; Petropoulos, F.; Assimakopoulos, V. An aggregate–disaggregate intermittent demand approach (ADIDA) to forecasting: An empirical proposition and analysis. *J. Oper. Res. Soc.* **2011**, *62*, 544–554. [\[CrossRef\]](#)
31. Lolli, F.; Gamberini, R.; Regattieri, A.; Balugani, E.; Gatos, T.; Gucci, S. Single-hidden layer neural networks for forecasting intermittent demand. *Int. J. Prod. Econ.* **2017**, *183*, 116–128. [\[CrossRef\]](#)
32. Hasni, M.; Aguir, M.; Babai, M.Z.; Jemai, Z. On the performance of adjusted bootstrapping methods for intermittent demand forecasting. *Int. J. Prod. Econ.* **2019**, *216*, 145–153. [\[CrossRef\]](#)
33. Flores, B.E.; Whybark, D.C. Multiple criteria ABC analysis. *Int. J. Oper. Prod. Manag.* **1986**, *6*, 38–46. [\[CrossRef\]](#)
34. Mitra, S.; Reddy, M.S.; Prince, K. Inventory control using FSN analysis—A case study on a manufacturing industry. *Int. J. Innov. Sci. Eng. Technol.* **2015**, *2*, 322–325.
35. Scholz-Reiter, B.; Heger, J.; Meinecke, C.; Bergmann, J. Integration of demand forecasts in ABC-XYZ analysis: Practical investigation at an industrial company. *Int. J. Product. Perform. Manag.* **2012**, *61*, 445–451. [\[CrossRef\]](#)
36. Botter, R.; Fortuin, L. Stocking strategy for service parts—a case study. *Int. J. Oper. Prod. Manag.* **2000**, *20*, 656–674. [\[CrossRef\]](#)
37. Nallusamy, S. Performance Measurement on Inventory Management and Logistics Through Various Forecasting Techniques. *Int. J. Perform. Eng.* **2021**, *17*, 216–228.
38. Eaves, A.H.; Kingsman, B.G. Forecasting for the ordering and stock-holding of spare parts. *J. Oper. Res. Soc.* **2004**, *55*, 431–437. [\[CrossRef\]](#)
39. Syntetos, A.A.; Boylan, J.E. On the bias of intermittent demand estimates. *Int. J. Prod. Econ.* **2001**, *71*, 457–466. [\[CrossRef\]](#)
40. Syntetos, A.; Boylan, J. Correcting the bias in forecasts of intermittent demand. In Proceedings of the 19th International Symposium on Forecasting, Washington, DC, USA, 27–30 June 1999.
41. Arzi, Y.; Bukchin, J.; Masin, M. An efficiency frontier approach for the design of cellular manufacturing systems in a lumpy demand environment. *Eur. J. Oper. Res.* **2001**, *134*, 346–364. [\[CrossRef\]](#)
42. Wallström, P.; Segerstedt, A. Evaluation of forecasting error measurements and techniques for intermittent demand. *Int. J. Prod. Econ.* **2010**, *128*, 625–636. [\[CrossRef\]](#)
43. Shenstone, L.; Hyndman, R.J. Stochastic models underlying Croston’s method for intermittent demand forecasting. *J. Forecast.* **2005**, *24*, 389–402. [\[CrossRef\]](#)
44. Levén, E.; Segerstedt, A. Inventory control with a modified Croston procedure and Erlang distribution. *Int. J. Prod. Econ.* **2004**, *90*, 361–367. [\[CrossRef\]](#)
45. Teunter, R.H.; Syntetos, A.A.; Babai, M.Z. Intermittent demand: Linking forecasting to inventory obsolescence. *Eur. J. Oper. Res.* **2011**, *214*, 606–615. [\[CrossRef\]](#)
46. Vasumathi, B.; Saradha, A. Enhancement of intermittent demands in forecasting for spare parts industry. *Indian J. Sci. Technol.* **2015**, *8*, 1–8. [\[CrossRef\]](#)
47. Prestwich, S.D.; Tarim, S.A.; Rossi, R.; Hnich, B. Forecasting intermittent demand by hyperbolic-exponential smoothing. *Int. J. Forecast.* **2014**, *30*, 928–933. [\[CrossRef\]](#)
48. Türkmen, A.C.; Januschowski, T.; Wang, Y.; Cemgil, A.T. Forecasting intermittent and sparse time series: A unified probabilistic framework via deep renewal processes. *PLoS ONE* **2021**, *16*, e0259764. [\[CrossRef\]](#)

49. Chua, W.K.W.; Yuan, X.M.; Ng, W.K.; Cai, T.X. Short term forecasting for lumpy and non-lumpy intermittent demands. In Proceedings of the 2008 6th IEEE International Conference on Industrial Informatics, Daejeon, Korea, 13–16 July IEEE: New York, NY, USA, 2008; pp. 1347–1352.
50. Wright, D.J. Forecasting data published at irregular time intervals using an extension of Holt's method. *Manag. Sci.* **1986**, *32*, 499–510. [\[CrossRef\]](#)
51. Altay, N.; Rudisill, F.; Litteral, L.A. Adapting Wright's modification of Holt's method to forecasting intermittent demand. *Int. J. Prod. Econ.* **2008**, *111*, 389–408. [\[CrossRef\]](#)
52. Sani, B.; Kingsman, B.G. Selecting the best periodic inventory control and demand forecasting methods for low demand items. *J. Oper. Res. Soc.* **1997**, *48*, 700–713. [\[CrossRef\]](#)
53. Ghobbar, A.A.; Friend, C.H. Evaluation of forecasting methods for intermittent parts demand in the field of aviation: A predictive model. *Comput. Oper. Res.* **2003**, *30*, 2097–2114. [\[CrossRef\]](#)
54. Chatfield, D.C.; Hayya, J.C. All-zero forecasts for lumpy demand: A factorial study. *Int. J. Prod. Res.* **2007**, *45*, 935–950. [\[CrossRef\]](#)
55. Gutierrez, R.S.; Solis, A.O.; Mukhopadhyay, S. Lumpy demand forecasting using neural networks. *Int. J. Prod. Econ.* **2008**, *111*, 409–420. [\[CrossRef\]](#)
56. Rosenblatt, F. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychol. Rev.* **1958**, *65*, 386. [\[CrossRef\]](#)
57. Huang, G.B.; Zhu, Q.Y.; Siew, C.K. Extreme learning machine: Theory and applications. *Neurocomputing* **2006**, *70*, 489–501. [\[CrossRef\]](#)
58. Hotta, L.; Neto, J.C. The effect of aggregation on prediction in autoregressive integrated moving-average models. *J. Time Ser. Anal.* **1993**, *14*, 261–269. [\[CrossRef\]](#)
59. Souza, L.R.; Smith, J. Effects of temporal aggregation on estimates and forecasts of fractionally integrated processes: A Monte-Carlo study. *Int. J. Forecast.* **2004**, *20*, 487–502. [\[CrossRef\]](#)
60. Athanasopoulos, G.; Hyndman, R.J.; Song, H.; Wu, D.C. The tourism forecasting competition. *Int. J. Forecast.* **2011**, *27*, 822–844. [\[CrossRef\]](#)
61. Rostami-Tabar, B.; Babai, M.Z.; Syntetos, A.; Ducq, Y. Demand forecasting by temporal aggregation. *Nav. Res. Logist. (NRL)* **2013**, *60*, 479–498. [\[CrossRef\]](#)
62. Rostami-Tabar, B.; Babai, M.Z.; Ali, M.; Boylan, J.E. The impact of temporal aggregation on supply chains with ARMA (1, 1) demand processes. *Eur. J. Oper. Res.* **2019**, *273*, 920–932. [\[CrossRef\]](#)
63. Kourentzes, N.; Petropoulos, F. Forecasting with multivariate temporal aggregation: The case of promotional modelling. *Int. J. Prod. Econ.* **2016**, *181*, 145–153. [\[CrossRef\]](#)
64. Hua, Z.; Zhang, B.; Yang, J.; Tan, D. A new approach of forecasting intermittent demand for spare parts inventories in the process industries. *J. Oper. Res. Soc.* **2007**, *58*, 52–61. [\[CrossRef\]](#)
65. Petropoulos, F.; Kourentzes, N.; Nikolopoulos, K. Another look at estimators for intermittent demand. *Int. J. Prod. Econ.* **2016**, *181*, 154–161. [\[CrossRef\]](#)
66. Bartezzaghi, E.; Verganti, R.; Zotteri, G. A simulation framework for forecasting uncertain lumpy demand. *Int. J. Prod. Econ.* **1999**, *59*, 499–510. [\[CrossRef\]](#)
67. Verganti, R. Order overplanning with uncertain lumpy demand: A simplified theory. *Int. J. Prod. Res.* **1997**, *35*, 3229–3248. [\[CrossRef\]](#)
68. Bartezzaghi, E.; Verganti, R.; Zotteri, G. Measuring the impact of asymmetric demand distributions on inventories. *Int. J. Prod. Econ.* **1999**, *60*, 395–404. [\[CrossRef\]](#)
69. Zotteri, G. The impact of distributions of uncertain lumpy demand on inventories. *Prod. Plan. Control* **2000**, *11*, 32–43. [\[CrossRef\]](#)
70. Willemain, T.R.; Smart, C.N.; Shocker, J.H.; DeSautels, P.A. Forecasting intermittent demand in manufacturing: A comparative evaluation of Croston's method. *Int. J. Forecast.* **1994**, *10*, 529–538. [\[CrossRef\]](#)
71. Syntetos, A. Forecasting of Intermittent Demand. Ph.D. Thesis, Brunel University Uxbridge, London, UK, 2001.
72. Anderson, C.J. Forecasting Demand for Optimal Inventory with Long Lead Times: An Automotive Aftermarket Case Study. Ph.D. Thesis, University of Missouri-Saint Louis, Saint Louis, MO, USA, 2021.
73. Teunter, R.H.; Duncan, L. Forecasting intermittent demand: A comparative study. *J. Oper. Res. Soc.* **2009**, *60*, 321–329. [\[CrossRef\]](#)
74. Hemeimat, R.; Al-Qatawneh, L.; Arafeh, M.; Masoud, S. Forecasting spare parts demand using statistical analysis. *Am. J. Oper. Res.* **2016**, *6*, 113. [\[CrossRef\]](#)
75. Kourentzes, N. On intermittent demand model optimisation and selection. *Int. J. Prod. Econ.* **2014**, *156*, 180–190. [\[CrossRef\]](#)
76. Prestwich, S.; Rossi, R.; Armagan Tarim, S.; Hnich, B. Mean-based error measures for intermittent demand forecasting. *Int. J. Prod. Res.* **2014**, *52*, 6782–6791. [\[CrossRef\]](#)
77. Makridakis, S.; Spiliotis, E.; Assimakopoulos, V. The M5 Accuracy Competition: Results, Findings and Conclusions. 2020. Available online: https://www.researchgate.net/publication/344487258_The_M5_Accuracy_competition_Results_findings_and_conclusions (accessed on 25 May 2022).
78. Syntetos, A.A.; Boylan, J.E. On the variance in intermittent demand estimates. *Int. J. Prod. Econ.* **2010**, *128*, 546–555. [\[CrossRef\]](#)
79. Regattieri, A.; Gamberi, M.; Gamberini, R.; Manzini, R. Managing lumpy demand for aircraft spare parts. *J. Air Transp. Manag.* **2005**, *11*, 426–431. [\[CrossRef\]](#)
80. Quintana, R.; Leung, M. Adaptive exponential smoothing versus conventional approaches for lumpy demand forecasting: Case of production planning for a manufacturing line. *Int. J. Prod. Res.* **2007**, *45*, 4937–4957. [\[CrossRef\]](#)

81. Amirkolaii, K.N.; Baboli, A.; Shahzad, M.; Tonadre, R. Demand forecasting for irregular demands in business aircraft spare parts supply chains by using artificial intelligence (AI). *IFAC-PapersOnLine* **2017**, *50*, 15221–15226. [[CrossRef](#)]
82. Gomez, G.C.G. *Lumpy Demand Characterization and Forecasting Performance Using Self-Adaptive Forecasting Models and Kalman Filter*; The University of Texas at El Paso: Texas, TX, USA, 2008.
83. Kiefer, D.; Grimm, F.; Bauer, M.; Van Dinther, C. Demand forecasting intermittent and lumpy time series: Comparing statistical, machine learning and deep learning methods. In Proceedings of the 54th Hawaii International Conference on System Sciences, Virtual, 4–9 January, 2021; p. 1425.
84. Nikolopoulos, K. We need to talk about intermittent demand forecasting. *Eur. J. Oper. Res.* **2021**, *291*, 549–559. [[CrossRef](#)]
85. Laptev, N.; Yosinski, J.; Li, L.E.; Smyl, S. Time-series extreme event forecasting with neural networks at uber. In Proceedings of the International Conference on Machine Learning, Ho Chi Minh, Vietnam, 13–16 January 2017; Volume 34, pp. 1–5.
86. Rožanec, J.M.; Kažič, B.; Škrjanc, M.; Fortuna, B.; Mladenčić, D. Automotive OEM demand forecasting: A comparative study of forecasting algorithms and strategies. *Appl. Sci.* **2021**, *11*, 6787. [[CrossRef](#)]
87. Lowas III, A.F.; Ciarallo, F.W. Reliability and operations: Keys to lumpy aircraft spare parts demands. *J. Air Transp. Manag.* **2016**, *50*, 30–40. [[CrossRef](#)]
88. Prokhorenkova, L.; Gusev, G.; Vorobev, A.; Dorogush, A.V.; Gulin, A. CatBoost: Unbiased boosting with categorical features. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 3–8 December 2018; pp. 6638–6648.
89. Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.Y. Lightgbm: A highly efficient gradient boosting decision tree. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 3146–3154.
90. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
91. Stone, M. Cross-validatory choice and assessment of statistical predictions. *J. R. Stat. Soc. Ser. B (Methodol.)* **1974**, *36*, 111–133. [[CrossRef](#)]
92. Song, H.; Perello-Nieto, M.; Santos-Rodriguez, R.; Kull, M.; Flach, P. Classifier Calibration: How to assess and improve predicted class probabilities: A survey. *arXiv* **2021**, arXiv:2112.10327.

Chapter 3

Probability Calibration

We have described a novel approach to forecasting lumpy and intermittent demand in Chapter 2. Given that lumpy and intermittent demand have a sparse demand occurrence, the approach features two models to forecast demand event occurrence and demand quantity. When predicting demand occurrence, machine learning models provide a predictive score. Such predictive scores are usually not calibrated: each machine learning algorithm creates a characteristic predictive score distribution. A calibrator model can be used to bridge the gap between predicted scores and probability scores. A perfect calibrator would output scores such that, given a particular class, the score would indicate the probability of occurrence of that class. The better the calibrator, the more reliable the probability scores. Building high-quality calibrators for the lumpy and intermittent demand event models is of utmost importance. Such calibrators provide a probabilistic interpretation of forecasted demand events, which can be useful to decision-makers.

Building high-quality calibrators requires appropriate metrics to determine the quality of the calibration. While some calibration metrics have been widely adopted, multiple flaws are described in the literature. Furthermore, research on calibrators and calibration metrics has recently been a topic of tutorials at prestigious machine learning conferences (e.g., "*How to assess and improve classifier confidence and uncertainty*" tutorial at the ECML-PKDD 2020 conference¹). Probability calibration is receiving increasing attention, given they enable better interpretability of classification models.

This chapter comprises a journal paper, an amendment to the results, and a set of complementary results devoted to the visual inspection problem while introducing a new set of measures for probability calibration. The paper is titled *Active Learning and Novel Model Calibration Measurements for Automated Visual Inspection in Manufacturing* and was published in the Journal of Intelligent Manufacturing (Springer) in 2023 (Q1 - IF 7.136) [35]. The paper first introduces how machine learning can alleviate manual visual inspection based on data provided by *Philips Consumer Lifestyle BV*. *Philips Consumer Lifestyle BV* is located in Drachten, The Netherlands, and is one of Philips' biggest development and production centers in Europe. A second part of the paper is devoted to probability calibration. The paper introduces multiple new metrics that can be used to assess calibration quality. Among the advantages of such metrics is the ability (a) to measure the quality of the calibrator even when no prediction scores exist in the bin, (b) to consider multiple target calibration scenarios (e.g., consider calibrating the whole probability range or only a range of interest), and (c) measure the quality of calibration even when no ground truth is available and under specific assumptions, such as a lack of concept drift.

When measuring the calibration quality when no ground truth is available, we con-

¹*How to assess and improve classifier confidence and uncertainty*. Tutorial at the ECML-PKDD 2020 conference: <https://classifier-calibration.github.io/>

sider the predicted labels of the classifier as target labels (see Fig. 3.1). When a classifier performs perfectly, the predicted labels match those any human annotator provides. Conversely, when the classifier’s performance is imperfect, the imperfections can be considered as labeling noise.

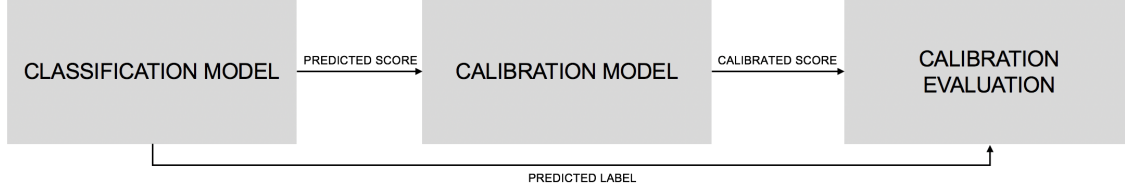


Figure 3.1: The classification model issues a prediction score and a label. The prediction score is used by the calibration model to issue a calibrated score. The calibrated score and the label predicted by the classification model are used to assess the calibrator’s performance.

We consider the contribution regarding the calibration metrics relevant given the proposed metrics overcome limitations of the Expected Calibration Error (ECE), a widely adopted calibration metric. Furthermore, we demonstrate how leveraging information about the ideal score distribution of a perfect calibrator; we can estimate the quality of calibration when no ground truth is available. This particular characteristic is highly relevant when monitoring models deployed in production environments. Furthermore, our research shows that the proposed metrics better capture the calibration information otherwise visually available through calibration plots.

The paper proposes the Additive Probability Calibration Score (APCS) and the Weighted Additive Probability Calibration Score (wAPCS). Both metrics could be reformulated by introducing a weighting factor λ , so that the component evaluating how accurate the classifier is and the term related to the calibration quality could be weighted in some arbitrary proportion. Both metrics, weighting components by a λ factor, are presented in Eq. 3.1 and Eq. 3.2.

$$APCS = \lambda \cdot K_{AUCROC} + (1 - \lambda) \cdot APCS_W \quad (3.1)$$

Equation 3.1: APCS metric definition.

$$wAPCS = \lambda \cdot K_{AUCROC} + (1 - \lambda) \cdot wAPCS_W \quad (3.2)$$

Equation 3.2: wAPCS metric definition.

Finally, we must remark that while in our work, we present three sample reference histograms that could be used to assess the quality of a classifier, our metrics can be contrasted against any reference histogram that mirrors the calibration quality requirements of a particular use case.

After the paper was published, we noticed some computation errors, which did not affect the manuscript’s conclusions. We wrote to the editor asking whether they could be amended, but this was not possible according to the journal policy. We, therefore, publish the amended results and updated analysis in Section 3.1.



Active learning and novel model calibration measurements for automated visual inspection in manufacturing

Jože M. Rožanec^{1,2,5} · Luka Bizjak² · Elena Trajkova³ · Patrik Zajec² · Jelle Keizer⁴ · Blaž Fortuna⁵ · Dunja Mladenec²

Received: 14 September 2022 / Accepted: 16 February 2023
© The Author(s) 2023

Abstract

Quality control is a crucial activity performed by manufacturing enterprises to ensure that their products meet quality standards and avoid potential damage to the brand's reputation. The decreased cost of sensors and connectivity enabled increasing digitalization of manufacturing. In addition, artificial intelligence enables higher degrees of automation, reducing overall costs and time required for defect inspection. This research compares three active learning approaches, having single and multiple oracles, to visual inspection. Six new metrics are proposed to assess the quality of calibration without the need for ground truth. Furthermore, this research explores whether existing calibrators can improve performance by leveraging an approximate ground truth to enlarge the calibration set. The experiments were performed on real-world data provided by *Philips Consumer Lifestyle BV*. Our results show that the explored active learning settings can reduce the data labeling effort by between three and four percent without detriment to the overall quality goals, considering a threshold of $p = 0.95$. Furthermore, the results show that the proposed calibration metrics successfully capture relevant information otherwise available to metrics used up to date only through ground truth data. Therefore, the proposed metrics can be used to estimate the quality of models' probability calibration without committing to a labeling effort to obtain ground truth data.

Keywords Active learning · Probability calibration · Artificial intelligence · Machine learning · Smart manufacturing · Automated visual inspection

Introduction

Quality control is one of the key parts of the manufacturing process, which comprehends inspection, testing, and identification to ensure the manufactured products comply with specific standards and specifications (Kurniati et al., 2015; Wuest et al., 2014; Yang et al., 2020). For example, the inspection tasks aim to determine whether a specific part features assembly integrity, surface finish, and adequate geometric dimensions (Newman & Jain, 1995). In addition, product quality is key to the business since it (i) builds trust with the customers, (ii) boosts customer loyalty, and (iii) reinforces the brand reputation.

Jože M. Rožanec
joze.rozanec@ijs.si

Luka Bizjak
luka.bizjak@ijs.si

Elena Trajkova
trajkova.elena.00@gmail.com

Patrik Zajec
patrik.zajec@ijs.si

Jelle Keizer
jelle.keizer@philips.com

Blaž Fortuna
blaz.fortuna@qlector.com

Dunja Mladenec
dunja.mladenec@ijs.si

¹ Jožef Stefan International Postgraduate School, Jamova 39, 1000 Ljubljana, Slovenia

² Jožef Stefan Institute, Jamova 39, 1000 Ljubljana, Slovenia

³ Faculty of Electrical Engineering, University of Ljubljana, Tržaška c. 25, 1000 Ljubljana, Slovenia

⁴ Philips Consumer Lifestyle BV, Oliemolenstraat 5, Drachten, The Netherlands

⁵ Qlector d.o.o., Rovšnikova 7, 1000 Ljubljana, Slovenia

One such quality inspection activity is the visual inspection, considered a bottleneck activity in some instances (Zheng et al., 2020). Visual inspection is associated with many challenges. Some visual inspections require a substantial amount of reasoning capability, visual abilities, and specialization (Newman & Jain, 1995). Furthermore, reliance on humans to perform such tasks can affect the scalability and quality of the inspection. When considering scalability, human inspection requires training inspectors to develop inspection skills; their inspection execution tends to be slower when compared to machines, they fatigue over time and can become absent at work (due to sickness or other motives) (Selvi & Nasira, 2017; Vergara-Villegas et al., 2014). The quality of inspection is usually affected by the inherent subjectiveness of each human inspector, the task complexity, the job design, the working environment, the inspectors' experience, well-being, and motivation, and the management's support and communication (Cullinane et al., 2013; Kujawińska et al., 2016; See, 2012). Manual visual inspection's scalability and quality shortcomings can be addressed through an automated visual inspection.

Automated visual inspection can be realized with Machine Learning models. Technological advances [e.g., Internet of Things or Artificial Intelligence (Rai et al., 2021; Zheng et al., 2021)], and trends in manufacturing [e.g., the Industry 4.0 and Industry 5.0 paradigms (Barari et al., 2021; Rožanec et al., 2022)] have enabled the timely collection of data and foster the use of machine learning models to automate manufacturing tasks while reshaping the role of the worker (Carvajal Soto et al., 2019; Chouchene et al., 2020). Automated visual inspection was applied in several use cases in the past (Beltrán-González et al., 2020; Duan et al., 2012; Jiang & Wong, 2018; Villalba-Diez et al., 2019). Nevertheless, it is considered that the field is still in its early stages and that artificial intelligence has the potential to revolutionize product inspection (Aggour et al., 2019).

While machine learning models can be trained to determine whether a manufactured piece is defective and do so in an unsupervised or supervised manner, no model is perfect. At least three challenges must be faced: (a) how to improve the models' discriminative capabilities over time, (b) how to calibrate the models' prediction scores into probabilities to enable the use of standardized decision rules (Silva Filho et al., 2021), and (c) how to alleviate the manual labeling effort.

This paper presents our approach to addressing these three challenges as follows. Active learning enhances the classification model to address the first challenge. Pool-based and stream-based settings are compared, considering different active learning sample query strategies across five machine learning algorithms. Platt scaling, a popular probability calibration technique, addresses the second challenge. Finally, two scenarios were considered when addressing the reduction of manual labeling effort: (i) manual inspection of cases

where the machine learning model does not predict with enough confidence and (ii) data labeling to acquire ground truth data for the model calibration. The first scenario was addressed by exploring the usage of multiple oracles and soft labeling to reduce the manual inspection effort. Finally, the second and third scenarios were addressed by approximating the ground truth with models' predictions to calibrate the model. Furthermore, several novel metrics to measure the quality of calibration were proposed. The results confirm that they can measure the quality of such calibration without needing a ground truth.

This work extends our previous research described in paper *Streaming Machine Learning and Online Active Learning for Automated Visual Inspection* (Rožanec et al., 2022). In that paper, research was performed to measure the impact of active learning on streaming algorithms. This paper explored batch and online settings, active learning policies, and oracles. This research overcomes some of the shortcomings of the previous research. First, it does not only consider the models' uncertainty to derive data instances to oracles but also a certain quality acceptance level. Second, it calibrates the machine learning models so that through probability calibration, they issue probabilities rather than predictive scores. Third, it increases the amount of data devoted to active learning to ensure more meaningful results. Finally, it focuses on batch machine learning models (which achieve a greater discriminative performance) and studies them in batch and streaming active learning settings. In addition to the above-mentioned items, multiple metrics were developed to assess the calibration quality of a calibrator. The metrics overcome some shortcomings of widely adopted metrics and enable measuring calibration quality when no ground truth is available. The research was performed on a real-world use case with images provided by *Philips Consumer Lifestyle BV* corporation. The dataset comprises images regarding the printed logo on manufactured shavers. The images are classified into three classes: good prints, double prints, and interrupted prints.

The Area Under the Receiver Operating Characteristic Curve [AUC ROC, see (Bradley, 1997)] was used to evaluate the discriminative capability of the classification models. AUC ROC estimates the quality of the model for all possible cutting thresholds. It is invariant to a priori class probabilities and, therefore, suitable for classification tasks with strong class imbalance. Furthermore, given that the models were evaluated in a multiclass setting, the AUC ROC was computed with a one-vs-rest strategy. Furthermore, the performance of multiple probability calibration approaches was measured through the Estimated Calibration Error (ECE) and several novel metrics proposed in this research.

This paper is organized as follows. Section “[Related work](#)” describes the current state of the art and related works. Section “[Approximate model's probabilities calibra-](#)

tion describes novel metrics proposed for probability calibration and how calibration methods can leverage approximate ground truth to enlarge the calibration set. The main novelty regarding the proposed probabilities calibration metrics is the ability to measure calibration quality without needing a ground truth. Section “Use case” describes the use case, while section “Methodology” provides a detailed description of the methodology followed. Section “Experiments” describes the experiments performed, while section “Results and evaluation” presents and discusses the results obtained. Finally, section “Conclusion” presents the conclusions and outlines future work.

Related work

This section provides a short overview of three topics relevant to this research: (i) the use of machine learning for quality inspection, (ii) active learning, and (iii) probabilities calibration. The following subsections are devoted to each of them.

Machine learning for quality inspection

A comprehensive and reliable quality inspection is indispensable to the manufacturing process, and high inspection volumes turn inspection processes into bottlenecks (Schmitt et al., 2020). Machine Learning has been recognized as a technology that can drive the automation of quality inspection tasks in the industry. Multiple authors report applying it for early prediction of manufacturing outcomes, which can help drop a product that will not meet quality expectations and avoid investment in expensive manufacturing stages. Furthermore, similar predictions can be used to determine whether the product can be repaired and therefore avoid either throwing away a piece to which the manufacturing process was invested or selling a defective piece with the corresponding costs for the company (Weiss et al., 2016). Automated visual inspection refers to image processing techniques for quality control, usually applied in the production line of manufacturing industries (Beltrán-González et al., 2020). It has been successfully applied to determine the end quality of the products. It provides many advantages, such as performing non-contact inspection that is not affected by the type of target, surface, or ambient conditions (e.g., temperature) (Park et al., 2016). In addition, visual inspection systems can perform multiple tasks simultaneously, such as object, texture, or shape classification and defect segmentation, among other inspections. Nevertheless, automated visual inspection is a challenging task given that collecting the dataset is usually expensive, and the methods developed for that purpose are dataset-dependent (Ren et al., 2017).

Jian et al. (2017) considers three approaches that exist toward automated visual inspection: (a) classification, (b) background reconstruction and removal (reconstruct and remove background to find defects in the residual image), and (c) template reference (comparing a template image with a test image). Tsai and Lai (2008) describe how TFT-LCD panels and LCD color filters were inspected by comparing surface segments containing complex periodic patterns. Lin et al. (2019) describes how defect inspection on LED chips was automated using deep Convolutional Neural Networks (CNN). Kang and Liu (2005) successfully applied feed-forward networks to detect surface defects on cold-rolled strips. In the same line, Yun et al. (2014) proposed a novel defect detection algorithm for steel wire rods produced by the hot rolling process. Valavanis and Kosmopoulos (2010) compared multiple machine learning models (Support Vector Machine, Neural Network, and K-nearest neighbors (kNN)) on defect detection in weld images. Park et al. (2016) developed a CNN and compared it to multiple models (particle swarm optimization-imperialist competitive algorithm, Gabor-filter, and random forest with variance-of-variance features) to find defects on silicon wafers, solid paint, pearl paint, fabric, stone, and wood surfaces. Furthermore, Aminzadeh and Kurfess (2019) described how Bayesian classification enabled online quality inspection in a powder-bed additive manufacturing setting. Multiple authors developed machine learning algorithms for visual inspection leveraging feature extraction from pre-trained models (Cohen & Hoshen, 2020; Li et al., 2021; Jezek et al., 2021). While much research was devoted to supervised machine learning methods, unsupervised defect detection was explored by many authors, who explored using Fourier transforms to remove regularities and highlight irregularities (defects) (Aiger & Talbot, 2012) or employed autoencoders to find how a reference image differs from the expected pattern (Mujeeb et al., 2018; Zavrtanik et al., 2021, 2022).

Active learning

Active learning is a subfield of machine learning that studies how an active learner can best identify informative unlabeled instances and requests their labels from some *oracle*. Typical scenarios involve (i) membership query synthesis (a synthetic data instance is generated), (ii) stream-based selective sampling (the unlabeled instances are drawn one at a time, and a decision is made whether a label is requested or the sample is discarded), and (iii) pool-based selective sampling (queries samples from a pool of unlabeled data). Among the frequently used querying strategies are (i) uncertainty sampling (select an unlabeled sample with the highest uncertainty, given a certain metric, or machine-learning model (Lewis & Catlett, 1994)), or (ii) query-by-committee [retrieve the unlabeled sample with the highest disagreement between a set of

forecasting models (*committee*) (Cohn et al., 1994; Settles, 2009)] can be found. More recently, new scenarios have been proposed leveraging reinforcement learning, where an agent learns to select images based on their similarity, and rewards obtained are based on the oracle's feedback (Ren et al., 2020). In addition, it has been demonstrated that ensemble-based active learning can effectively counteract class imbalance through newly labeled image acquisition (Beluch et al., 2018). While active learning reduces the required volume of labeled images, it is also essential to consider that it can produce an incomplete ground truth by missing the annotations of defective parts classified as false negatives and not queried by the active learning strategy (Cordier et al., 2021).

Active learning was successfully applied in manufacturing, but scientific literature remains scarce on this domain (Meng et al., 2020). Some use cases include the automatic optical inspection of printed circuit boards (Dai et al., 2018), media news recommendation in a demand forecasting setting (Zajec et al., 2021), and the identification of the local displacement between two layers on a chip in the semiconductor industry (van Garderen, 2018).

Probabilities calibration

Probabilities denote the likelihood that a particular event will occur and are expressed as a real number between zero and one (Cheeseman, 1985). Many machine learning models output prediction scores which cannot be directly interpreted as probabilities. Therefore, such models can be calibrated (mapped to a known scale with known properties), ensuring the prediction scores are converted to probabilities. Probability calibration aims to provide reliable estimates of the true probability that a sample is a member of a class of interest. Such calibration (a) usually does not decrease the classification accuracy, (b) enables using provides thresholds on the decision rules and therefore minimizes the classification error, (c) ensures decision rules and their maximum posterior probability are fully justified from the theoretical point of view, (d) can be easily adapted to changes in class and cost distributions, and therefore (e) is key to decision-making tasks (Cohen & Goldszmidt, 2004; Song et al., 2021).

The k-class probabilistic classifier is considered well-calibrated if the predicted k-dimensional probability vector has a distribution that approximates the distribution of the test instances. While a single accepted notion of probabilistic calibration exists for binary classifiers, the definition for multiclass settings has multiple nuances. Three kinds of probability calibration are described in the literature for multiclass settings: (i) *confidence calibration* [aims only to calibrate the classifier's most likely predicted class (Song et al., 2021)], (ii) *class-wise calibration* (attempts to calibrate the scores for each class as marginal probabilities), and (iii) *multi-class calibration* (seeks to create an entire vector of predicted prob-

abilities so that for any prediction vector the proportion of classes among all possible instances getting the same prediction, are equal to the probabilities for those classes in the predicted vector).

Multiple probability calibration methods have been proposed in the scientific literature. The post-hoc techniques aim to learn a calibration map for a machine-learning model based on hold-out validation data. In addition, popular calibration methods for binary classifiers include logistic calibration (Platt scaling), isotonic calibration, Beta calibration, temperature calibration, and binning calibration.

Empirical binning builds the calibration map by computing the empirical frequencies within a set of score intervals. It can therefore capture arbitrary prediction score distributions (Kumar et al., 2019). Isotonic regression computes a regression assuming the uncalibrated model has a set of non-decreasing constant segments corresponding to bins of varying widths. Given its non-parametric nature, it avoids a model misfit, and due to the monotonicity assumption, it can find optimal bin edges. Nevertheless, training times and memory consumption can be high on large datasets and give sub-optimal results if the monotonicity assumption is violated. Platt scaling (Platt, 2000) aims to transform prediction scores into probabilities through a logistic regression model, considering a uniform probability vector as the target. While the implementation is straightforward and the training process is fast, it assumes the input values correspond to a real scalar space and restricts the calibration map to a sigmoid shape. Probability calibration trees evolve the concept of Platt scaling, identifying regions of the input space that lead to poor probability calibration and learning different probability calibration models for those regions, achieving better overall performance (Leathart et al., 2017). Beta calibration was designed for probabilistic classifiers. It assumes that the scores of each class can be approximated with two Beta distributions and is implemented as a bivariate logistic regression. Temperature scaling uses a scalar parameter $T > 0$ (where T is considered the temperature) to rescale logit scores before applying a softmax function to achieve recalibrated probabilities with better spread scores between zero and one. It is frequently applied to deep learning models, where the prediction scores are frequently strongly skewed towards one or zero. Furthermore, the method can be applied to generic probabilistic models by transforming the prediction scores with a logit transform (Guo et al., 2017). This enables calculating the score against a reference class and obtaining the ratio against other classes. Nevertheless, the method is not robust in capturing epistemic uncertainty (Ovadia et al., 2019). Finally, the concept of temperature scaling is extended in vector scaling, which considers that a different temperature for each class can be specified, and matrix scaling, which considers a matrix and intercept parameters (Song et al., 2021).

Several metrics and methods were proposed to assess the quality of the calibration. Reliability diagrams plot the observed relative frequency of predicted scores against their values. They, therefore, enable to quickly assess whether the event happens with a relative frequency consistent with the forecasted value (Bröcker & Smith, 2007). On the other hand, validity plots aim to convey the bin frequencies for every bin and therefore provide valuable information regarding mis-calibration bounds (Gupta & Ramdas, 2021). Among the metrics, the binary ECE measures the average gap across all bins in a reliability diagram, weighted by the number of instances in each bin, considering the labeled samples of a test set. In the same line, the binary Maximum Calibration Error computes the maximum gap across all bins in a reliability diagram. The Confidence Estimated Calibration Error measures the average difference between accuracy and average confidence across all bins in a confidence reliability diagram, weighted by the number of instances per bin. A different approach is followed by the Brier score, which measures the mean squared difference between the predicted probability and the actual outcome. While the ECE metric is widely accepted, research has shown that it is subject to shortcomings (Nixon et al., 2019; Posocco & Bonnefoy, 2021). One of such shortcomings is that when using fixed calibration ranges, some bins contain most of the data, resulting in the metric's decreased sharpness. Furthermore, ECE is measured across non-empty bins, failing to account for the overall distribution of positives across the mean predicted probabilities. Measuring probabilistic calibration remains a challenge (Nixon et al., 2019).

While many probability calibration methods and metrics have been developed, most of them were conceived considering probability calibration must be done based on some ground truth. Nevertheless, acquiring data for such ground truth is expensive (requires labeled instances), limits the amount of data seen to build such a probability calibration map, and therefore introduces inaccuracies due to the inherent characteristics of the sample. To address this void, this research proposes labeling each predicted data instance according to the predicted class with the highest score or most likely class if the highest predicted scores are equal. Assuming the classifier could perform with perfect discriminative power in the best case, such labels would equal the ground truth. Furthermore, this research proposes metrics to assess the discrepancy between an ideal probability calibration scenario and the calibrated classifier to measure the quality of probability calibration achieved. By doing so, the calibrators' quality over time can be measured without needing any data labeling for such an assessment. Furthermore, it enables exploring approximate model's probabilities calibration, training a calibrator from a ground truth approximated with predicted labels. This idea is further explained

and developed in section "[Approximate model's probabilities calibration](#)".

Approximate model's probabilities calibration

Towards approximate probability calibration models

This research proposes metrics and an approach to calibrating machine learning prediction scores to probabilities using a ground truth approximation. The approach considers building an initial calibration set, as it is common practice for probability calibration methods. A calibration set has (a) several prediction scores used to perform the probability calibration and (b) the ground truth labels for the corresponding data instances. Using both, a mapping is created between the prediction scores and the probability of a class outcome. Nevertheless, the limited amount of data in the calibration set can impact the fidelity of the calibration. In particular, the distribution of predictive scores between the calibration set and the predictions performed in a production environment can differ.

The final prediction of a calibrated model has at least two sources of error: (a) the classification model, which does not perfectly predict the target class, and (b) the probability calibration technique, which does not produce a perfect probabilistic mapping between the predicted scores and the target class. While (a) directly affects the refinement loss (loss produced by assigning the same probability for instances from different classes), (b) affects the calibration loss (loss due to the difference between the predicted probabilities and observed positive instances for such an output). While metrics and plots exist to assess the quality of the probability calibration, such means require a ground truth to evaluate the probability calibration. While the requirement for a ground truth allows for an exact estimate of the classifier on that particular hold-out data, it has at least two drawbacks: (i) it requires labeling a certain amount of data to perform the evaluation, and (ii) such data may not be representative of current or future data distributions observed in a production environment.

Intuitions behind a calibration without a ground truth

Current scientific literature considers the quality of a model calibration can be measured by comparing, given a fixed class, whether the fraction of positives does correspond to the predicted mean probability of a given classifier. The fraction of positives empirically measures the likelihood of positive class events for the class under consideration within a specific mean probability range (bin). In a well-calibrated model,

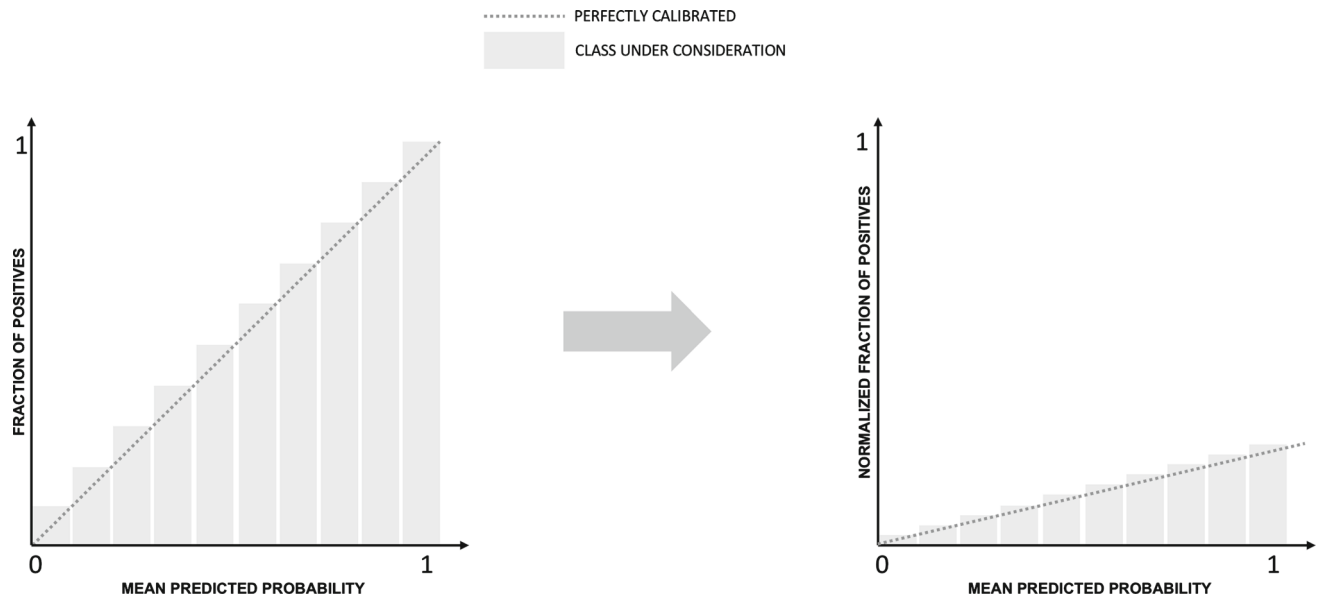


Fig. 1 The figure presents two calibration plots. On the left, the calibration plot shows a perfectly calibrated calibrator (where the fraction of positives for the class under consideration equals the mean predicted

probability). On the right, the same information is presented, but normalizing the values of the plot on the right to ensure the sum of their values equals one

the likelihood of the occurrence of positive class events in a particular bin for the class under consideration matches the mean predicted probability, revealing a linear relationship between the mean predicted probability and the likelihood of the occurrence in that bin of the positive class event for the class considered (see Fig. 1). Furthermore, a perfectly calibrated classifier is only possible for a binary classification problem with no class imbalance. Class imbalance or multiple classes introduce distortions regarding the frequency with which the positive class is observed within a given predicted mean probability range compared to the frequency with the other events occurring within that mean probability range.

For a well-calibrated classifier, each of the predicted classes is expected to behave as shown in Fig. 1. Therefore, while class imbalance or a multi-class setting can introduce distortions to the histogram's shape, the distance to the ideal case could be measured by comparing the histogram shape of a perfectly calibrated model for a given class and the shape of the histogram in the real-world case under consideration. To estimate how close the histograms are from each other, optimal transport is used (Peyré et al., 2019; Villani, 2009). In particular, the Wasserstein distance measures the distance between the two histogram distributions. We consider the Wasserstein distance between the histograms representing the existing calibration and a perfect one. The distance denotes the improvement opportunity regarding the specific calibration model to achieve a perfect calibration (or a desired calibration according to the reference histogram). Nevertheless, the fraction of positives for a given class cannot

be computed when no ground truth is available. Therefore, we reframe the problem so that the goodness of a model calibration can be evaluated even without considering a ground truth.

Considering the information available in Fig. 1 and a particular class j , and considering each prediction regarding class j an event x , we are interested on two types of events: $E_1 = \{x \text{ corresponds to bin } i\}$, and $E_2 = \{x \text{ corresponds to class } j\}$. Furthermore, we are interested in calibrating the model so that the resulting score indicates $p_j(E_2|E_1)$.

Intuition 1: Considering a perfectly calibrated classifier

Let us consider the case of a perfectly calibrated classifier. Given a perfectly calibrated classifier, the fraction of positives for a given class must match the mean predicted probability. The fraction of positives within a certain bin i can be considered the empirical computation of $p_j(E_2|E_1)$. E_1 and E_2 are not independent events, given the probability of belonging to class j should be higher in bins representing a higher mean predicted probability. Therefore, $p_j(E_2|E_1) = \frac{p_j(E_2 \cap E_1)}{p_j(E_1)}$. Considering a balanced binary classification problem, the number of predictions issued for each mean predicted probability range must be equal to verify the symmetry regarding the fraction of positives observed in the mean probability ranges for both classes. Fluctuations regarding the fraction of positives observed in the mean predicted probability ranges translate into an unequal number of

predictions in them and directly impact the quality of the calibration. Based on this observation, given the abovementioned equation, $p_j(E_2|E_1) = \frac{p_j(E_2 \cap E_1)}{p_j(E_1)}$, $p_j(E_1)$ is constant, and can be empirically computed as $p_j(E_1) = \frac{1}{\# \text{ of bins}}$. The number of predictions for a given class j is computed as the count of predictions where the highest predicted value was issued for that class j . While $p_j(E_2 \cap E_1)$ cannot be computed without ground truth, the expected values that must be satisfied for each bin for $p_j(E_2|E_1)$ are known. Therefore, we envision at least two ways to estimate the mismatch between the ideal case and the case under consideration. First, the value of $p_j(E_2 \cap E_1)$ can be inferred based on the expected $p_j(E_2|E_1)$ for a particular bin and the empirical computation of $p_j(E_1)$ to then measure the Wasserstein distance between the resulting distributions. Second, it could be estimated by only considering $p_j(E_1)$ and measuring the Wasserstein distance between the ideal distribution (an equal number of predictions per mean predicted probability range) and the distribution of predictions obtained from the calibrated classifier under consideration (number of predictions per bin, that are empirically measured—usually the amount of predictions is not equal across bins given the calibrated classifier’s imperfection). Each class’s calibration quality could be estimated in both cases by comparing two histograms: the ideal case and the calibration model under consideration. The distance between both distributions computes measures how far the particular calibrator is from a perfectly calibrated case.

While the case above was demonstrated for a balanced binary classification problem, it approximately holds for multiclass settings and cases with class imbalance. In these scenarios, we aim to calibrate each class as perfectly as possible, even though a perfect calibration cannot be achieved. Nevertheless, how well-calibrated each class is against the ideal case can still be assessed by comparing the distributions described above.

Intuition 2: Considering a perfect classifier

Let us consider the case of a perfect classifier. Given a perfect classifier, the prediction equals the ground truth regarding a positive class event for the class under consideration. Therefore, two scenarios are considered: (a) degrade the classifiers’ performance to achieve a calibrated classifier, or (b) spread the predicted values within a specific range so that they emulate particular calibration. It must be noted that while (a) can still satisfy the definition of probability considered for calibration, (b) does not.

For (a), the classifier’s performance must be degraded due to the inherent definition of probabilities used in this problem: the calibration model will ensure a proportion of positive events regarding a class given a mean predicted probability

bin. Therefore, given $n = \text{number of classes}$, the highest predicted value for each class will not issue only data instances of that class above $1/n$. Furthermore, some cases will be lost under the $1/n$ threshold.

On the other hand, for (b), the abovementioned equation $p_j(E_2|E_1) = \frac{p_j(E_2 \cap E_1)}{p_j(E_1)}$ can be considered. It is known that for a perfect classifier, the following is true: $p_j(E_2) = 0$ or $p_j(E_2) = 1$. Furthermore, E_1 and E_2 can be considered dependent events, given $p_j(E_2) = 0$ for bins below a certain threshold, and $p_j(E_2) = 1$ otherwise (see Fig. 2). In addition, the mean predicted probability would not match the fraction of positives, given the classifier is perfect: each prediction perfectly identifies the target class. Therefore, this scenario *per se* violates the idea behind probabilities calibration. Nevertheless, the best approximation towards Fig. 1 would be to achieve an increasing number of predictions per mean predicted probability range (histogram bin) for a specific class. To avoid degrading the models’ discriminative power, such a mapping function will not issue scores below $1/n$, where $n = \text{number of classes}$.

Intuitions materialized

From intuitions to approximate calibrators

To perform model calibration, a function that can map the predictive scores of a machine-learning model to probability scores is required. Ideally, such probability scores would indicate $p_j(E_2|E_1)$. When no ground truth is available, the intuitions described above can be considered to reproduce some scenarios where the resulting probability score distribution can be compared against an ideal probability score distribution. Therefore, we consider labeling the predicted data instances with the class with the highest predicted score. In case two classes hold equal scores, we decide on the most probable one based on the class imbalance observed in the train test. For balanced datasets, the class can be assigned randomly, given no other information exists to guide the decision. The more perfect the classification model, the closer will the assigned labels be to the ground truth. Given data instances with predicted scores and assigned labels, a calibrator can be fitted to map the classifier’s output to a calibrated probability.

To realize the abovementioned calibration without ground truth, at least the following preconditions must be met: (a) no concept drift exists, (b) no covariate shift exists, and (c) the values of the features in the production environment remain within the ranges considered when training the machine learning model.

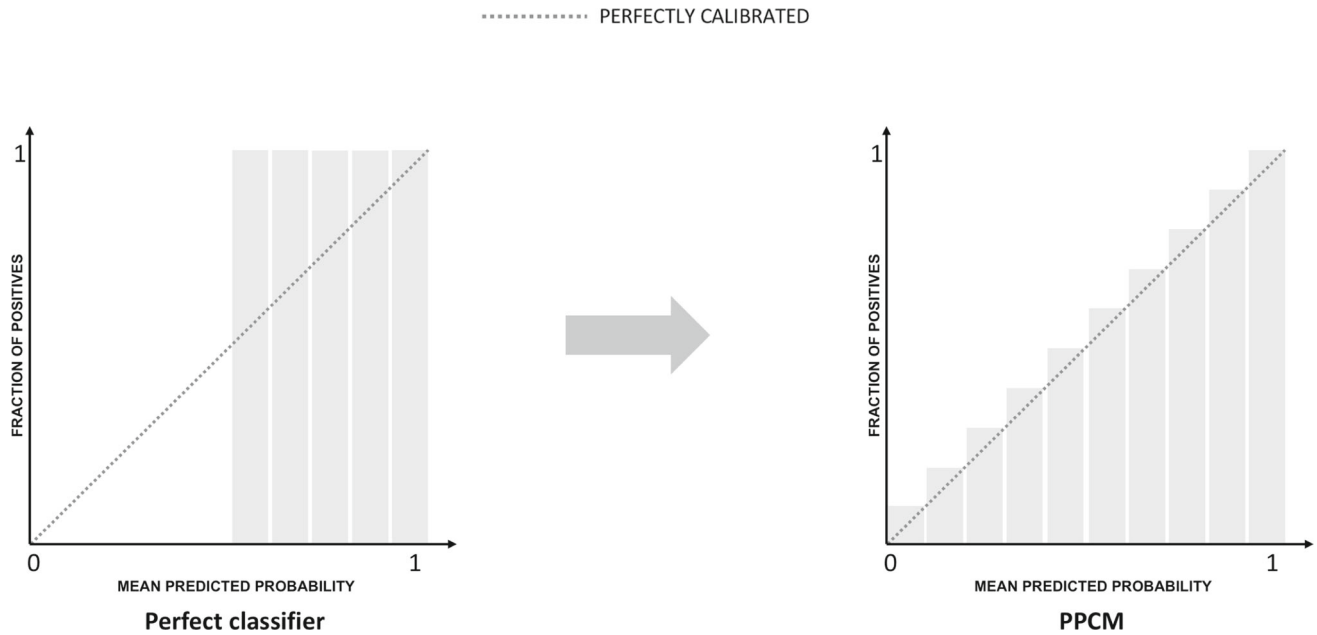


Fig. 2 The figure presents two calibration plots. On the left, the calibration plot shows a perfect binary classifier, while on the right we find a perfectly calibrated binary classifier

From intuitions to metrics

In Sections “[Intuition 1: Considering a perfectly calibrated classifier](#)” and “[Intuition 2: Considering a perfect classifier](#)”, the cases of a perfectly calibrated model and a perfect classifier were considered. While in the case of a perfect classifier, a ground truth is not needed (the predicted labels equal the ground truth), non-perfect classifiers approximate such a ground truth to a certain degree (measured as the classifiers’ performance). Furthermore, regardless of the calibration technique, it was shown that a certain correlation between the calibration quality and the calibration score distribution exists. In particular, it was shown that for each class k a histogram could be computed showing (a) the number of predictions per bin and (b) the proportion of positive class occurrences per mean predicted probability bin. Both could then be compared against ideal cases. A certain advantage of (a) is that it does not require ground truth or ground truth approximation to determine whether some bins are under or over-assigned. While such an imbalance certainly signals a calibration error, the histogram lack information regarding the composition of each bin. In particular, they provide no information on whether the positive class occurrences increase according to the value of the mean predicted probability bin. This can only be measured in (b), comparing all cases against an ideal calibration histogram. For multiclass problems, each class could be compared against such a histogram, and the resulting scores averaged (Fig. 3).

To estimate how close a probability calibration method is w.r.t. the target (ideal) histogram, optimal transport is used

(Peyré et al., 2019; Villani, 2009). In particular, the Wasserstein distance between two histogram distributions is considered: a histogram constructed with the calibrator scores and a histogram corresponding to the ideal scenario. Based on them, we propose a metric that can be used to estimate the quality of calibration of any calibrator given certain ground truth. We name it *Probability Calibration Score* (PCS—see Eq. 1). The proposed metrics issue a value between zero and one: PCS is zero when the model is not calibrated and one when the model is perfectly calibrated. Furthermore, a weighted metric variant can also be considered (wPCS—see Eq. 2), where the proportion of each class among the observed instances weights the Wasserstein distances.

$W_1(h_i, h_{ref})$ is the 1-Wasserstein distance between the histogram h_i and the reference histogram h_{ref} and n is the number of classes.

$$PCS = \sum_{i=1}^n \frac{1 - W_1(h_i, h_{ref})}{n} \quad (1)$$

$W_1(h_i, h_{ref})$ is the 1-Wasserstein distance between the histogram h_i and the reference histogram h_{ref} and w_i is the weight of a particular class. n indicates the number of classes under consideration.

$$wPCS = \sum_{i=1}^n (1 - W_1(h_i, h_{ref})) \cdot w_i \quad (2)$$

To ensure the histograms are comparable, they are normalized, ensuring that the sum of their values equals one. To ensure the Wasserstein distance remains between zero

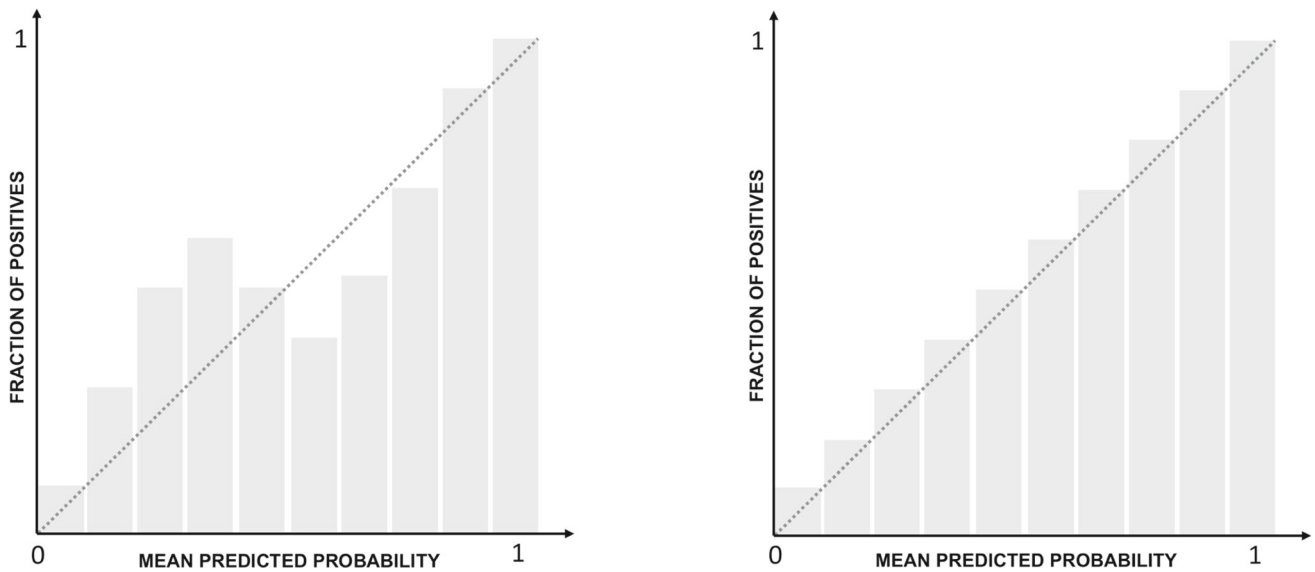


Fig. 3 The figure illustrates two sample histograms: the histogram on the left corresponds to some sub-optimally calibrated classifier. In contrast, the histogram on the right (reference histogram) corresponds to a perfectly calibrated classifier

and one, the distance between both distributions is divided by the distance measured between the worst-case scenario and the reference ideal histogram (see Fig. 4). In Fig. 4, we consider the Wasserstein distance between the case on the left and the distribution of a Perfect Probability Calibration Model (PPCM) to be the highest among possible calibration scenarios.

When assessing the performance of an approximate calibrated model, two errors must be taken into account: (i) the classification error, given the classifier does not perfectly predict the target class (and the ground truth is approximated with such predictions), and (ii) the probability calibration technique, which does not produce a perfect probabilistic mapping between the predicted scores and the (approximated) target class. To measure (i), we choose the AUC ROC metric, which is not affected by the class imbalance. AUC ROC can be computed in a multiclass setting with a one-vs-rest or one-vs-one strategy. We measure it on the test set. We consider (ii) can be measured using the Wasserstein distance, comparing the ideal calibration histogram and a histogram where the proportion of positive class occurrences (given the approximate ground truth) is considered per mean predicted probability bin.

We propose two metrics, which we name Additive Probability Calibration Score (APCS—see Eq. 6) and Multiplicative Probability Calibration Score (MPCS—see Eq. 8). Both summarize the calibrated models' performance, considering the classifier's imperfection (see Eq. 3) and the calibration error incurred due to the lack of ground truth. To ensure the Wasserstein distance remains between zero and one, we compute a normalized histogram, ensuring the area of the entire histogram equals one. The proposed metrics issue a

value between zero and one, and in both cases, the higher the value, the better the model. Furthermore, we also provide a weighted version of both metrics (wAPCS (see Eq. 7) and wMPCS (see Eq. 9)), which aim to weight the Wasserstein distance between the normalized histograms obtained from a calibrator and the ideal histogram with the class weights (see Eqs. 4 and 5 for $APCS_W$ and $wAPCS_W$, and Eqs. 1 and 2 for MPCS and wMPCS).

APCS is zero when the model has no discriminative power and is not calibrated, and one when the model is perfectly calibrated and shows no classification error on the test set. The APCS metric is detailed in Eq. 6.

K is used to measure classifiers' discriminative power. $AUCROC_{Classifier_{test}}$ corresponds to the classifiers' AUC ROC measured on the test set.

$$K_{AUCROC} = |0.5 - AUCROC_{Classifier_{test}}| \quad (3)$$

Component for Wasserstein distance measurement between an ideal calibrator and the calibrator under consideration, as used for the APCS metric.

$$APCS_W = 0.5 \cdot PCS \quad (4)$$

Component for Wasserstein distance measurement between an ideal calibrator and the calibrator under consideration, as used for the wAPCS metric.

$$wAPCS_W = 0.5 \cdot wPCS \quad (5)$$

APCS metric definition.

$$APCS = K_{AUCROC} + APCS_W \quad (6)$$

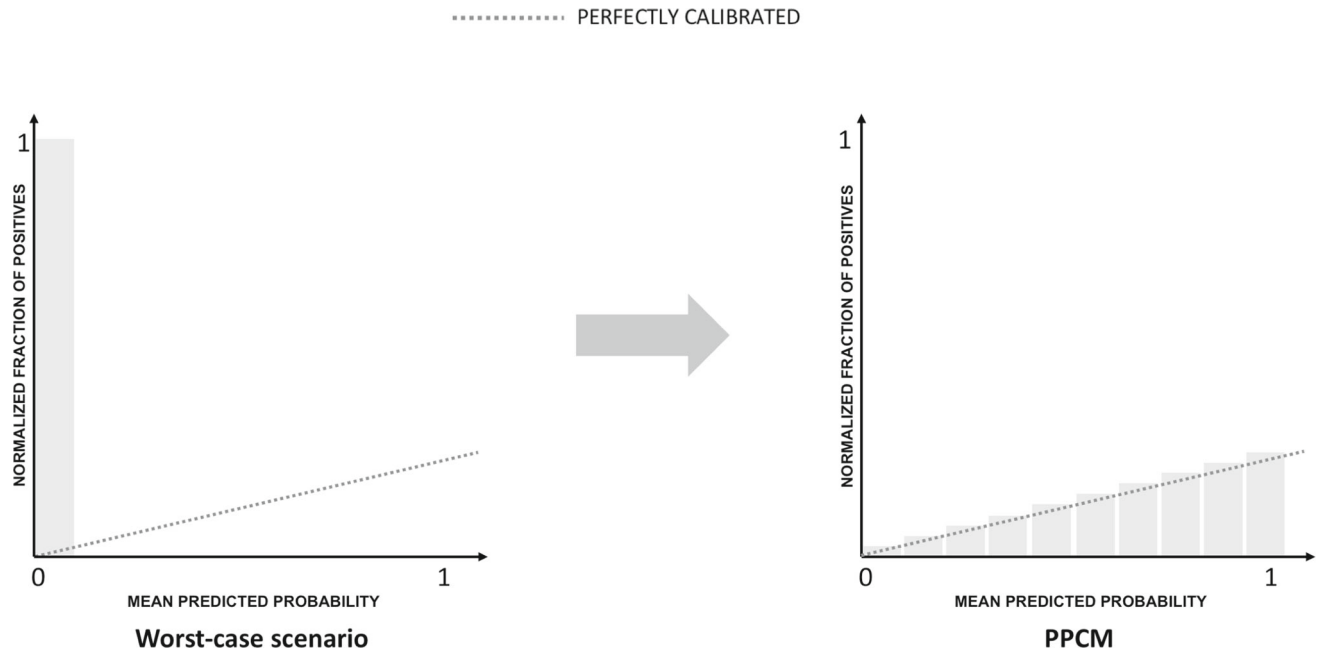


Fig. 4 The figure illustrates two sample calibration plots: the calibration plot on the left corresponds to a calibrated classifier where all positives were assigned to a zero mean predicted probability (worst-case scenario). In contrast, the calibration plot on the right (reference

histogram) corresponds to a perfectly calibrated classifier. Both calibration plots correspond to normalized cases, where the sum of the values equals one

wAPCS metric definition.

$$wAPCS = K_{AUCROC} + wAPCS_W \quad (7)$$

On the other hand, MPCS and wMPCS correspond to zero when (a) the classifiers' predictive ability is no better than random guessing or (b) the Wasserstein distance between histograms is highest (equal to one). Moreover, MPCS and wMPCS correspond to one when (a) the classifiers' predictive ability is perfect, and (b) the calibration is perfect w.r.t. the target histogram h of choice. The MPCS metric is detailed in Eq. 8.

MPCS metric definition

$$MPCS = K_{AUCROC} \cdot PCS \quad (8)$$

MPCS metric definition

$$wMPCS = K_{AUCROC} \cdot wPCS \quad (9)$$

For models' probability calibration, PCS, APCS, and MPCS assume an ideal reference histogram. Three histograms are presented in Fig. 5 corresponding to (a) a Perfect Probability Calibration Model (PPCM), (b) an Almost Perfect Probability Calibration Model (APPCM), and (c) Perfect Classification with Perfect Confidence (PPwPC). While only PPCM can be used for strict probability calibration, the other

two reference histograms measure how far the distributions of the predicted values are from other desired distribution shapes. In particular, APPCM achieves a similar spread of predicted probabilities as PPCM but neglects the segment of predictions below $1/n$ (with $n = \text{number of classes}$), where the classifier would become suboptimal. On the other hand, PPwPC advocates for a classifier where all scores are pushed toward the highest possible score for a given class. This research only considers the PPCM reference histogram to compute the above-described metrics.

Use case

Philips Consumer Lifestyle BV in Drachten, The Netherlands, is one of Philips' biggest development and production centers in Europe. They use cutting-edge production technology to manufacture products ceaselessly. One of their improvement opportunities is related to visual inspection, where they aim to identify when the company logo is not properly printed on the manufactured products. They have multiple printing pad machines, from which the products are handled and inspected on their visual quality and removed if any error is detected. Experts estimate that a fully automated procedure would speed up the process by more than 40%. Currently, there are two defects associated with the printing quality of the logo (see Fig. 6): double prints (the whole logo

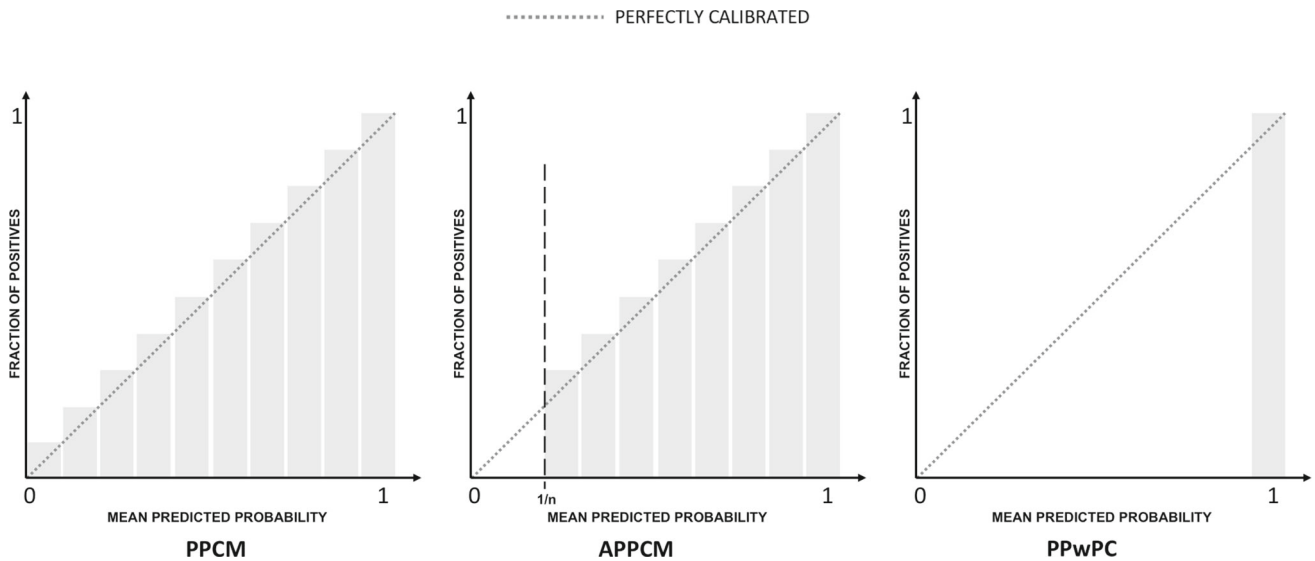


Fig. 5 The figure illustrates three histograms that correspond to ideal cases described in this section: Perfect Probability Calibration Model (PPCM), Almost Perfect Probability Calibration Model (APPCM), and Perfect Classification with Perfect Confidence (PPwPC)



Fig. 6 The images shown above correspond to three possible classes: good (no defect), double print (defective), and interrupted print (defective)

is printed twice with a varying overlap degree) and interrupted prints (the logo displays small non-pigmented areas, similar to scratches).

Machine learning models can be developed to automate the visual inspection procedure (Rippel et al., 2021; Zavrtnik et al., 2022). However, given that such models are imperfect, the manual revision can be used as a fallback to inspect the products about which the uncertainty of the machine learning model exceeds a certain threshold. Such decisions can be made based on simple decision rules, quality policies, and the probability of obtaining a defective product given a particular prediction score. Furthermore, products sent for manual inspection can be prioritized using different criteria to enhance the existing defect detection machine learning model. This research explores the abovementioned capabilities through multiple experiments, building supervised models, leveraging active learning, and comparing six machine learning algorithms. Furthermore, new measures for probability calibration are explored, and experiments are executed to determine whether existing calibration techniques would benefit from enlarging the calibration set with approximate ground truth. The experiments were conducted on a

dataset of 3518 labeled images, all corresponding to manufactured shavers.

Methodology

The research presented in this paper was performed using the Python language, and open source libraries, such as scikit-learn (Buitinck et al., 2013) and netcal (Küppers et al., 2020).

Methodological aspects to evaluate active learning strategies

We frame the automated defect detection as a supervised, multiclass classification problem. A ResNet-18 model (He et al., 2016) was used for feature extraction. 512 values long vectors were extracted for each image obtained from the average pooling layer. To avoid overfitting, the procedure suggested by Hua et al. (2005) was followed by selecting the *top K* features, with $K = \sqrt{N}$, where N is the number of data instances in the train set. Features' relevance was assessed considering the mutual information score, which measures any relationship between random variables. It is considered that the mutual information score is not sensitive to feature transformations if these transformations are invertible and differentiable in the feature space or preserve the order of the original elements of the feature vectors (Vergara & Estévez, 2014) (Fig. 7).

To evaluate the models' and active learning scenarios' performance, a stratified k -fold cross validation (Zeng & Martinez, 2000) was applied, considering $k=10$ based on recommendations by Kuhn and Johnson (2013). One fold

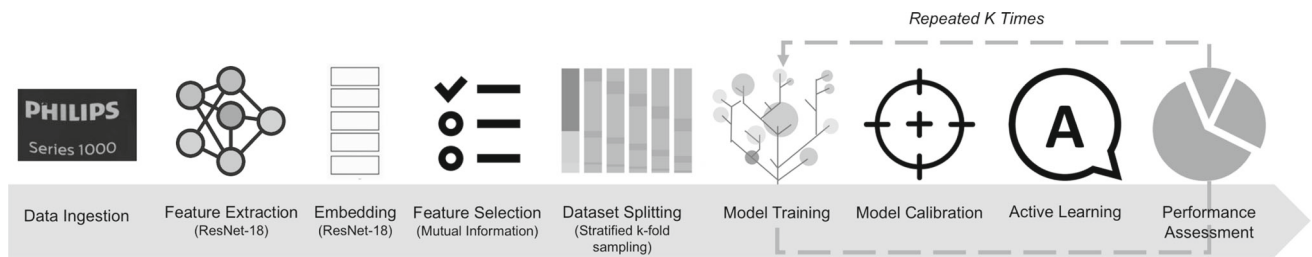


Fig. 7 The methodology we followed to train and assess machine learning models and active learning scenarios

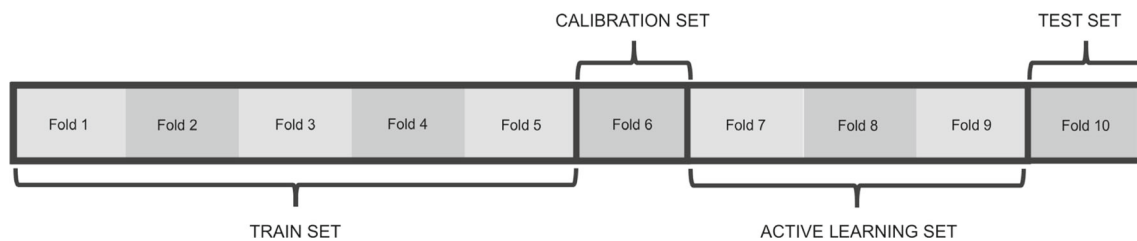


Fig. 8 A 10-fold stratified cross-validation was used. The dataset was split for four purposes: train, test, probabilities calibration, and simulate unlabeled data under an active learning setting

was used for testing (*test set*), and one for machine learning models' probabilities calibration (*calibration set*). Three folds were used to simulate a pool of unlabeled data for active learning (*active learning set*), and the rest to train the model (*train set*) (see Fig. 8). Samples are selected from the *active learning set* to be annotated by the oracle and then added to the training set, on which the models are retrained. In this research, two types of oracles were considered: (a) machine oracles, which can be imperfect, and (b) human annotators (assumed to be ideal). Five machine learning algorithms were evaluated: Gaussian Naïve Bayes, CART (*Classification and Regression Trees*, similar to C4.5, but it does not compute rule sets), Linear SVM, kNN, and Multilayer perceptron (MLP).

To evaluate the discriminative power of the machine learning models and how it is enhanced over time through active learning, the AUC ROC metric was computed. Given the multiclass setting, the “one-vs-rest” heuristic was selected, splitting the multiclass dataset into multiple binary classification problems and computing their average, weighted by the number of true instances for each class. In addition, to assess the usefulness of the active learning approaches, the AUC ROC values obtained by evaluating the model against the test fold for the first (Q1) and last (Q4) quartiles of instances queried in an active learning setting were compared. The amount of manual work saved under each active learning setting and the soft-labeling approaches' precision were also evaluated.

Through different experiments (detailed in section “[Experiments](#)”), a visual inspection pipeline was simulated (see Fig. 9). First, a stream of images is directed toward the machine learning model trained to identify possible defects. Then, based on the prediction score, a decision is made

on whether the manufactured product should remain in the production line or be deferred to manual inspection. If the product is unlikely to be defective, such a decision can be considered a label (it is considered a soft label when not made by a human annotator). The label is then persisted, enlarging the existing dataset. The enlarged dataset can be used to retrain the model and replace the existing one after a successful deployment.

Methodological aspects to evaluate probability calibration metrics and strategies

wECE metric definition. n indicates the number of classes under consideration.

$$wECE = \sum_{i=1}^n ECE_i \cdot w_i \quad (10)$$

A similar procedure was followed in the previous subsection to evaluate the proposed probability calibration metrics and techniques, avoiding the active learning step. Furthermore, a different dataset split was considered (see Fig. 10). After training the machine learning model and calibrating it with the calibration set, the non-calibrated model was used to issue a prediction for each instance in the *unlabeled data set*. The predicted class is then used to adjust further (train) the calibrator. While this introduces some noise, we expect that the better the classification model, the more it would benefit the calibrator, as explained in section “[Approximate model's probabilities calibration](#)”. Eleven performance metrics were measured: AUC ROC, ECE (computed as the ECE for each class and averaged, assigning the same weight to all classes),

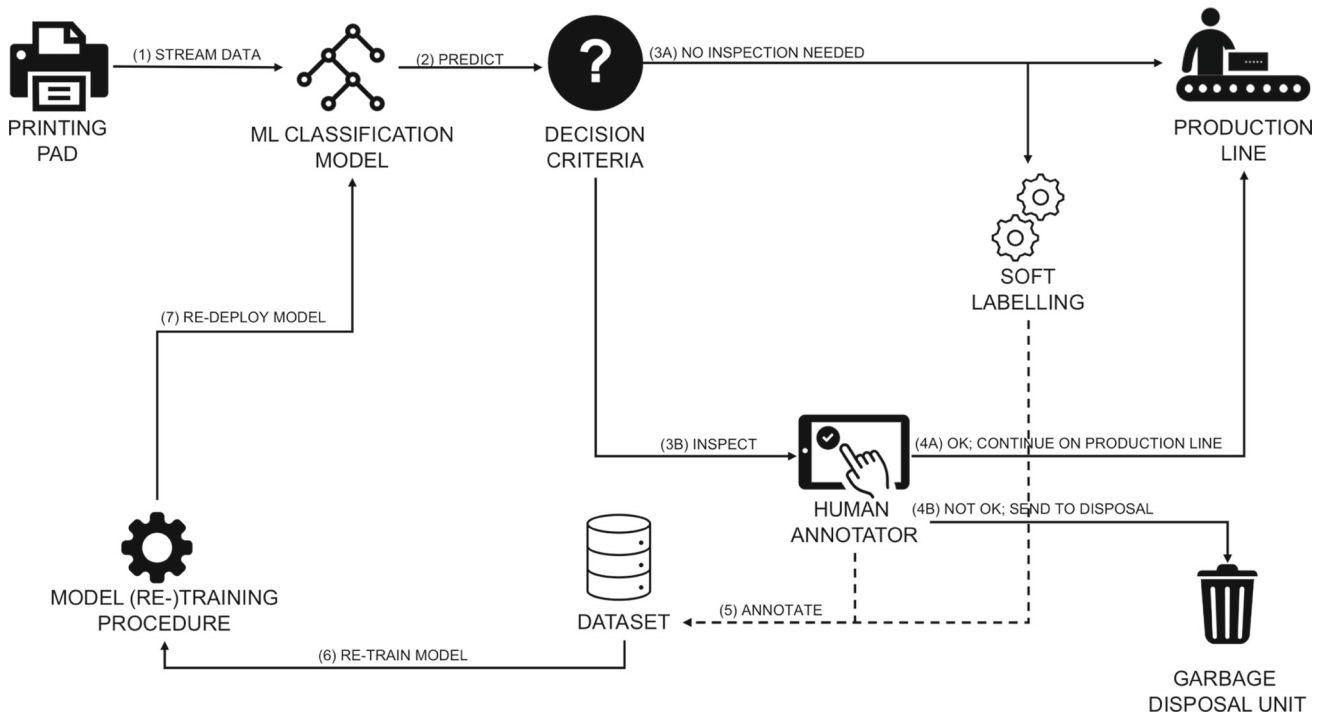


Fig. 9 Expected visual inspection pipeline in a production setting. Multiple active learning strategies were assessed to identify which would drive the best results

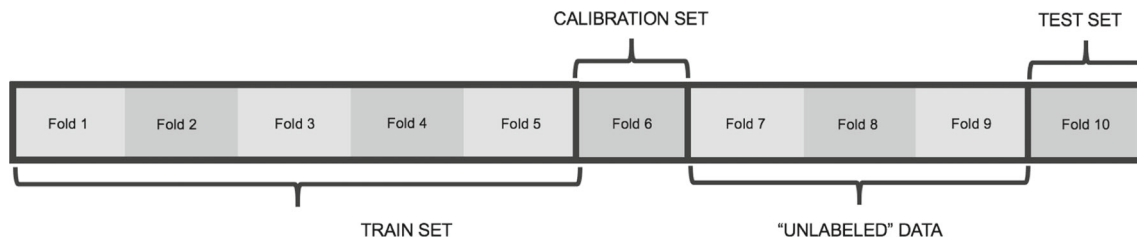


Fig. 10 A 10-fold stratified cross-validation was used. The data was split into train set, test set, calibration set, and *unlabeled data*. *Unlabeled data* was used to simulate a stream of unlabeled data and assess whether a histogram-based calibration method without ground truth can enhance its performance over time

wECE (computed as the classwise ECE—see Eq. 10), PCS, wPCS, APCS_w, wAPCS_w, APCS, wAPCS, MPCS, and wMPCS. AUC ROC measures the discriminative capability of the model and provides insights into how such capability is affected by different calibration techniques. ECE evaluates the expected difference between the accuracy and confidence of a calibration model. The ECE metric was used to compare the calibration quality for the multiple calibration techniques and the newly proposed PCS, wPCS, APCS, wAPCS, MPCS, and wMPCS metrics. Furthermore, given that the newly proposed metrics were built on a similar concept as the ECE metric, we are interested in how much they capture the same information. The Kendall τ [see Kendall (1938)] and the Pearson correlation between ECE and the newly proposed metrics were measured. The Kendall correlation measures the ordinal association between two measured quantities. In this case, it measures to what extent both metrics increase or

decrease, given the predictions for a given machine learning model and calibrators. The Pearson correlation, on the other side, was used to assess whether the correlation between metrics was linear.

The metrics were computed on the test set against the ground truth (class annotations) and the approximate ground truth (predicted classes). The results were analyzed to understand how well the metrics capture the models' performance and calibration when no ground truth is available. Furthermore, the weighted and non-weighted metrics were compared to understand how class weighting influences the final score and perception regarding the quality of the calibration.

Experiments

Experimenting with active learning strategies

For this research, two active learning settings were explored (pool-based and stream-based), using four distinct strategies to label the queried data instances in an active learning setting. Two strategies were used to select data from the active learning set under the pool-based active learning setting: (a) random sampling and (b) instances for which the classification model was the most uncertain. The model's uncertainty was assessed by considering the highest score for a given class for a given instance and selecting the instance with the lowest score among the scores provided for the data instances in the active learning set. In both cases, data were sampled until the set's exhaustion. Under the streaming active learning setting, a slightly different policy was used. When random sampling was used, a decision was made whether to keep or discard the instance with a probability threshold of 0.5. Under the highest uncertainty selection criteria, the prediction for each data instance was analyzed and derived to the oracles for labeling if it was below a certain confidence threshold ($p = 0.95$ or $p = 0.99$).

Three oracle settings were considered (see Fig. 11): (A) human labeler as the only source of truth, (B) machine oracle (classifier model) for data instances where the classifier had a high certainty, and a human labeler otherwise; and (C) machine oracle (classifier model) for data instances where the classifier had a high certainty, and requesting an additional *opinion* to another machine oracle when uncertain about the outcome. This second oracle queries the closest labeled image from three randomly picked images (one per class). In (C), the machine oracle issues a label only when both machine oracles are unanimous on the label; otherwise, the instance labeling is delegated to a human labeler. The decision regarding which oracle to query was based on the models' confidence regarding the outcome and a probability threshold set based on manufacturing quality policies. It was assumed that the second machine oracle in (C) is accessible at a certain cost (e.g., paid external service) and, therefore, cannot be used for every prediction. Such a service was simulated by computing the Structural Similarity Index Measure (SSIM) score over the queried image.

Eight scenarios were set up (see Table 1), and experimented with two quality thresholds (0.95 and 0.99 probability that the item corresponded to a certain class) and five machine learning models. The machine learning models were calibrated using a sigmoid model based on Platt logistic model (Platt, 1999) (see Eq. 11).

Platt classifier calibration logistic model. y_i denotes the truth label, and f_i denotes the uncalibrated classifier's prediction for a particular sample. A and B denote adjusted

parameters when fitting the regressor.

$$P(y_i = 1 | f_i) = \frac{1}{1 + \exp(Af_i + B)} \quad (11)$$

Experiments assessing probability calibration metrics and techniques

In an automated visual inspection setting, a labeling effort is required to (a) label data to train and calibrate the machine learning models and (b) perform a manual inspection when the models cannot determine the class of a given data instance accurately. To understand how the probability calibration affects the machine learning models, the models' predictions were compared against those obtained by (a) not calibrating the model (*No calibration*) and calibrating the model with (b) a sigmoid model based on the Platt logistic model (*Platt*), (c) temperature scaling (*Temperature*), and (d) *Histogram* calibration. Two aspects were considered in the experiments: (i) how calibration techniques compare against each other and (ii) whether calibrating a model without a ground truth can provide comparable results to models calibrated with ground truth.

Results and evaluation

Results and evaluation of active learning strategies

The active learning strategies were analyzed from two points of view. First, whether they contributed to better learning of the machine learning model. Second, how much manual work could be saved by adopting such strategies.

For the first case, the AUC ROC was measured over time (see Table 2). In particular, the models' average performance was contrasted when they consumed data within the Q1 and Q4 of the active learning pool. The best outcomes were observed for Experiment 2 (highest uncertainty with human labeler) settings, while the second-best performance was observed for Experiment 8 (highest uncertainty, with the machine and human oracles). Overall, it was observed that the streaming setting had a better average performance when compared to the pool-based experiments, despite achieving only the second-best results with Platt scaling. Furthermore, in two cases, the machine learning model degraded its performance between Q1 and Q4. This happened for Experiment 3 ($p = 0.95$) and Experiment 4 ($p = 0.95$).

Given that (a) in both experiments, a machine oracle was used, (b) no performance decrease was observed for $p = 0.99$, and (c) that the same setting did not affect the streaming case, we were tempted to conclude that most likely the machine oracles mislabeled certain instances, confusing the model when retrained and therefore reducing the

Fig. 11 Three oracle settings are explored in this research: **A** human annotator, **B** soft-labeling with classification model's outcomes for instances with high-confidence scores, and human annotator for instances where the model has low confidence; and **C** which is analogous to **B**, but the machine oracle takes into account the classifier's output score and whether the predicted class matches the class with of a labeled image with the shortest distance towards the active sample. In **C**, the sample is sent to manual revision if there is a class mismatch in the machine oracle. Samples are only discarded in a streaming setting

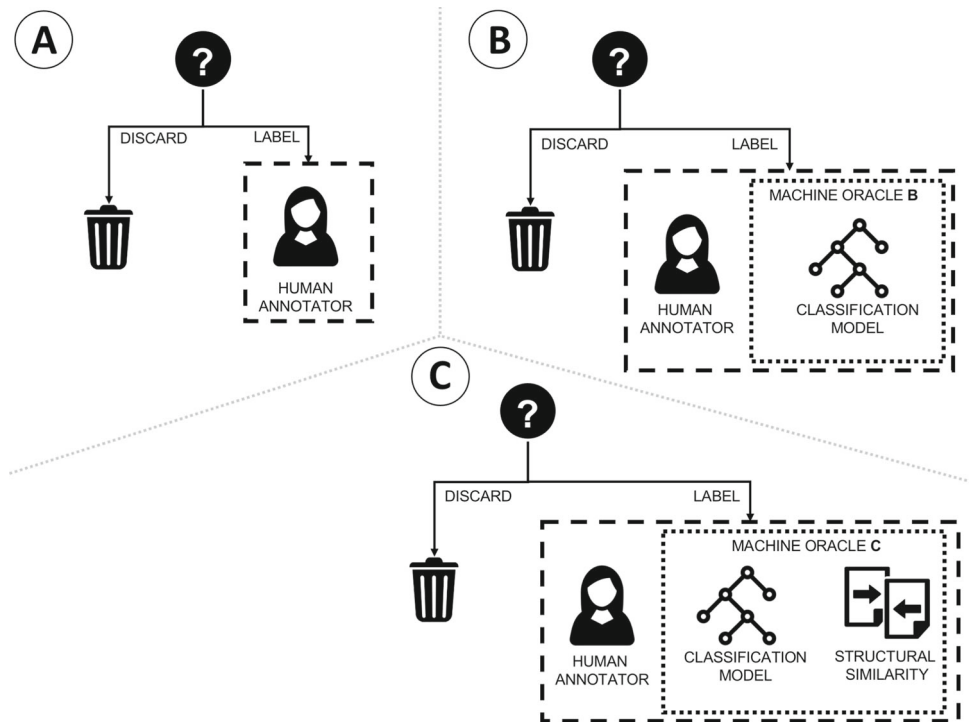


Table 1 Proposed experiments to evaluate the best active learning setting regarding how it influences the models' learning and its impact on the manual revision workload

Experiment	AL setting	AL data selection	Oracle
1	Pool-based	Random sampling	Human labeler
2	Pool-based	Highest uncertainty	Human labeler
3	Pool-based	Highest uncertainty	Machine Oracle B + Human labeler
4	Pool-based	Highest uncertainty	Machine Oracle C + Human labeler
5	Stream-based	Random sampling	Human labeler
6	Stream-based	Highest uncertainty	Human labeler
7	Stream-based	Highest uncertainty	Machine Oracle B + Human labeler
8	Stream-based	Highest uncertainty	Machine Oracle C + Human labeler

Table 2 Mean values for the mean AUC ROC computed across ten folds for five machine learning models

Setup	Experiment	p=0.95		p=0.99	
		Q1	Q4	Q1	Q4
Pool-based	1-Random, human oracle	0.8428	0.8612	0.8431	0.8623
	2-Uncertainty, human oracle	0.8594	0.8693	0.8594	0.8693
	3-Uncertainty, oracle (machine B + human)	0.8398	0.8396	0.8398	0.8398
	4-Uncertainty, oracle (machine C + human)	0.8349	0.8348	0.8358	0.8358
Streaming	5-Random, human oracle	0.8460	0.8559	0.8460	0.8559
	6-Uncertainty, human oracle	0.8525	0.8647	0.8529	0.8647
	7-Uncertainty, oracle (machine B + human)	0.8505	0.8608	0.8529	0.8647
	8-Uncertainty, oracle (machine C + human)	<i>0.8550</i>	<i>0.8665</i>	<i>0.8553</i>	<i>0.8668</i>

The results show how different active learning policies influence the models' learning over time [Q1 (first quartile) vs. Q4 (last quartile)]. Two probability thresholds (0.95 and 0.99) were considered as a soft labeling cut-off. The best results are bolded, and the second-best ones are displayed in italics

model's performance over time. Nevertheless, further analysis revealed a small fraction of soft-labeled data and that most cases were accurately labeled. While soft labeling was detrimental for the pool-based active learning settings, it led to superior results in a streaming setting, achieving results close to the best ones obtained across all experiments.

In Table 3 we report the performance of machine learning models for Experiment 2 and compare how they performed after Q1 and Q4 of the active learning pool data was shown to them. We found that the best performance was attained by the MLP, followed by the SVM by at least 0.05 AUC ROC points. Furthermore, while the MLP increased its performance over time, the SVM slightly reduced it in Q4. No other model had a performance decrease over time. Since Experiment 2 only considered a human oracle and the annotations are accurate, the performance decrease cannot be attributed to mislabeling. Furthermore, while the model's discriminative capacity loss could be attributed to the class imbalance, we consider this improbable, given that the rest of the models could better discern among the classes over time. Finally, the CART model obtained the worst results, which lagged slightly more than 0.16 AUC ROC points compared to the best one.

As mentioned at the beginning of this section, another relevant aspect of evaluating active learning strategies is their potential to reduce data annotation efforts. This could be analyzed from two perspectives. First, whether the additional data annotations provide enough knowledge to enhance the models' performance significantly. If not, the data annotation can be avoided. Second, a strategy can be devised (e.g., a machine oracle) to reduce the manual annotation effort. In this work, we focused on the second one. Table 4 presents the results for a cut-off value of $p = 0.95$. For $p = 0.99$, no instances were retrieved and given to machine oracles; therefore, no analysis was performed on them. The task required annotating 2460 samples on average.

When considering the cut-off value of 0.95, it was noticed that the Platt calibration considered a negligible number of cases for each experiment. While the quality of the annotations was high, using machine oracles would not strongly alleviate the manual labeling effort. The highest amount of soft-labeled instances corresponded to experiments with streaming settings (Experiment 7 and Experiment 8), which soft-labeled 4% and 3% of all data instances, respectively. Furthermore, 96% of samples were correctly labeled in both cases, meeting the quality threshold of $p = 0.95$. The decrease in the amount of soft labeled samples for Experiment 8 was due to discrepancies between the machine learning model and the SSIM score. Furthermore, the best machine labeling quality was achieved when considering *Oracle C* (unanimous vote of two machine oracles). When contrasting with the AUC ROC results obtained for those experiments, it was observed that while Experiments 3 and 4 slightly decreased

discriminative power, Experiments 7 and 8 increased their performance for at least 0.01 AUC ROC points.

Results and evaluation of probability calibration metrics and techniques

The experiments performed in this research aimed to validate whether the metrics proposed to measure the quality of a calibrator can be used to understand the performance of a calibrator even when no ground truth is available. Furthermore, it aimed to validate whether predictions on unlabeled data could enhance the calibrators' performance. The results are presented in Tables 5, 6, and 7. The PCS, APCS, and MPCS (along with the weighted variants) metrics were computed considering the PPCM histogram, which denotes a perfect calibration.

To understand whether the proposed metrics can measure the calibration quality without ground truth, the Pearson and Kendall correlations were computed between the ECE, wECE, APCS_W, wAPCS_W, PCS, and wPCS metrics (see Table 5). While ECE and wECE are always computed considering the ground truth at the test set, PCS, APCS_W, wPCS, and wAPCS_W were calculated considering two cases: ground truth (golden standard) and predicted labels (approximate ground truth) at the test set. Furthermore, the correlations between the metrics were evaluated in two separate moments: after calibrating the models with the calibration set (CS) and after calibrating the models with additional samples retrieved from the *unlabeled data* set (CS+UD). The results show that the correlation between ECE, PCS, APCS_W, wPCS, and wAPCS_W metrics is consistent across all cases. Furthermore, little variation exists between the values obtained when PCS or wPCS were computed on the ground truth or the approximate ground truth. While the Pearson correlation decreases after training the calibrator with predicted labels from the unlabeled data set, the Kendall correlation grew stronger when PCS or APCS_W were just averaged across classes and not weighted by the frequency of occurrence of each class. We consider the correlations moderate (negative Pearson correlation was measured between 0.50 and 0.61) or strong (negative Kendall correlation was above 0.33 and slightly below 0.40). Given the abovementioned results, we consider the PCS, wPCS, APCS_W, and wAPCS_W metrics adequately capture information conveyed by the ECE metric regardless of the source of truth used to measure the quality of the calibration. Therefore, we conclude that PCS, wPCS, APCS_W, and wAPCS_W can be used to assess the calibrators' quality when no ground truth is available.

Tables 6 and 7, compare the calibrators across multiple metrics to assess how an approximate calibration affects their discriminative power (AUC ROC) and whether it helps to enhance the calibrators' quality (ECE, PCS, APCS, MPCS,

Table 3 Mean AUC ROC values computed across ten test folds for five machine learning models

Model	Q1	Q4	DS(p=0,95)
MLP	0.9309±0.0004	0.9448±0.0003	Yes
SVM	<i>0.8788±0.0007</i>	<i>0.8767±0.0007</i>	Yes
NB	0.8628±0.0005	0.8675±0.0005	Yes
KNN	0.8575±0.0006	0.8720±0.0006	Yes
CART	0.7669±0.0007	0.7854±0.0008	Yes

The results show how the machine learning models learn over time (Q1 vs. Q4) under the Experiment 2 setting. Furthermore, we analyze if the differences were statistically significant at a p-value = 0.95 [DS(p = 0.95)]. The best results are bolded, and the second-best results are displayed in italics

Table 4 Proportion and quality of soft labeling through different settings, considering a predicted probability cut-off value of p = 0.95

Experiment	p=0.95 SL (%)	SL OK (%)	ML SL OK (%)	SSIM SL OK (%)
3	0.0077	0.9684	0.0075	NA
4	0.0033	0.9756	0.0050	0.0033
7	0.0413	0.9685	0.0400	NA
8	0.0343	0.9692	0.0483	0.0334

SL (%) denotes the percentage of soft annotated data instances w.r.t. the total, *SL OK (%)* denotes the percentage of correctly soft annotated instances, *ML SL OK (%)* denotes the percentage of soft annotated data instances w.r.t. the total that would be correctly annotated considering the ML model score, *SSIM SL OK (%)* denotes the percentage of soft annotated data instances w.r.t. the total that would be correctly annotated considering the SSIM score

Table 5 The results were obtained for different models and probability calibration techniques

Source of truth	Correlation	Pearson		Kendall	
	Calibration data	CS	CS+UD	CS	CS+UD
Golden standard	ECE vs. PCS	− 0.6100	− 0.5937	− 0.3360	− 0.3981
	wECE vs. wPCS	− 0.5113	− 0.5070	− 0.3574	− 0.3525
	ECE vs. APCS _W	− 0.6100	− 0.5939	− 0.3316	− 0.3981
	wECE vs. wAPCS _W	− 0.5112	− 0.5070	0.3574	− 0.3480
Predicted labels	ECE vs. PCS	− 0.6100	− 0.5937	− 0.3360	− 0.3981
	wECE vs. wPCS	− 0.5084	− 0.5017	− 0.3360	− 0.3308
	ECE vs. APCS _W	− 0.6100	− 0.5939	− 0.3316	− 0.3981
	wECE vs. wAPCS _W	− 0.5083	− 0.5016	0.3423	− 0.3308

We show Pearson and Kendall correlation coefficients when comparing the ECE, APCS_W, wAPCS_W, PCS, and wPCS metrics. APCS_W, wAPCS_W, PCS, and wPCS are measured considering the ground truth (golden standard) or approximate ground truth (predicted labels). The ECE metric is always computed considering the ground truth. *CS* stands for *Calibration Set*, while *CS+UD* abbreviates *Calibration Set + Unlabeled Data*

and their weighted variants). Furthermore, Fig. 12 presents calibration plots for each calibrator for CS and CS+UD for visual assessment.

When comparing the calibrators through non-weighted metrics (see Table 6), we consider the Platt calibrator achieved the most stable performance. While the measured quality of calibration slightly decreased with the approximate calibration, it must be noticed that a higher proportion of positives was allocated at higher scores. Furthermore, while with the approximate calibration, the model's overall discriminative power slightly decreased, it remained superior against other models (even the not calibrated) by at least 0.02 AUC

ROC points. The Histogram and Temperature calibrators provide an interesting case, given both had a similar initial (CS) calibration quality if measured with the ECE, PCS, APCS, or MPCS metrics. Nevertheless, the metrics at CS+UD showed discrepancies: while ECE slightly increased for the Histogram calibrator (showing a worse calibration quality), it remained the same for the Temperature calibrator. On the other hand, PCS, APCS, and MPCS decreased (signaling a worse calibration quality) for both the Histogram and Temperature calibrator. Furthermore, the decrease in the metrics' values was more pronounced for the Histogram calibrator. When visually assessing both calibrators, we found that they

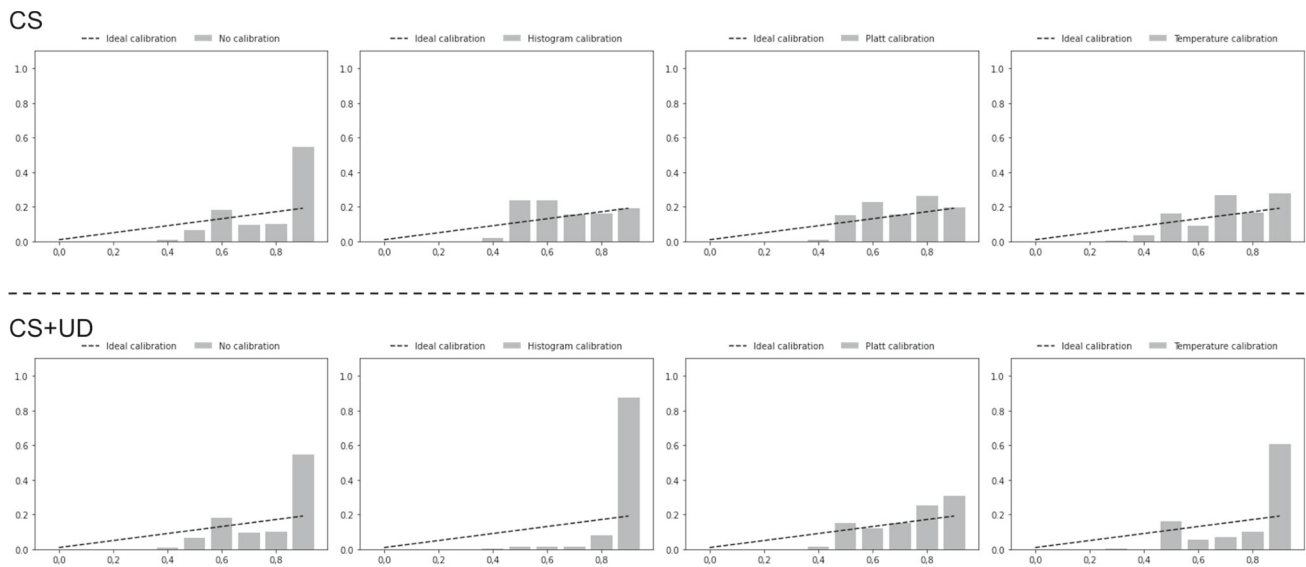


Fig. 12 Eight calibration plots, comparing *No calibration*, *Histogram calibration*, *Platt calibration*, and *Temperature calibration* at CS (calibrated with the calibration set (ground truth)) and CS + UD (calibrated with a calibration set and predicted labels over time). The calibration plots have been adapted, showing normalized values (their sum is one)

rather than the usual fraction of positives on the dependent variable axis. The x-axis denotes the mean predicted probability for a given class. The histograms average predictions across classes and calibrated machine learning models

had a similar initial distribution (CS), but the Histogram calibrator ended up much more skewed than the Temperature calibrator at CS+UD. While the ECE metric did not capture this behavior, it was successfully summarized in the PCS, APCS, and MPCS metrics. We found the same patterns could be observed when analyzing the weighted metrics (see Table 7).

From the results above, we confirm that the proposed metrics can accurately measure the quality of calibration of a given calibrator when no ground truth is available. Furthermore, the metrics have shown to provide a more accurate measurement of the calibrators' quality than ECE, overcoming some of its shortcomings (e.g., providing a more holistic

view of the distribution of positives along the mean predicted probability, taking into account empty bins).

Our research shows that tracking predictions over time did not enhance the quality of calibration for any of the methods involved (Histogram calibration, Platt calibration, or Temperature calibration). Finding accurate calibration models for probability calibration, given a lack of ground truth, remains a matter of future work.

Table 6 The results were obtained for different probability calibration techniques

Source of truth	Calibration	AUC ROC ↑		ECE ↓		PCS ↑		APCS ↑		MPCS ↑	
		CS	CS+UD	CS	CS+UD	CS	CS+UD	CS	CS+UD	CS	CS+UD
Golden standard	None	0.8630	0.8630	0.1090	0.1090	0.7636	0.7636	0.7448	0.7448	0.5647	0.5647
	Histogram	0.8432	0.8442	0.1051	0.1084	0.8050	0.6509	0.7458	0.6697	0.5539	0.4507
	Platt	0.8907	0.8903	0.0914	0.0955	0.7523	0.7421	0.7669	0.7613	0.5904	0.5820
	Temperature	0.8614	0.8609	0.1090	0.1090	0.8073	0.7548	0.7651	0.7383	0.5886	0.5517
Predicted labels	None	0.8630	0.8630	0.1090	0.1090	0.7636	0.7636	0.7448	0.7448	0.5647	0.5647
	Histogram	0.8432	0.8442	0.1051	0.1084	0.8050	0.6509	0.7458	0.6697	0.5539	0.4507
	Platt	0.8907	0.8903	0.0914	0.0955	0.7523	0.7421	0.7669	0.7613	0.5904	0.5820
	Temperature	0.8614	0.8609	0.1090	0.1090	0.8073	0.7548	0.7651	0.7383	0.5886	0.5517

PCS, APCS, and MPCS are measured considering the ground truth (golden standard) or approximate ground truth (predicted labels). The AUC ROC and ECE metrics are always computed considering the ground truth. The best results are bolded

Table 7 The results were obtained for different probability calibration techniques

Source of truth	Calibration	AUC ROC ↑		wECE ↓		wPCS ↑		wAPCS ↑		wMPCS ↑	
		CS	CS+UD	CS	CS+UD	CS	CS+UD	CS	CS+UD	CS	CS+UD
Golden standard	None	0.8630	0.8630	0.1457	0.1457	0.7829	0.7829	0.7545	0.7545	0.5801	0.5801
	Histogram	0.8432	0.8442	0.1410	0.1449	0.8265	0.6494	0.7565	0.6689	0.5685	0.4505
	Platt	0.8907	0.8903	0.1230	0.1285	0.7655	0.7527	0.7735	0.7666	0.6010	0.5902
	Temperature	0.8614	0.8609	0.1457	0.1457	0.8232	0.7773	0.7730	0.7496	0.6014	0.5697
Predicted labels	None	0.8630	0.8630	0.1457	0.1457	0.7826	0.7826	0.7543	0.7543	0.5799	0.5799
	Histogram	0.8432	0.8442	0.1410	0.1449	0.8265	0.6494	0.7565	0.6689	0.5686	0.4505
	Platt	0.8907	0.8903	0.1230	0.1285	0.7641	0.7517	0.7728	0.7662	0.5998	0.5894
	Temperature	0.8614	0.8609	0.1457	0.1457	0.8230	0.7765	0.7730	0.7492	0.6013	0.5692

wPCS, wAPCS, and wMPCS are measured considering the ground truth (golden standard) or approximate ground truth (predicted labels). The AUC ROC and wECE metrics are always computed considering the ground truth. The best results are bolded

Conclusions and future work

This work explored active learning with multiple oracles to alleviate the manual inspection of manufactured products and the labeling of inspected products. Our active learning settings can save up to four percent of the manual inspection and data labeling load while not compromising on the quality of the outcome for a quality threshold of $p = 0.95$. It must be noted that labeling savings depend on the machine learning model deployed, the acceptance quality levels, and the quality of the active learning machine oracles under consideration. Furthermore, multiple probability calibration techniques were compared, and several new metrics to measure the quality of a calibrator were proposed. The metrics enable measuring the calibrators' quality even when no ground truth is available. The experiments demonstrated that the proposed metrics capture relevant data otherwise summarized in the ECE metric—a popular metric to measure the quality of a probability calibration model. Nevertheless, the behavior of the proposed metrics under concept drift was not studied yet, and we consider it a matter of future research.

We envision multiple lines of investigation for future work. Regarding active learning, we are interested in enriching our current setup by adopting different strategies to decide how interesting an upcoming image is (e.g., learning distance metrics for each class or learning to predict which piece of data would enhance the classifier the most) and enhancing the calibration techniques to display the desired behavior for high-confidence thresholds. We will conduct further research on probabilities calibration to understand how the proposed metrics behave when concept drift occurs. Finally, we will explore new approximate probability calibration approaches leading to enhanced calibrators when no ground truth is available.

Acknowledgements This work was supported by the Slovenian Research Agency and the European Union's Horizon 2020 project STAR under grant agreement H2020-956573.

Author Contributions JMR: conceptualization, methodology, software programming, validation, formal analysis, investigation, writing (original draft, review and editing), visualization, supervision. LB: conceptualization, validation, writing (review and editing). ET: software programming, writing (review and editing). PZ: software programming, writing (review and editing). JK: resources, data curation. BF: resources, writing (review and editing), supervision, project administration, funding acquisition. DM: resources, writing (review and editing), supervision, project administration, funding acquisition.

Funding Open access publishing supported by the Slovenian Research Agency and Central Technical Library in Ljubljana.

Data availability The datasets analyzed during the current study are not publicly available for confidentiality reasons.

Declarations

Conflict of interest The authors have no competing interests to declare relevant to this article's content.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Aggour, K. S., Gupta, V. K., Ruscitto, D., Ajdelsztajn, L., Bian, X., Brosnan, K. H., Kumar, N. C., Dheeradhada, V., Hanlon, T., Iyer, N., et al. (2019). Artificial intelligence/machine learning in manufacturing and inspection: A GE perspective. *MRS Bulletin*, 44(7), 545–558. <https://doi.org/10.1557/mrs.2019.157>

- Aiger, D., & Talbot, H. (2012). The phase only transform for unsupervised surface defect detection. In S. Barthel, J. Colding, & T. Elmqvist (Eds.), *Emerging topics in computer vision and its applications* (pp. 215–232). World Scientific. <https://doi.org/10.1109/CVPR.2010.5540198>
- Aminzadeh, M., & Kurfess, T. R. (2019). Online quality inspection using Bayesian classification in powder-bed additive manufacturing from high-resolution visual camera images. *Journal of Intelligent Manufacturing*, 30, 2505–2523.
- Barari, A., de Sales Guerra Tsuzuki, M., Cohen, Y., & Macchi, M. (2021). Intelligent manufacturing systems towards industry 4.0 era. *Journal of Intelligent Manufacturing*, 32, 1793–1796.
- Beltrán-González, C., Bustreo, M., & Del Bue, A. (2020). External and internal quality inspection of aerospace components. In *2020 IEEE 7th international workshop on metrology for aerospace (MetroAeroSpace)*, (pp. 351–355). IEEE. <https://doi.org/10.1109/MetroAeroSpace48742.2020.9160103>
- Beluch, W. H., Genewein, T., Nürnberger, A., & Köhler, J. M. (2018). The power of ensembles for active learning in image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, (pp. 9368–9377). <https://doi.org/10.1109/CVPR.2018.00976>
- Bradley, A. P. (1997). The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7), 1145–1159. [https://doi.org/10.1016/S0031-3203\(96\)00142-2](https://doi.org/10.1016/S0031-3203(96)00142-2)
- Bröcker, J., & Smith, L. A. (2007). Increasing the reliability of reliability diagrams. *Weather and Forecasting*, 22(3), 651–661. <https://doi.org/10.1175/WAF993.1>
- Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Grobler, J., Layton, R., VanderPlas, J., Joly, A., Holt, B., & Varoquaux, G. (2013). *API design for machine learning software: experiences from the scikit-learn project* (pp. 108–122). In ECML PKDD Workshop: Languages for Data Mining and Machine Learning.
- Carvajal Soto, J., Tavakolizadeh, F., & Gyulai, D. (2019). An online machine learning framework for early detection of product failures in an industry 4.0 context. *International Journal of Computer Integrated Manufacturing*, 32(4–5), 452–465. <https://doi.org/10.1080/0951192X.2019.1571238>
- Cheeseman, P. C. (1985). In defense of probability. In *IJCAI*, 85, 1002–1009.
- Chouchene, A., Carvalho, A., Lima, T. M., Charrua-Santos, F., Osório, G. J., & Barhoumi, W. (2020). Artificial intelligence for product quality inspection toward smart industries: quality control of vehicle non-conformities. In *2020 9th international conference on industrial technology and management (ICITM)*, (pp. 127–131). IEEE. <https://doi.org/10.1109/ICITM48982.2020.9080396>
- Cohen, I. & Goldszmidt, M. (2004). Properties and benefits of calibrated classifiers. In *European conference on principles of data mining and knowledge discovery*, (pp. 125–136). Springer. https://doi.org/10.1007/978-3-540-30116-5_14
- Cohen, N. & Hoshen, Y. (2020). Sub-image anomaly detection with deep pyramid correspondences. [arXiv:2005.02357](https://arxiv.org/abs/2005.02357)
- Cohn, D., Atlas, L., & Ladner, R. (1994). Improving generalization with active learning. *Machine Learning*, 15(2), 201–221. <https://doi.org/10.1007/BF00993277>
- Cordier, A., Das, D., & Gutierrez, P. (2021). Active learning using weakly supervised signals for quality inspection. [arXiv:2104.02973](https://arxiv.org/abs/2104.02973)
- Cullinane, S.-J., Bosak, J., Flood, P. C., & Demerouti, E. (2013). Job design under lean manufacturing and its impact on employee outcomes. *Organizational Psychology Review*, 3(1), 41–61. <https://doi.org/10.1177/2041386612456412>
- Dai, W., Mujeeb, A., Erdt, M., & Sourin, A. (2018). Towards automatic optical inspection of soldering defects. In *2018 International Conference on Cyberworlds (CW)*, (pp. 375–382). IEEE. <https://doi.org/10.1109/CW.2018.00074>
- Duan, G., Wang, H., Liu, Z., & Chen, Y.-W. (2012). A machine learning-based framework for automatic visual inspection of microdrill bits in PCB production. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(6), 1679–1689. <https://doi.org/10.1109/TSMCC.2012.2216260>
- Guo, C., Pleiss, G., Sun, Y., & Weinberger, K. Q. (2017). On calibration of modern neural networks. In *International conference on machine learning*, (pp. 1321–1330). PMLR.
- Gupta, C. & Ramdas, A. (2021). Distribution-free calibration guarantees for histogram binning without sample splitting. In *International Conference on Machine Learning*, (pp. 3942–3952). PMLR.
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, (pp. 770–778). <https://doi.org/10.48550/arXiv.1512.03385>
- Hua, J., Xiong, Z., Lowey, J., Suh, E., & Dougherty, E. R. (2005). Optimal number of features as a function of sample size for various classification rules. *Bioinformatics*, 21(8), 1509–1515. <https://doi.org/10.1093/bioinformatics/bti171>
- Jezek, S., Jonak, M., Burget, R., Dvorak, P., & Skotak, M. (2021). Deep learning-based defect detection of metal parts: evaluating current methods in complex conditions. In *2021 13th International congress on ultra modern telecommunications and control systems and workshops (ICUMT)*, (pp. 66–71). IEEE. <https://doi.org/10.1109/ICUMT54235.2021.9631567>
- Jiang, J. & Wong, W. (2018). Fundamentals of common computer vision techniques for textile quality control. In *Applications of computer vision in fashion and textiles*, (pp. 3–15). Elsevier. <https://doi.org/10.1016/B978-0-08-101217-8.00001-4>
- Jian, C., Gao, J., & Ao, Y. (2017). Automatic surface defect detection for mobile phone screen glass based on machine vision. *Applied Soft Computing*, 52, 348–358. <https://doi.org/10.1016/j.asoc.2016.10.030>
- Kang, G.-W. & Liu, H.-B. (2005). Surface defects inspection of cold rolled strips based on neural network. In *2005 international conference on machine learning and cybernetics*, (Vol. 8, pp. 5034–5037). IEEE. <https://doi.org/10.1109/ICMLC.2005.1527830>
- Kendall, M. G. (1938). A new measure of rank correlation. *Biometrika*, 30(1/2), 81–93. <https://doi.org/10.2307/2332226>
- Kuhn, M., Johnson, K., et al. (2013). *Applied predictive modeling* (Vol. 26). Springer. <https://doi.org/10.1007/978-1-4614-6849-3>
- Kujawińska, A., Vogt, K., & Hamrol, A. (2016). The role of human motivation in quality inspection of production processes. In *Advances in ergonomics of manufacturing: managing the enterprise of the future*, (pp. 569–579). Springer. <https://doi.org/10.1007/978-1-4614-6849-3>
- Kumar, A., Liang, P. S., & Ma, T. (2019). Verified uncertainty calibration. *Advances in Neural Information Processing Systems*, 32, 1–8.
- Küppers, F., Kronenberger, J., Shantia, A., & Haselhoff, A. (2020). Multivariate confidence calibration for object detection. In *The IEEE/CVF conference on computer vision and pattern recognition (CVPR) Workshops*
- Kurniati, N., Yeh, R.-H., & Lin, J.-J. (2015). Quality inspection and maintenance: The framework of interaction. *Procedia Manufacturing*, 4, 244–251. <https://doi.org/10.1016/j.promfg.2015.11.038>
- Leathart, T., Frank, E., Holmes, G., & Pfahringer, B. (2017). Probability calibration trees. In *Asian Conference on Machine Learning*, (pp. 145–160). PMLR
- Lewis, D. D. & Catlett, J. (1994). Heterogeneous uncertainty sampling for supervised learning. In *Machine learning proceedings 1994*, (pp. 148–156). Elsevier. <https://doi.org/10.1016/B978-1-55860-335-6.50026-X>

- Li, C.-L., Sohn, K., Yoon, J., and Pfister, T. (2021). Cutpaste: Self-supervised learning for anomaly detection and localization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, (pp. 9664–9674).
- Lin, H., Li, B., Wang, X., Shu, Y., & Niu, S. (2019). Automated defect inspection of led chip using deep convolutional neural network. *Journal of Intelligent Manufacturing*, 30, 2525–2534.
- Meng, L., McWilliams, B., Jarosinski, W., Park, H.-Y., Jung, Y.-G., Lee, J., & Zhang, J. (2020). Machine learning in additive manufacturing: A review. *JOM Journal of the Minerals Metals and Materials Society*, 72(6), 2363–2377. <https://doi.org/10.1007/s11837-020-04155-y>
- Mujeeb, A., Dai, W., Erdt, M., & Sourin, A. (2018). Unsupervised surface defect detection using deep autoencoders and data augmentation. In *2018 International conference on cyberworlds (CW)*, (pp. 391–398). IEEE. <https://doi.org/10.1109/CW.2018.00076>
- Newman, T. S., & Jain, A. K. (1995). A survey of automated visual inspection. *Computer Vision and Image Understanding*, 61(2), 231–262. <https://doi.org/10.1006/cviu.1995.1017>
- Nixon, J., Dusenberry, M. W., Zhang, L., Jerfel, G., & Tran, D. (2019). Measuring calibration in deep learning. In *CVPR Workshops*, volume 2.
- Ovadia, Y., Fertig, E., Ren, J., Nado, Z., Sculley, D., Nowozin, S., Dillon, J., Lakshminarayanan, B., & Snoek, J. (2019). Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift. *Advances in neural information processing systems*, 32, 1–12.
- Park, J.-K., Kwon, B.-K., Park, J.-H., & Kang, D.-J. (2016). Machine learning-based imaging system for surface defect inspection. *International Journal of Precision Engineering and Manufacturing-Green Technology*, 3(3), 303–310. <https://doi.org/10.1007/s40684-016-0039-x>
- Peyré, G., Cuturi, M., et al. (2019). Computational optimal transport: With applications to data science. *Foundations and Trends in Machine Learning*, 11(5–6), 355–607.
- Platt, J., et al. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in Large Margin Classifiers*, 10(3), 61–74.
- Platt, J. C. (2000). *5 Probabilities for SV machines*. *Advances in large margin classifiers* (p. 61). MIT Press.
- Posocco, N. and Bonnefoy, A. (2021). Estimating expected calibration errors. In *International conference on artificial neural networks*, (pp. 139–150). Springer. <https://doi.org/10.1007/978-3-030-86380-712>
- Rai, R., Tiwari, M. K., Ivanov, D., & Dolgui, A. (2021). Machine learning in manufacturing and industry 4.0 applications. <https://doi.org/10.1080/00207543.2021.1956675>
- Ren, R., Hung, T., & Tan, K. C. (2017). A generic deep-learning-based approach for automated surface inspection. *IEEE Transactions on Cybernetics*, 48(3), 929–940. <https://doi.org/10.1109/TCYB.2017.2668395>
- Ren, P., Xiao, Y., Chang, X., Huang, P.-Y., Li, Z., Chen, X., & Wang, X. (2020). A survey of deep active learning. [arXiv:2009.00236](https://arxiv.org/abs/2009.00236)
- Rippel, O., Haumering, P., Brauers, J., & Merhof, D. (2021). Anomaly detection for the automated visual inspection of pet preform closures. In *2021 26th IEEE international conference on emerging technologies and factory automation (ETFA)*, (pp. 1–7). IEEE. <https://doi.org/10.1109/ETFA45728.2021.9613298>
- Rozanec, J. M., Novalija, I., Zajec, P., Kenda, K., Tavakoli Ghinani, H., Suh, S., Veliou, E., Papamartzivanos, D., Giannetos, T., Menesidou, S.-A., et al. (2022). Human-centric artificial intelligence architecture for industry 5.0 applications. *International Journal Production Research*. <https://doi.org/10.1080/00207543.2022.2138611>
- Rožanec, J. M., Trajkova, E., Dam, P., Fortuna, B., & Mladenici, D. (2022). Streaming machine learning and online active learning for automated visual inspection. *IFAC-PapersOnLine*, 55(2), 277–282. <https://doi.org/10.1016/j.ifacol.2022.04.206>
- Schmitt, J., Bönig, J., Borggräfe, T., Beiting, G., & Deuse, J. (2020). Predictive model-based quality inspection using machine learning and edge cloud computing. *Advanced Engineering Informatics*, 45, 101101. <https://doi.org/10.1016/j.aei.2020.101101>
- See, J. E. (2012). Visual inspection: A review of the literature. Sandia Report SAND2012-8590, Sandia National Laboratories, Albuquerque, New Mexico. <https://doi.org/10.2172/1055636>
- Selvi, S. S. T., & Nasira, G. (2017). An effective automatic fabric defect detection system using digital image processing. *Journal of Water and Environmental Nanotechnology*, 6(1), 79–85. <https://doi.org/10.13074/jent.2017.03.171241>
- Settles, B. (2009). Active learning literature survey.
- Silva Filho, T., Song, H., Perello-Nieto, M., Santos-Rodriguez, R., Kull, M., & Flach, P. (2021). Classifier calibration: How to assess and improve predicted class probabilities: A survey. (pp. arXiv–2112). <https://doi.org/10.48550/arXiv.2112.10327>
- Song, H., Perello-Nieto, M., Santos-Rodriguez, R., Kull, M., Flach, P., et al. (2021). Classifier calibration: How to assess and improve predicted class probabilities: A survey. [arXiv:2112.10327](https://arxiv.org/abs/2112.10327)
- Tsai, D.-M., & Lai, S.-C. (2008). Defect detection in periodically patterned surfaces using independent component analysis. *Pattern Recognition*, 41(9), 2812–2832. <https://doi.org/10.1016/j.patcog.2008.02.011>
- Valavanis, I., & Kosmopoulos, D. (2010). Multiclass defect detection and classification in weld radiographic images using geometric and texture features. *Expert Systems with Applications*, 37(12), 7606–7614. <https://doi.org/10.1016/j.eswa.2010.04.082>
- van Garderen, K. (2018). Active learning for overlay prediction in semiconductor manufacturing.
- Vergara, J. R., & Estévez, P. A. (2014). A review of feature selection methods based on mutual information. *Neural Computing and Applications*, 24(1), 175–186. <https://doi.org/10.1007/s00521-013-1368-0>
- Vergara-Villegas, O. O., Cruz-Sánchez, V. G., Jesús Ochoa-Domínguez, H. d., Jesús Nandayapa-Alfaro, M. d., & Flores-Abad, Á. (2014). Automatic product quality inspection using computer vision systems. In *Lean manufacturing in the developing World*, (pp. 135–156). Springer.
- Villalba-Diez, J., Schmidt, D., Gevers, R., Ordieres-Meré, J., Buchwitz, M., & Wellbrock, W. (2019). Deep learning for industrial computer vision quality control in the printing industry 4.0. *Sensors*, 19(18), 3987. <https://doi.org/10.3390/s19183987>
- Villani, C. (2009). *Optimal transport: Old and new* (Vol. 338). Springer.
- Weiss, S. M., Dhurandhar, A., Baseman, R. J., White, B. F., Logan, R., Winslow, J. K., & Poindexter, D. (2016). Continuous prediction of manufacturing performance throughout the production lifecycle. *Journal of Intelligent Manufacturing*, 27(4), 751–763. <https://doi.org/10.1007/s10845-014-0911-x>
- Wuest, T., Irgens, C., & Thoben, K.-D. (2014). An approach to monitoring quality in manufacturing using supervised machine learning on product state data. *Journal of Intelligent Manufacturing*, 25(5), 1167–1180. <https://doi.org/10.1007/s10845-013-0761-y>
- Yang, J., Li, S., Wang, Z., Dong, H., Wang, J., & Tang, S. (2020). Using deep learning to detect defects in manufacturing: A comprehensive survey and current challenges. *Materials*, 13(24), 5755. <https://doi.org/10.3390/ma13245755>
- Yun, J. P., Choi, D.-C., Jeon, Y.-J., Park, C., & Kim, S. W. (2014). Defect inspection system for steel wire rods produced by hot rolling process. *The International Journal of Advanced Manufacturing Technology*, 70(9), 1625–1634. <https://doi.org/10.1007/s00170-013-5397-8>
- Zajec, P., Rožanec, J. M., Novalija, I., Fortuna, B., Mladenici, D., & Kenda, K. (2021). Towards active learning based smart assistant for manufacturing. In *IFIP international conference on advances in*

- production management systems*, (pp. 295–302). Springer. <https://doi.org/10.1007/978-3-030-85910-731>
- Zavrtanik, V., Kristan, M., & Skočaj, D. (2021). Draem-a discriminatively trained reconstruction embedding for surface anomaly detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, (pp. 8330–8339)
- Zavrtanik, V., Kristan, M., & Skočaj, D. (2022). Dsr—A dual subspace re-projection network for surface anomaly detection supplementary material.
- Zavrtanik, V., Kristan, M., & Skočaj, D. (2022). Dsr—A dual subspace re-projection network for surface anomaly detection. In *European conference on computer vision*, (pp. 539–554). Springer. <https://doi.org/10.1007/978-3-031-19821-231>
- Zeng, X., & Martinez, T. R. (2000). Distribution-balanced stratified cross-validation for accuracy estimation. *Journal of Experimental & Theoretical Artificial Intelligence*, 12(1), 1–12. <https://doi.org/10.1080/095281300146272>
- Zheng, T., Ardolino, M., Bacchetti, A., & Perona, M. (2021). The applications of industry 4.0 technologies in manufacturing context: A systematic literature review. *International Journal of Production Research*, 59(6), 1922–1954. <https://doi.org/10.1080/00207543.2020.1824085>
- Zheng, Z., Zhang, S., Yu, B., Li, Q., & Zhang, Y. (2020). Defect inspection in tire radiographic image using concise semantic segmentation. *IEEE Access*, 8, 112674–112687. <https://doi.org/10.1109/ACCESS.2020.3003089>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

3.1 Results amendment: Results and Evaluation of Probability Calibration Metrics and Techniques

As described in the previous section, the experiments performed in this research had two purposes. First, they aimed to validate whether the metrics proposed to measure the quality of a calibrator can be used to understand the performance of a calibrator even when no ground truth is available. Second, they aimed to validate whether predictions on unlabeled data could enhance the calibrators' performance. We present the updated results in Table 3.1, Table 3.2, and Table 3.3. The PCS, APCS, and MPCS (along with the weighted variants) metrics were computed considering the PPCM histogram, which denotes a perfect calibration.

The results in Table 3.1 show that the correlation between ECE, PCS, APCS_W, wECE, wPCS, and wAPCS_W metrics is consistent across all cases and little variation exists between the values obtained when PCS or wPCS were computed on the ground truth or the approximate ground truth. The Pearson and Kendall correlations increased in all but one case after training the calibrator with predicted labels from the unlabeled data set. We consider the correlations moderate (Pearson correlation was measured between 0.40 and 0.48, and Kendall correlation was above 0.20 and below 0.31). The correlations were tested statistically significant, considering a p-value of 0.05. Given the abovementioned results, we consider the PCS, wPCS, APCS_W, and wAPCS_W metrics adequately capture information conveyed by the ECE and wECE metrics regardless of the source of truth used to measure the quality of the calibration. Therefore, we conclude that PCS, wPCS, APCS_W, and wAPCS_W can be used to assess the calibrators' quality when no ground truth is available.

Table 3.2 and Table 3.3, compare the calibrators across multiple metrics to assess how an approximate calibration affects their discriminative power (AUC ROC) and whether it helps to enhance the calibrators' quality (ECE, PCS, APCS, MPCS, and their weighted variants). When comparing the calibrators through non-weighted metrics (see Table 3.2), we consider the Platt calibrator achieved the most stable performance. While the measured calibration quality slightly decreased with the approximate calibration, it must be noticed that a higher proportion of positives was allocated at higher scores. Furthermore, while with the approximate calibration, the model's overall discriminative power slightly decreased, it remained superior against other models (even the not calibrated) by at least 0.02 AUC ROC points. The Histogram and Temperature calibrators provide an interesting case, given both had a similar initial (CS) calibration quality if measured with the ECE, PCS, APCS, or MPCS metrics. Nevertheless, the metrics at CS+UD showed discrepancies: while ECE slightly increased for the Histogram calibrator (showing a worse calibration quality), it remained the same for the Temperature calibrator. On the other hand, PCS, APCS, and MPCS decreased (signaling a worse calibration quality) for both the Histogram and Temperature calibrator. Furthermore, the decrease in the metrics' values was more pronounced for the Histogram calibrator. When visually assessing both calibrators, we found that they had a similar initial distribution (CS), but the Histogram calibrator ended up much more skewed than the Temperature calibrator at CS+UD. While the ECE metric did not capture this behavior, it was successfully summarized in the PCS, APCS, and MPCS metrics. We found the same patterns could be observed when analyzing the weighted metrics (see Table 3.3).

From the results above, we confirm that the proposed metrics can accurately measure the quality of calibration of a given calibrator when no ground truth is available. Furthermore, the metrics have shown to provide a more accurate measurement of the calibrators' quality than ECE, overcoming some of its shortcomings (e.g., providing a more holistic view

of the distribution of positives along the mean predicted probability, taking into account empty bins).

Our research shows that tracking predictions over time did not enhance the quality of calibration for any of the methods involved (Histogram calibration, Platt calibration, or Temperature calibration). Finding accurate calibration models for probability calibration, given a lack of ground truth, remains a matter of future work.

Source of truth	Calibration data	Correlation when compared against (w)ECE							
		Pearson				Kendall			
		PCS	APCSW	wPCS	wAPCSW	PCS	APCSW	wPCS	wAPCSW
Golden standard	CS	-0.4646	-0.4646	-0.4011	-0.4011	-0.2645	-0.2645	-0.2089	-0.2089
	CS+UD	-0.4848	-0.4848	-0.4213	-0.4213	-0.3152	-0.3152	-0.2653	-0.2653
Predicted labels	CS	-0.4260	-0.4260	-0.4509	-0.4509	-0.2517	-0.2517	-0.2667	-0.2667
	CS+UD	-0.4436	-0.4436	-0.4159	-0.4159	-0.2681	-0.2681	-0.2383	-0.2383

Table 3.1: The results were obtained for different models and probability calibration techniques. We show Pearson and Kendall correlation coefficients when comparing the ECE, PCS, APCS_W, and the wECE, wPCS, and wAPCS_W metrics. APCS_W, wAPCS_W, PCS, and wPCS are measured considering the ground truth (golden standard) or approximate ground truth (predicted labels). The ECE and wECE metrics are always computed considering the ground truth. *CS* stands for *Calibration Set*, while *CS+UD* abbreviates *Calibration Set + Unlabeled Data*. This table relates to Table 5 in the paper [35].

Source of truth	Calibration	AUC ROC ↑		ECE ↓		PCS ↑		APCS ↑		MPCS ↑	
		CS	CS+UD	CS	CS+UD	CS	CS+UD	CS	CS+UD	CS	CS+UD
Golden standard	None	0.8630	0.8630	0.1090	0.1090	0.7636	0.7636	0.7448	0.7448	0.5647	0.5647
	Histogram	0.8432	0.8442	0.1051	0.1084	0.8050	0.6509	0.7458	0.6697	0.5539	0.4507
	Platt	0.8907	0.8903	0.0914	0.0955	0.7523	0.7421	0.7669	0.7613	0.5904	0.5820
	Temperature	0.8614	0.8609	0.1090	0.1090	0.8073	0.7548	0.7651	0.7383	0.5886	0.5517
Predicted labels	None	0.8630	0.8630	0.1090	0.1090	0.7801	0.7801	0.7531	0.7531	0.5773	0.5773
	Histogram	0.8432	0.8442	0.1051	0.1084	0.7950	0.6642	0.7407	0.6763	0.5480	0.4601
	Platt	0.8907	0.8903	0.0914	0.0955	0.7461	0.7358	0.7637	0.7581	0.5852	0.5767
	Temperature	0.8614	0.8609	0.1090	0.1090	0.8152	0.7701	0.7690	0.7460	0.5951	0.5638

Table 3.2: The results were obtained for different probability calibration techniques. PCS, APCS, and MPCS are measured considering the ground truth (golden standard) or approximate ground truth (predicted labels). The AUC ROC and ECE metrics are always computed considering the ground truth. The best results are bolded. This table relates to Table 6 in the paper [35].

Source of truth	Calibration	AUC ROC ↑		wECE ↓		wPCS ↑		wAPCS ↑		wMPCS ↑	
		CS	CS+UD	CS	CS+UD	CS	CS+UD	CS	CS+UD	CS	CS+UD
Golden standard	None	0.8630	0.8630	0.1457	0.1457	0.7829	0.7829	0.7545	0.7545	0.5801	0.5801
	Histogram	0.8432	0.8442	0.1410	0.1449	0.8265	0.6494	0.7565	0.6689	0.5685	0.4505
	Platt	0.8907	0.8903	0.1230	0.1285	0.7655	0.7527	0.7735	0.7666	0.6010	0.5902
	Temperature	0.8614	0.8609	0.1457	0.1457	0.8232	0.7773	0.7730	0.7496	0.6014	0.5697
Predicted labels	None	0.8630	0.8630	0.1457	0.1457	0.7984	0.7984	0.7622	0.7622	0.5914	0.5914
	Histogram	0.8432	0.8442	0.1410	0.1449	0.8062	0.6692	0.7463	0.6788	0.5570	0.4638
	Platt	0.8907	0.8903	0.1230	0.1285	0.7649	0.7503	0.7732	0.7654	0.6008	0.5885
	Temperature	0.8614	0.8609	0.1457	0.1457	0.8221	0.7796	0.7725	0.7508	0.6014	0.5718

Table 3.3: The results were obtained for different probability calibration techniques. wPCS, wAPCS, and wMPCS are measured considering the ground truth (golden standard) or approximate ground truth (predicted labels). The AUC ROC and wECE metrics are always computed considering the ground truth. The best results are bolded. This table relates to Table 7 in the paper [35].

3.2 Complementary Results: Spearman Correlations and Their Statistical Significance

In this section, we present a table (see Table 3.4) reporting values of the Spearman correlation between metrics (ECE, PCS, and APCS_W). The results complement the correlations reported in the previous section in Table 5. Moreover, the results support the paper’s findings: a correlation exists between ECE and the proposed metrics. When computing the correlations, we always computed ECE considering the golden standard and the proposed metrics either with the golden standard or predicted labels as a source of truth.

The Spearman correlation is a non-parametric measure that measures the rank-order monotonicity relationship between two datasets. In our case, we had 200 paired values obtained by training five machine learning models (K-nearest neighbors, Naive Bayes, C45, Support Vector Machine, and Multi-Layer Perceptron) and four probability calibration techniques (no calibration, histogram, temperature scaling, and Platt calibration) in a ten-fold cross-validation setting. We performed hypothesis testing, considering the null hypothesis that the two metrics are linearly uncorrelated. Given the small sample size, we performed a permutation test. The results show a weak or moderate correlation between ECE and the proposed metrics exists. The low p-values suggest that a linear correlation exists between ECE and the metrics under consideration and that such a correlation is statistically significant, considering a p-value of 0.05.

From the results presented in Table 3.4, we observe the strongest correlations in the CS+UD case regardless of the source of truth. We, therefore, consider that when the calibrator is trained with additional soft labels, the proposed metrics capture similar information as the ECE metric.

Source of truth	Calibration data	Spearman correlation when compared against ECE			
		PCS	APCSW	wPCS	wAPCSW
Golden standard	CS	-0.3830	-0.3830	-0.3066	-0.3066
	CS+UD	-0.4544	-0.4544	-0.3880	-0.3880
Predicted labels	CS	-0.3608	-0.3608	-0.3768	-0.3768
	CS+UD	-0.3946	-0.3946	-0.3540	-0.3540

Table 3.4: Spearman correlation between metrics. For all correlations, the computed p-value was 0.0002. Therefore, we consider the correlations significant for a p-value of 0.05.

Chapter 4

Providing Explainability to Demand Forecasts

Explainable Artificial Intelligence is a subfield of artificial intelligence that aims to develop means and metrics to explain the outputs of artificial intelligence models and measure to what extent such explanations capture the rationale behind the output and how understandable it is to the subject it is aimed at. The more artificial intelligence models permeate every aspect of our lives, the more decisions are based on such model outcomes. Nevertheless, when deciding on a subject of relevance, it is important to understand why the model gives a given outcome, weigh it, understand how reasonable it is, and decide accordingly [44]. Such understanding can provide solid ground for responsible use of artificial intelligence. In this chapter, we advocate that comprehensive explanations should not only provide insights into the models' reasoning but also highlight characteristics of the data on which the model was trained, whether the current data is different from it, and insights into relevant context (e.g., are there some events that must be considered, to understand better what is going on or how a forecast can be affected?).

In this chapter, we introduce four papers devoted to explainable artificial intelligence. In none of them do we introduce novel techniques. We rather use existing techniques to enrich the explanations with external sources and provide a more comprehensive understanding of the context in which the forecast takes place. *Explainable Demand Forecasting: A Data Mining Goldmine* [45], was a short paper presented during the Ph.D. consortium of the WWW 2021 conference. It provides a brief insight into the challenges and opportunities of Explainable Artificial Intelligence in demand forecasting. In *Explaining Bad Forecasts in Global Time Series Models* [37] (published in MDPI Applied Sciences in 2022 (Q2 - IF 2.838)), we describe an application we built to provide comprehensive explanations for bad forecasts in global time series models framed as regression problems. To detect whether a forecast can be considered good or bad, anomaly detectors were used to determine whether a value should be considered anomalous given the sequence of values in the time series. The user is informed whether the forecast is considered anomalous and the criteria used to determine such an anomalous state. Furthermore, the user is enriched with relevant insights such as the most influential features to the forecast, what training instances most likely influenced the model to learn patterns leading towards a bad forecast, counterfactual examples that would lead to a different result, and values obtained from a simulation, that would probably make a better forecast. The third paper is titled *Knowledge graph-based rich and confidentiality preserving Explainable Artificial Intelligence (XAI)* [46] and was published by Elsevier in the Information Fusion journal (Q1 - IF 17.564) in 2022. The paper describes an approach that leverages mapping between features and an ontology to provide a semantic interpretation of the features and preserve confidentiality when in-

forming the most influential features towards a forecast. Furthermore, multiple sources of data are used to enrich the explanations, e.g., displaying media news about past events that can provide insights regarding the feature values at a given point in time and open datasets that could be used to complement existing models' information and therefore lead to better forecasts based on such an enrichment. Finally, in *Enriching Artificial Intelligence Explanations with Knowledge Fragments* [47] (published in MDPI Future Internet in 2022), we further iterate the abovementioned paper and compare a new semantic approach for ranking media news and datasets while also enriching the explanations based on data queried from knowledge graphs.

4.1 Explainable Demand Forecasting: A Data Mining Goldmine

In this subsection, we consider three papers (*Explainable Demand Forecasting: A Data Mining Goldmine* (presented during the Ph.D. consortium of the WWW 2021 conference), *Knowledge graph-based rich and confidentiality preserving Explainable Artificial Intelligence (XAI)* (published by Elsevier in the Information Fusion journal (Q1 - IF 17.564) in 2022), and *Enriching Artificial Intelligence Explanations with Knowledge Fragments* (published in MDPI Future Internet in 2022)) where information is derived from explainable artificial intelligence techniques and external data to provide additional context and insights to the planners consuming demand forecasts. The three papers describe how explainable artificial intelligence techniques determine the most relevant features of a particular forecast. Furthermore, a mapping between the features and concepts is used, and the relevance of the features is propagated to the concepts' relevance. The concepts aim to represent unique aspects considered in each feature. Moreover, features and concepts can relate at different abstraction levels. Concepts can hold additional information. E.g., whether they represent some aspect of the reality on which the decision-maker can take action to modify the forecast outcomes.

The concepts' relevance is used to explore information that could be helpful to the decision-maker. In particular, we searched for news media events that were reported in the past and could relate to the relevant concepts. We expect that showing such events can help the decision-maker better understand the past context and eventually find it helpful to assess the soundness of a given forecast. Concepts' relevance is also explored to query open datasets, which may contain relevant information to enhance the forecasting models. Such an approach could automate the search for new potentially relevant data sources for data enrichment while drawing attention to non-intuitive aspects that can widen the domain knowledge. Finally, we performed similar queries to the Google Knowledge Graph to provide additional insights based on relevant wiki concepts that emerged from related news media events.

Our research shows that most entries for news media events, the extracted keywords, and the Google Knowledge Graph were precise but lacked diversity. On the other hand, entries suggesting new datasets were more diverse but less precise. Nevertheless, our research lacked experts' feedback to assess to what extent the information displayed could be helpful in real-world operations. In future work, we would like to incorporate such feedback to enhance the content displayed to the user. Furthermore, we would like to develop new approaches to determine causality between events reported in media news and demand forecasting outcomes. We envision such information as complementary and even more helpful to decision-makers than the one we could display in this research.

Explainable Demand Forecasting: A Data Mining Goldmine

Joze M. Rozanec

joze.rozanec@ijs.si

Jožef Stefan Institute and Jožef Stefan International Postgraduate School

Ljubljana, Slovenia

Qlector d.o.o.

Ljubljana, Slovenia

ABSTRACT

Demand forecasting is a crucial component of demand management. Value is provided to the organization through accurate forecasts and insights into the reasons driving the forecasts to increase confidence and assist decision-making. In this Ph.D., we aim to develop state-of-the-art demand forecasting models for irregular demand, develop explainability mechanisms to avoid exposing models fine-grained information regarding the model features, create a recommender system to assist users on decision-making and develop mechanisms to enrich knowledge graphs with feedback provided by the users through artificial intelligence-powered feedback modules. We have already developed models for accurate forecasts regarding steady and irregular demand and architecture to provide forecast explanations that preserve sensitive information regarding model features. These explanations highlighting real-world events that provide insights on the general context captured through the dataset features while highlighting actionable items and suggesting datasets for future data enrichment.

CCS CONCEPTS

• **Computing methodologies** → *Information extraction; Machine learning approaches*; • **Applied computing** → *Forecasting*.

KEYWORDS

Explainable Artificial Intelligence (XAI), Demand forecasting, Lumpy demand, Artificial Intelligence (AI), Machine learning, Decision support systems

ACM Reference Format:

Joze M. Rozanec. 2021. Explainable Demand Forecasting: A Data Mining Goldmine. In *Companion Proceedings of the Web Conference 2021 (WWW '21 Companion)*, April 19–23, 2021, Ljubljana, Slovenia. ACM, New York, NY, USA, 2 pages. <https://doi.org/10.1145/3442442.3453708>

1 PROBLEM STATEMENT

Demand forecasting attempts to provide an early and accurate forecast of future demand to guide supply chain decision-making. It is considered a key component of supply chain management since it directly affects production planning and orders fulfillment: operational and strategic decisions are made on resources (allocation

and scheduling), workers (scheduling, training, and promotions), manufactured products (market share increase, production diversification) and logistics for deliveries.

The multiple facets of demand forecasting make it a data mining goldmine, providing multiple challenges to be solved. Among them, we may list: (i) create accurate forecasts at multiple time horizons and granularity levels; (ii) provide forecast explanations to assist decision-making; (iii) convey explanations in such a way that sensitive details of the model remain hidden from the end-users; (iv) create recommendations to users (e.g., on datasets that can enrich the forecasting model or decision-making options that can be taken in a given context); (v) gather users' feedback (e.g., regarding forecast explanations or decision-making options provided).

Each of the listed challenges can be a subject of multiple approaches, given the richness of demand characteristics, either by the type of goods or economic sector considered demand patterns and dynamics, or granularity levels of interest. Demand forecasting models for different time granularities require different approaches since demand patterns change: demands with regular occurrence (regular demand) at a monthly level can have a very sparse occurrence (irregular demand) when considered daily. Such differences require alternative approaches. While regression models are widely used to forecast regular demand, statistical, machine learning, and hybrid models were proposed to predict irregular demand. While such models struggle with data sparsity [1, 10], self-improving (e.g., ADIDA[4]) mechanisms were developed to remove data sparsity and provide a better forecast with regular regression methods. While removing sparsity simplifies demand size forecasting, it introduces an additional challenge: disaggregating the forecast. Sparse demand makes it hard to measure a demand forecasting model performance since zeros distort metric results[2]. Though alternative metrics were proposed, there is no consensus on a set of metrics to be used in such cases.

Regardless of the forecasts' quality, explanations are needed to help the user understand which variables had the most influence on a given forecast and increase confidence and trust in the forecasting model. Though researchers developed many techniques to understand which features are relevant to black-box models[6], feature relevance alone does make a good explanation. Good explanations should resemble a logic explanation[5], target a specific user profile[7], and focus on actionability[9]. The language and content of the explanation should be adapted based on the end-users profile. Such content adaptation must consider which details can be and how they should be disclosed to end-users to avoid providing insight into the forecasting models' sensitive details if not needed. Along with forecast explanations, decision-making options can be

This paper is published under the Creative Commons Attribution 4.0 International (CC-BY 4.0) license. Authors reserve their rights to disseminate the work on their personal and corporate Web sites with the appropriate attribution.

WWW '21 Companion, April 19–23, 2021, Ljubljana, Slovenia

© 2021 IW3C2 (International World Wide Web Conference Committee), published under Creative Commons CC-BY 4.0 License.

ACM ISBN 978-1-4503-8313-4/21/04.

<https://doi.org/10.1145/3442442.3453708>

provided to hint users about possible actions that can be taken and learn from their decisions.

Though much research was performed on what features explain certain forecasts, little research provides user evaluations of forecast explanations[8]. Such evaluation is required to obtain insights into their perceived quality and influence on decision-making. Such research gap urges to develop feedback mechanisms that allow collecting data on given explanations, which can improve them or identify forecasting model issues. Such feedback can also be provided for decision-making options. When available, they can help learn on new decision-making options not yet considered in the system, develop models to predict most likely decisions for a given context, or rank decision-making options based on different criteria and experience. Some of the challenges associated with these tasks are how to design artificial intelligence-guided feedback mechanisms to collect feedback data and build state-of-the-art ranking or predictive models for each case.

1.1 Research questions

The Ph.D. thesis seeks to provide solutions to a subset of research questions related to the problems described above:

- How does a compound model (classification model for demand occurrence and regression model for demand size) compare to the existing models used for forecasting irregular demand?
- What set of measures can be used to assess irregular demand forecasts, overcoming distortions due to demand events sparsity?
- Can we leverage ontologies to obfuscate forecasting model features when providing forecast explanations?
- Can we develop a methodology to assess explanations usability?
- How can we assess the quality of forecast explanations and decision-making options presented to the user?

2 PROPOSED APPROACH AND METHODOLOGY

We propose creating two models to forecast irregular demand: a classification model to predict demand occurrence and a regression model to predict demand size. To assess irregular demand forecasts, we chose three metrics: Area Under the Receiver Operating Characteristic Curve (AUC ROC) for demand occurrence, Mean Absolute Scaled Error (MASE) for demand size, and Stock-keeping-oriented Prediction Error Costs (SPEC[3]) for inventory. AUC ROC is widely used for classification problems and can provide a good estimate of how well the model predicts demand occurrence, regardless of the sparsity of events. MASE provides a magnitude agnostic measurement of the regression model performance. Finally, SPEC describes how forecasting errors impact inventory costs.

To provide forecast explanations without exposing model features, we use an ontology that describes a hierarchy of concepts related to the model features. The hierarchy is traversed to find common concepts among relevant features and inform them to the users. Our explanations also inform of potentially related real-world events reported in media and datasets that could enhance

the models. To assess the quality of explanations provided, we randomly sample and manually annotate them with a binary value (good/bad) and compute the proportion of good media events and datasets. We also assess their average precision@K.

3 FIRST RESULTS AND OUTLOOK

Our first results show that a compound model (classification for demand occurrence and regression model for demand size) provides state-of-the-art results for irregular demand compared to widely adopted models cited in the literature. Our model achieves the best performance across three metrics (AUC ROC, MASE, and SPEC) and outperforms the self-improving mechanism ADIDA for a subset of time-series. Through these metrics, we found that while the models we found in the literature adopt different strategies to consider demand occurrence, they fail at this task. Their demand size estimates mainly drive final performance differences. Regarding forecast explanations, we developed an architecture and tested it on a real-world use case. The architecture provides means to obfuscate relevant features by leveraging concepts we previously modeled in an ontology.

Our future research work will focus on (I) enhancing demand forecasting models' explainability and (II) develop a methodology to assess explanations usability.

ACKNOWLEDGMENTS

This work was supported by the Slovenian Research Agency and the European Union's Horizon 2020 program projects FACTLOG under grant agreement H2020-869951, and STAR under grant agreement number H2020-956573. We want to thank Ph.D. supervisors Dunja Mladenici and Blaž Fortuna for the patient guidance, encouragement, and advice since the beginning of the Ph.D.

REFERENCES

- [1] J Do Croston. 1972. Forecasting and stock control for intermittent demands. *Journal of the Operational Research Society* 23, 3 (1972), 289–303.
- [2] Raghad Hemeimat, Lina Al-Qatawneh, Mazen Arafeh, Shadi Masoud, et al. 2016. Forecasting spare parts demand using statistical analysis. *American Journal of Operations Research* 6, 02 (2016), 113.
- [3] Dominik Martin, Philipp Spitzer, and Niklas Kühl. 2020. A New Metric for Lumpy and Intermittent Demand Forecasts: Stock-keeping-oriented Prediction Error Costs. *arXiv preprint arXiv:2004.10537* (2020).
- [4] Konstantinos Nikolopoulos, Aris A Syntetos, John E Boylan, Fotios Petropoulos, and Vassilis Assimakopoulos. 2011. An aggregate-disaggregate intermittent demand approach (ADIDA) to forecasting: an empirical proposition and analysis. *Journal of the Operational Research Society* 62, 3 (2011), 544–554.
- [5] Dino Pedreschi, Fosca Giannotti, Riccardo Guidotti, Anna Monreale, Luca Pappalardo, Salvatore Ruggieri, and Franco Turini. 2018. Open the black box data-driven explanation of black box decision systems. *arXiv preprint arXiv:1806.09936* (2018).
- [6] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "Why should I trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 1135–1144.
- [7] Wojciech Samek and Klaus-Robert Müller. 2019. Towards explainable artificial intelligence. In *Explainable AI: interpreting, explaining and visualizing deep learning*. Springer, 5–22.
- [8] Jasper van der Waa, Elisabeth Nieuwburg, Anita Cremers, and Mark Neerincx. 2021. Evaluating XAI: A comparison of rule-based and example-based explanations. *Artificial Intelligence* 291 (2021), 103404.
- [9] Sahil Verma, John Dickerson, and Keegan Hines. 2020. Counterfactual Explanations for Machine Learning: A Review. *arXiv preprint arXiv:2010.10596* (2020).
- [10] Thomas R Willemain, Charles N Smart, and Henry F Schwarz. 2004. A new approach to forecasting intermittent demand for service parts inventories. *International Journal of forecasting* 20, 3 (2004), 375–387.



Contents lists available at ScienceDirect

Information Fusion

journal homepage: www.elsevier.com/locate/inffus

Knowledge graph-based rich and confidentiality preserving Explainable Artificial Intelligence (XAI)

Jože M. Rožanec^{a,b,c,*}, Blaž Fortuna^{a,b}, Dunja Mladenić^a

^a Jožef Stefan Institute, Jamova 39, 1000 Ljubljana, Slovenia

^b Qlector d.o.o., Rovšnikova 7, 1000 Ljubljana, Slovenia

^c Jožef Stefan International Postgraduate School, Jamova 39, 1000 Ljubljana, Slovenia

ARTICLE INFO

Keywords:

Explainable Artificial Intelligence
Knowledge Graph
Demand forecasting
Smart manufacturing
Confidentiality
Privacy

ABSTRACT

The paper proposes a novel architecture for explainable artificial intelligence based on semantic technologies and artificial intelligence. We tailor the architecture for the domain of demand forecasting and validate it on a real-world case study. The explanations provided result from knowledge fusion regarding concepts describing features relevant to a particular forecast, related media events, and metadata regarding external datasets of interest. The Knowledge Graph enhances the quality of explanations by informing concepts at a higher abstraction level rather than specific features. By doing so, explanations avoid exposing sensitive details regarding the demand forecasting models, thus preserving confidentiality. In addition, the Knowledge Graph enables linking domain knowledge, forecasted values, and forecast explanations while also providing insights into actionable aspects on which users can take action. The ontology and dataset we developed for this use case are publicly available for further research.

1. Introduction

The increasing digitalization of manufacturing in the context of Industry 4.0 provides a growing amount of data describing assets and operations. This data increases the transparency of production processes, accelerates the information flow through the company, and is a valuable asset to build predictive models. Demand forecasting is an essential component for any manufacturing company. A growing body of research has addressed techniques that identify demand-type and develop statistical and Artificial Intelligence (AI) models to predict future demand. In critical operations, such as demand forecasting, it is crucial to provide a forecast, the associated uncertainty and convey some explanations to the user. In this way, the user can make an informed decision, which may have far-reaching consequences, and avoid costly mistakes. Such explanations enhance the AI models, increase trust in the system, and help identify errors and performance issues.

Researchers developed several approaches related to AI models' explainability, dividing the models into glass-box (the AI algorithm can explain its prediction) and black-box (we need another algorithm to obtain an explanation of a model). Such explanations can be provided on a global level (the forecasting model in general) or at a local level (for every prediction instance). Some authors also envisioned that the inclusion of semantic technologies could be used to determine

the semantic closeness of concepts encoded in data features or integrate knowledge graph embeddings to a forecasting model to produce explanations along with the forecasts [1,2].

Explanation of an AI model should provide cues on factors that influenced a forecast, provide means to discern relevant context, supporting information, and emphasize actionable aspects to assist decision-making. Our research focuses on local-level explanations so that planners can still weigh cues provided for each case. We develop a domain-specific ontology and knowledge graph to provide domain knowledge and context for creating explanations. The knowledge graph allows us to identify which cues correspond to semantically close concepts and variables people may influence, supporting information from the dataset and external systems. In particular, we integrate a Media Event Retrieval System to match provided cues to relevant news and identify other concepts that may be relevant to the case. The system then proposes to the user such concepts and potential datasets that could be used to enrich the existing data.

The main scientific contributions of the presented research are the following:

1. a modular architecture to provide demand forecasting explanations, which consider semantic proximity, highlight factors that can be influenced by the user, relevant context, and opportunities for data enrichment;

* Corresponding author at: Jožef Stefan Institute, Jamova 39, 1000 Ljubljana, Slovenia.

E-mail addresses: joze.rozanec@ijs.si (J.M. Rožanec), blaz.fortuna@ijs.si (B. Fortuna), dunja.mladenic@ijs.si (D. Mladenić).

2. a unified ontology to capture domain knowledge regarding demand forecasting, events, and other elements relevant to the explanations;
3. the architecture implementation and validation on a real-world use case in manufacturing;
4. publicly available dataset containing: features relevant to forecasts, retrieved media events, metadata regarding external datasets, and a subset of annotated media events and external datasets metadata. Such a dataset can be a valuable resource for further research on identifying, ranking relevant, and summarizing entries to explain forecasts better.

To evaluate the overall architecture, we manually annotate a set of 528 media events, 401 datasets metadata entries, and 168 forecast explanations and assess their goodness based on metrics and criteria described in Section 6.

The rest of this paper is structured as follows: Section 2 presents related work, Section 3 describes the proposed architecture in the context of demand forecasting, Section 4 describes the ontology we used to construct the knowledge graph, Section 5 describes the use case for which we implemented the architecture, Section 6 provides the results we obtained, and Section 7 details some limitations and improvement opportunities regarding the work described in this paper. Finally, in Section 8, we provide our conclusions and outline future work ideas.

2. Related work

2.1. Demand forecasting

Demand forecasting is a crucial component for any manufacturing company. Accurate forecasts translate to production planning, influencing raw materials purchases, production line schedules, and inventory. It is thus essential that planners, who create the demand forecasts, understand the demand of the products they sell. The intrinsic properties of the product influence such demand. Among other factors affecting the demand, we must mention the market they target, the economic context, and customer expectations [3].

It has been recognized that AI models can play a pivotal role in enhancing demand forecasting [4–6]. To develop such models, it is critical to gather data from a wide range of sources that enable the model to learn patterns and correlations latent in the data. The increasing digitalization of industry, fostered by the decreased cost of sensors and connectivity [7], and the introduction of new technologies and paradigms (e.g., Industrial Internet of Things, Cloud Computing, Cyber-Physical Systems, or Digital Twins [8–11]), have produced an explosive growth of data in the manufacturing domain [12], which can be leveraged to develop such models. Most authors, thus, take advantage of enriching demand data with information regarding exogenous factors that can influence demand, achieving more accurate predictions. In particular, authors addressing demand forecasting in the automotive industry, reported using data regarding the unemployment rate [13], people's personal income [14], the inflation rate [15], fuel prices [16], or the Gross Domestic Product [17], among others.

Authors have observed that demand can adopt different patterns. Furthermore, they have developed criteria to categorize demand into specific demand types. For example, Williams [18] introduced the concept of demand variance during lead times, Johnston and Boylan [19] described the average demand interval (relationship between total time buckets and those with demand occurrence), and Syntetos et al. [20] introduced the concept of coefficient of variation (extent of demand size variability to the mean). Studies conducted by multiple researchers confirmed that demand types relate to demand distribution shapes [21,22].

While demand forecasting can be conceived as a time series forecasting problem, it can also be framed as a supervised regression learning

problem [23]. Doing so enables learning from past demand while considering additional information regarding exogenous factors or demand shapes described above. Among frequently used machine learning models for demand forecasting in the automotive industry, we find the Multiple Linear Regression [13,24], Support Vector Machine [13,25], and the Artificial Neural Networks [25–27].

It has been observed that AI models can outperform human forecasting. While human forecasts are subject to a wide range of biases, AI models are not prone to such biases and provide better means towards identifying systematic patterns and integrating a wide range of sources of information related to forecasting processes [28–31]. Nevertheless, the planners are responsible for making judgmental adjustments and record such adjustments when vital information about forthcoming events is available to them, but not to the forecasting models [28]. Furthermore, they are accountable for decisions made based on such forecasts.

Demand forecasts with their associated uncertainty do not provide enough ground for planners' decision-making. While authors consider that justifying decisions is a precondition for responsible decision-making [32], and such justifications will be considered ever more critical in the future [33], it has been shown that explanations enhance users' acceptance and satisfaction, and provide means to understand if causality is captured by the model [32]. Furthermore, model explanations have been found to influence human decision-making [34]. Therefore, providing such explanations to planners helps them understand the reasons driving such a forecast (e.g., features' contribution to the forecast [35]). Furthermore, enriching such explanations with domain context [32] enables the planners to evaluate the forecasts' soundness, make sure they agree on it, and eventually correct the forecast [36] before making a decision [37]. Such explanations can be achieved through different means, studied by the subfield of AI known as Explainable Artificial Intelligence (XAI).

2.2. XAI: black-box models and semantic technologies

Human-understandable explanations in the demand forecasting domain are essential as they help the planner to (i) understand which of the model attributes are strong drivers in determining the expected demand, (ii) identify some bias in the model, and (iii) understand how the model can be improved.

Several taxonomies have been proposed to classify the XAI techniques and ease their study. Adadi and Berrada [38] proposed classifying the XAI methods into three categories, based on the complexity of the method (the degree up to which the method is interpretable), their scope (local or global), and whether the methods are specific to a model or agnostic. Inherently interpretable models are known as *white-box models*, while opaque models are known as *black-box models* [39]. A similar classification was developed by [40], who divided XAI approaches into three categories: scope (whether the explanation targets a single forecast or attempts to explain the whole model), methodology (if focused on the input data or model parameters), and usage (if integrated to the model or applied to any model in general). Liao et al. [41] divided XAI techniques into four categories, whether they explain the model (global), explain the forecast (local), inspect the counterfactual, or provide an example. Finally, Angelov et al. [42] proposed a wider taxonomy, taking into consideration whether the explanation is local or not if the models are transparent or opaque, if the techniques are model-specific or model-agnostic, and whether explanations are created by simplification, conveyed through visualizations, or based on feature relevance. In this research, we focus on black-box explanation methods, since (i) white-box models, due to their complexity, are not intelligible to most users who are not AI experts [37], and given that (ii) black-box models can be applied to any kind of AI models [35]. By doing so, we decouple the forecast and explainability dimensions and provide greater flexibility in choosing the best approach for each of them.

Most of the XAI approaches for time series were developed for deep learning models [43]. Those approaches are conditioned by the deep learning model's architecture, which determines how they extract features from data and learn, thus requiring different ways to convey such information to the users. One such method is the *Gradient*Input*, applied to convolutional networks, which computes the partial derivative of a layer against the input, and multiplies the result by the input, to obtain an insight into how do the filters activate [44]. ConvTimeNet [45] follows another approach occluding time series sub-segments to measure how the forecast is affected when compared against one produced by the original sequence, and thus understand the contribution of each time series' sub-sequence to the forecast. Tonekaboni et al. [46] leveraged the same concept. However, instead of occluding time series sub-segments, he replaced them with synthetically generated ones. In the same line, Ozyegen et al. [47] suggests replacing time series' sub-segments with a local or global mean, while Mercier et al. [48] developed a novel loss to generate such patches using a patch generative network. In recurrent neural networks, insights provided by an attention mechanism are considered a source of explainability since they provide information regarding the temporal dynamics learned by the model [49].

Time series forecasting can also be framed as a supervised learning problem [23]. Doing so enables using a different set of explainability techniques appropriate for supervised learning models. Among most frequently cited posthoc model-agnostic and feature-relevance technique for local explanations [40,42] we find LIME [50], and its variants (e.g.: k-LIME [51], DLIME [52], and LIMETree [53]), Anchors [54], Local Foil Trees [55], or LoRE [56]. These approaches build surrogate models for each prediction sample, learning the reference model's behavior on the particular case of interest by introducing perturbations to the feature vector variables. By doing so, they can provide a local feature importance estimate, which is considered an indirect method to explain a model [32]. In particular, we opted for the feature relevance explanation above other post-hoc model agnostic techniques (e.g., rule extraction) due to the simplicity of implementation. While other explanations can also be built considering the ontology abstraction, care must be put to ensure these remain valid. E.g., in the case of rule extraction, the non-contradiction principle must be enforced since while the rule can be valid at a feature level, the same is not guaranteed at higher abstraction levels. While such approaches do not consider the temporal dimension *per se*, features' metadata can be considered to capture it and enrich the forecast explanations [57].

When designing a system providing explanations for AI models, it must be considered the explanations must serve multiple stakeholders (and thus target different user profiles [58]), serve different purposes, and that their effectiveness must be assessed quantitatively (through algorithmic validation) and qualitatively (estimating user satisfaction, trust, and other factors) [59]. Good explanations should convey meaningful information, resemble a logic explanation [60], focus on actionability [61], and if possible, provide some counterfactuals. Since different explanations on the reasons behind a models' forecast can be deduced depending on the users' background knowledge, the system must provide enough information to the user so that the user can unequivocally understand the models' rationale and enable users' responsible decision-making [62]. Given the recipient-dependent nature of the explanations, it is important that those are designed following a human-centered approach [41,63]. When focusing on a specific user profile, it is important to adapt the language and content of relevant information that should or can be shown to it [64]. The actionable dimension relates to the ability to distinguish mutable features that can be influenced by the user [65,66].

Multiple authors envisioned the usage of semantic technologies in the explainability domain. Doctor XAI [2] develops an agnostic XAI technique for ontology-linked data classification by training a surrogate model and extracting rules from it. Gaur et al. [1] make use of a Knowledge Graph to feed deep learning models with it to enhance

their explainability. Samek and Müller [58] envision Knowledge Graphs can be used to compact large tree models by combining nodes into unique probabilistic concepts. In addition to the opportunities mentioned above, we consider semantic technologies shall (A) provide background knowledge which can be leveraged to provide semantic meaning to dataset features, (B) inform their characteristics (e.g., valid ranges, if they are mutable or immutable if the user can influence their values), or (C) issue explanations with adequate language and context, considering if certain information shall be or not informed to the user.

A modular architecture that decouples forecasts from their explanations is required to provide the planners with adequate explanations regarding machine demand forecasts. Such architecture has the advantage of allowing to iterate both components at their own pace. Black-box explainability models mostly lack domain knowledge and inform relevant forecasting features without a broader context and a more profound interpretation. A third module can provide such context and interpretation, taking into account black-box explanations and domain-knowledge encoded in an ontology and instantiated in a Knowledge Graph to create a better explanation for the end-user.

2.3. Domain specific ontologies

Demand Forecasting is an essential activity in manufacturing, and manufacturing domain ontologies provide some related concepts. E.g., Ameri and Dutta [67] developed the Manufacturing Service Description Language ontology as a formal representation of manufacturing services, and Lemaignan et al. [68] described knowledge regarding individual operations in MASON.

Since our architecture links concepts from datasets used to train the AI models that issue the forecasts, this should also be encoded in the unified ontology. To promote the reuse of concepts, following the MIREOT principle [69], we analyzed ontologies related to the artificial intelligence domain. Cannataro and Comito [70] developed the DAMON (Data Mining Ontology for Grid Programming), which provides a reference model for data mining tasks, methodologies, and available software. A heavyweight ontology was developed by Panov et al. [71,72], who provide means to represent data mining entities, inductive queries, and data mining scenarios. Diamantini et al. [73] developed KDDONTO, focusing on data mining algorithms discovery.

The explanation must provide information regarding features that are most influential to a forecast. This view shall be complemented with context, describing events that illustrate possible reasons behind the values captured in the features and point to similar events observed in the present. Such events shall help the planner draw conclusions that exceed the mere forecast provided. The concept of *Event* can be found in different ontologies [74,75], having a prominent place in the YAMATO ontology [76] as one of the core concepts.

The ontologies mentioned above provide a solid ground to create a unified ontology that serves the purpose of semantically enhancing explanations of demand forecasts issued by AI models.

2.4. Methodologies for ontology design in manufacturing

Multiple general methodologies were proposed to build an ontology. Uschold [77], Uschold and King [78], Fernández-López et al. [79] suggest first identifying the purpose of the ontology, and Uschold [77], Kim et al. [80] stress the importance of defining its scope and level of formality. Next, Kim et al. [80] suggests defining a problem statement and competency questions, which provide common ground to elicit knowledge from multiple sources, identify key concepts and relationships. When defining terms to be used in the ontology, an effort should be made to integrate pre-existing ontologies where possible.

Some authors identified steps specific to the creation of manufacturing domain ontologies. Ameri et al. [81] developed a four-step methodology using a Simple Knowledge Organization System framework to develop a thesaurus of concepts, and then identify relevant classes and

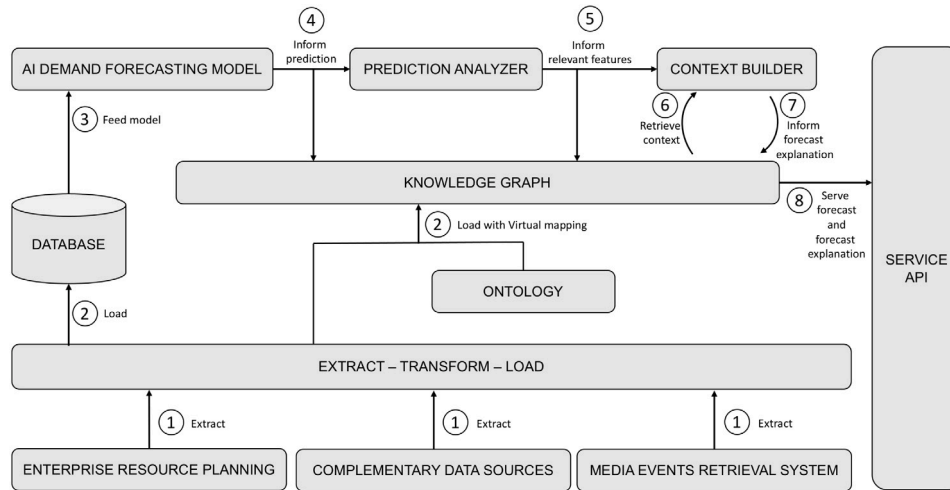


Fig. 1. Semantic XAI architecture for demand forecasting.

provide logical constraints and rules. A similar approach was developed by Chang et al. [82], with an emphasis on manufacturing design.

The methodologies we mentioned above provide us valuable guidance when developing our domain-specific ontology, aiming to integrate domain-specific knowledge regarding demand forecasting with the demand forecasting models, predictions, media events, external datasets, and explanations.

3. The proposed architecture

It has been noted in the literature that AI explanations can be enhanced by complementing insights on models' inner workings with domain knowledge [83,84]. Furthermore, the XAI's ability to explain models can be used to enhance data fusion capabilities, discovering relevant data sources that further enhance the model and deepen the users' domain understanding [32]. However, we found no author addressed the intersection between data fusion, confidentiality, and model explainability from the literature we reviewed. We thus propose a modular architecture to provide forecast explanations that address the challenges mentioned above. The main novelty is combining semantic technologies and media events to build the context and provide an informed prediction explanation while also recommending additional data sources that could be used to enrich the existing dataset.

3.1. Architecture overview

The proposed supporting architecture enables us to realize the explanations for each prediction. The architecture integrates predictions, gathers insights on relevant features, incorporates domain knowledge and context to each prediction, and provides a forecast explanation to the end-user. The architecture (see Fig. 1) comprises the following components:

- **Enterprise Resource Planning:** software used by the manufacturing company to keep track of operations at different levels. For the specific use case of demand forecasting, it provides data regarding strategic sales planning, past and open sales, buyers risk assessment, and past demand.
- **Complementary data sources:** sources of data that provide valuable complementary data to train demand forecasting models.
- **Media Event Retrieval System:** is a system that keeps track of events reported in the media. Media Event Retrieval System uses Natural Language Processing to identify entities and concepts reported in media news and identify which news report about the same events. Wikification is applied to identify concepts, cluster news, and link events.

- **Ontology:** encodes domain knowledge regarding demand forecasting, demand forecasting models, events retrieved from the Media Event Retrieval System, and available datasets. The Extract-Transform-Load module uses it to guide a virtual mapping procedure and instantiate the Knowledge Graph.
- **Knowledge Graph:** is the instantiation of the ontology, based on data provided by the Extract-Transform-Load module and outcomes informed by multiple architecture components. The Knowledge Graph keeps track of data up to a configurable time window.
- **Extract-Transform-Load:** provides means to interface with the Enterprise Resource Planning and Media Events Retrieval System to retrieve relevant data, transform it as required, and ingest it to a database or into the Knowledge Graph. A virtual mapping procedure specifies how data pieces are related to concepts defined in the ontology and how they should be instantiated to the Knowledge Graph. The Extract-Transform-Load module consists of a series of batch processes. The processes are executed regularly to ensure that the KG and database information is updated regarding the Enterprise Resource Planning software and Media Event Retrieval System.
- **Database:** stores data relevant to the AI models, which can be used to train them.
- **AI Demand Forecasting model:** such a model can be built with a variety of algorithms, depending on the demand type and data available. Such models can be machine learning regression [13, 85], deep learning [4,6] or probabilistic models [86]. Demand forecasting models shall provide a demand estimate for future points in time together with some uncertainty estimation. This information is stored in the Knowledge Graph.
- **Prediction Analyzer:** interfaces with the AI Demand Forecasting model to determine which features are most important to a given prediction. To that end, it may use different black-box explainability techniques to assess relevant features under different criteria. The analysis outcomes are stored in the Knowledge Graph.
- **Context Builder:** has the responsibility to gather required pieces of data to issue good explanations. Interfaces with the Prediction Analyzer to get relevant features to each prediction. With the Knowledge Graph to obtain context regarding feature values, related events reported in the media, and complementary data that may be used to enhance the model. The Knowledge Graph also provides information regarding high-level concepts that describe the features (see Fig. 3). These are used to inform relevant features at a higher abstraction level, avoid exposing model features, and enable more concise explanations.

Table 1

Data samples retrieved from the Media Event Retrieval System (MERS), EU Open Data Portal, and the Enterprise Resource Planning (ERP) software. While the actual entries contain more fields and data, we summarized them to the most relevant fields. While the names of the ERP fields are accurate, the row of data we display does not correspond to a genuine entry.

Data source	Data type	Data sample
MERS	JSON	{ 'articleCounts': {'eng': 15}, 'categories': [{'label': 'news/Business', 'uri': 'news/Business', 'wgt': 92}, ...], 'concepts': [{'label': {'eng': 'Car finance'}, 'score': 100, 'type': 'wiki', 'uri': 'http://en.wikipedia.org/wiki/Car_finance'}, ...], 'eventDate': '2019-10-01', 'location': None, 'relevance': 4, 'sentiment': 0.06666666666666665, 'summary': {'eng': 'About a third of auto loans for new vehicles taken in the first half of 2019 ...'}, 'title': {'eng': 'The Seven-Year Auto Loan: America's Middle Class Can't Afford Their Cars'}, 'totalArticleCount': 15, 'uri': 'eng-5135186', 'wgt': 307584000}
EU Open Data Portal	JSON	{ 'result': {'results': [{'dataset': { 'identifier_dcterms': [{'datatype': None, 'lang': None, 'type': 'literal', 'value_or_uri': 'http://data.europa.eu/89h/jrc-eplca-744e255f-3b81-4eed-b2d3-7794353e0efb'}], 'identifier_dcterms': [{'datatype': None, 'lang': 'en', 'type': 'literal', 'value_or_uri': 'Raw data for polymerization and intermediate products ...'}], 'distribution_dcat': [{'accessURL_dcat': [{'uri': 'http://eplca.jrc.ec.europa.eu/ELCD3/'}], 'format_dcterms': {'uri': 'http://publications.europa.eu/resource/authority/file-type/ZIP'}, 'license_dcterms': [{'uri': 'http://publications.europa.eu/resource/authority/licence/OP_DATPRO'}], ...}]}] }
ERP	CSV	VBELN,POSNR,MATNR,LFIMG,WADAT_IST 0000000000,000010,0000000000000001111,100.000,20190909

- **Service API:** serves predictions and their explanations to the end-users. It hides explanation details based on the end-user profile to expose only facts that are relevant to the user.

Among the modules described above, three are data sources from which data is extracted. In Table 1 we provide data samples that correspond to each of them.

The proposed architecture considers several sources of information to build the explanations (e.g., features relevance, their mapping to concept abstractions provided by a local ontology, relevant media events, and suggestions regarding potentially beneficial datasets to be used for further data enrichment). We consider that a similar integration could be followed to extend the information sources to different semantic databases and federated ontologies. While multiple link discovery and ontology matching approaches have been studied in the literature [87–90], we consider at least two complementary ways that could be applied in our case to execute such integration. First, external semantic databases and federated ontologies could be integrated based on a common ground established through wikified concepts, wikifying the keywords associated with each feature [91,92]. An alternative could be to consider embeddings semantic meanings and integrate the sources based on concepts' embedding proximity. The embeddings approach could be used either (a) to create mappings from the model features to external ontologies and semantic databases using text embeddings [93–95], or (b) to automatically align instances from the local ontology to the semantic databases and federated ontologies using text or graph embeddings [96,97].

3.2. Context building

The goal of the architecture is to support semantically enhanced explanations for demand forecasting models. A key role is played by the *Prediction Analyzer* and *Context Builder* components. The *Prediction*

Analyzer provides an abstraction layer on top of black-box explainability models to inform on features that are most important to certain predictions. As mentioned in Section 2.2, good explanations should convey useful information, target a specific user profile, and focus on actionability. In this line, the *Context Builder* holds the responsibility to provide explanations that (I) convey the predicted demand and uncertainty, (II) provide insights on what concepts were considered to create the prediction, (III) provide relevant context, and (IV) target a specific user. The predicted value and the associated uncertainty are queried from the Knowledge Graph.

To realize (II), the *Context Builder* interfaces with the *Prediction Analyzer* and the *KG* to obtain information regarding relevant features to certain prediction. These features are mapped to a hierarchy of attribute concept abstractions (see Fig. 3). At least one level of abstraction is considered to avoid exposing sensitive information regarding features used in the demand forecasting model and provide more concise explanations. To determine the level of abstraction exposed to the user, we followed the heuristic from Listing 1. The procedure could be further enhanced by weighting the concepts based on the underlying features' individual contribution towards the particular forecast. It is important to highlight that attribute concept abstractions are considered actionable if and only if all of their underlying concepts are actionable.

Listing 1: Algorithm used to compute attribute abstractions.

```
# ontology: an ontology with attribute concept abstractions.
# relevant_features: list of relevant features to the given forecast
# n_concepts: maximum number of concepts to be shown to the user
given ontology, relevant_features, n_concepts

def get_abstraction(ontology, concept_id):
    abstraction = ontology.get_parent(concept_id)
    if abstraction == None:
        return concept_id
```

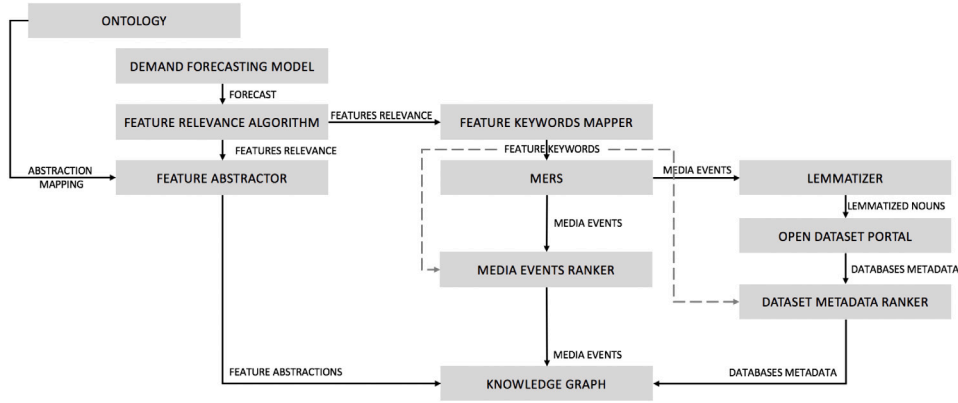


Fig. 2. Flow diagram detailing the procedure followed to provide insights on what concepts were considered to create the prediction and enrich it with relevant context (media events and suggested external datasets).

```

return abstraction

def include_concepts_until_desired_limit ( abstractions_set ,
    sorted_candidate_concepts, n_concepts):
    idx = 0
    while abstractions_set.length() < n_concepts and idx <
        sorted_candidate_concepts.length():
        abstractions_set.add(sorted_candidate_concepts[idx])
        idx++

def compute_feature_abstractions(ontology, relevant_features, n_concepts):
    feature_abstractions = [get_abstraction(ontology, feature) for feature in
        relevant_features]

    final_abstractions = set()
    sorted_commons = Counter(feature_abstractions).most_common()

    most_relevant_concepts = [x[0] for x in sorted_commons if x[1]>1]
    remaining_relevant_concepts = [x[0] for x in sorted_commons if x
        [1]==1]

    include_concepts_until_desired_limit ( final_abstractions ,
        most_relevant_concepts, n_concepts)
    include_concepts_until_desired_limit ( final_abstractions ,
        remaining_relevant_concepts, n_concepts)

    return final_abstractions

compute_feature_abstractions(ontology, relevant_features, n_concepts)

```

To realize (III), it is necessary to consider specific data used to build the feature vector, related events, and additional knowledge we may gain from them. To that end, we provide a set of keywords that describe each feature. The Extract-Transform-Load process considers the keywords to issue queries against the Media Event Retrieval System. To do so, we consider the point in time each feature in the feature vector refers to. Based on it, we associate related events that are temporally close to it. Natural Language Processing techniques are applied to retrieved media events to extract novel keywords relevant to the demand forecasting problem. Those new keywords are used to query for external datasets that could enrich the existing demand forecasting model. Media events, keywords we extracted from them, and metadata regarding external datasets are persisted to the Knowledge Graph and linked to specific features from feature vector instances. We summarize the procedure described above in Fig. 2.

Finally, (IV) is realized by providing a multi-part explanation, where components are displayed or hidden, based on what information is relevant to the end-user or confidentiality policies. This responsibility is delegated to the *Service API*. E.g., a target audience could be planners interested in the demand forecast, associated uncertainty, and some

context regarding feature and media events related to them. Another target audience could be experts who develop demand forecasting models. They would benefit from the data mentioned above and the possibility to drill down from described concepts to specific model features. Such an explanation could be enriched with insights on frequent keywords found in related events reported in the media and open datasets that may be useful to the model.

4. Ontology

Following related work elaborated in Section 2.3, we created an ontology that describes concepts related to demand forecasting. Specifically, it describes the use of data and AI models to predict demand, demand forecasts, and forecast explanations. The forecast explanations provide context regarding feature relevance, potentially related media events, and external datasets that could be used to enrich future models. The ontology was published and is available online [98].

Upper ontologies provide a guide on how to think about the target domain when building a domain ontology. We use BFO as our upper ontology, which supports a world's snapshot view. Our entities can be of two types: *occurent* (entities that occur) or *continuent* (entities that persist in time).

Regarding manufacturing, we reused the concepts of *Product*, and *Event* from the ontologies developed by Leitão and Restivo [99], Borgo and Leitão [100], and Kourtis et al. [101]. In Cannataro and Comito [70], Panov et al. [72], we found the concepts of *Dataset Specification*, *Dataset*, *Algorithm*, *Datamining Algorithm*, *Regression Algorithm*, *Predictive Model*, *Regression Model*, and *Prediction*. Gottschalk and Demidova [75] provided us the concept of *Information Provenance* to represent data sources. Finally, we introduced the following concepts:

- *Media Reported Event*: relates to some *Event* that was reported in news media;
- *Media Reported Event Keyword*: keyword obtained from some *Media Reported Event*;
- *External Dataset Metadata*: a piece of metadata describing some external dataset;
- *Forecast Explanation*: a description providing reasons for a certain forecast. Relates to a specific *Prediction*, *Media Reported Event*, and *External Dataset Metadata*;
- *Attribute*: a quantity describing an instance;
- *Attribute abstraction*: a high-level concept describing some aspect of a dataset attribute;
- *Feature Vector*: a vector containing attribute values describing certain an instance of data;

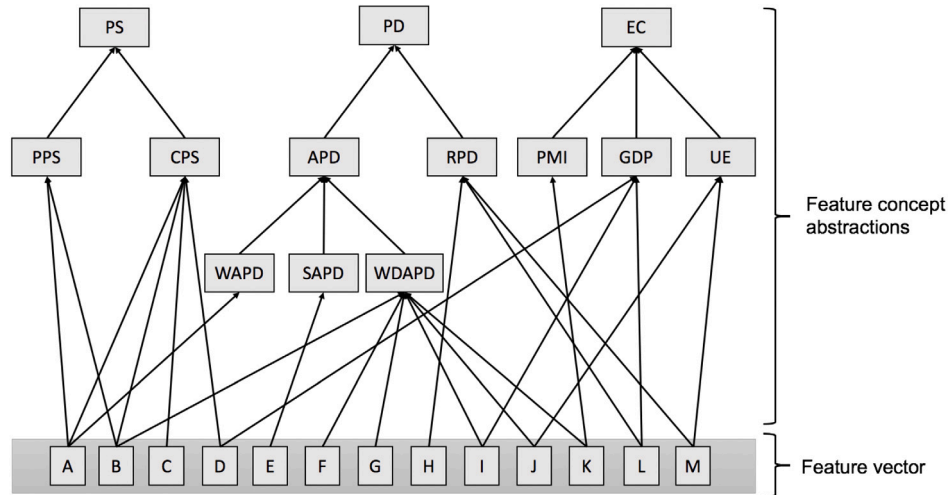


Fig. 3. Feature vector attributes and high-level concepts hierarchy associated with them. Considered attribute abstractions are PS (Planned Sales), PPS (Past Planned Sales), CPS (Current Planned Sales), PD (Past Demand), APD (Adjusted PD), SAPD (Scaled APD), WAPD (Weighted APD), WDAPD (Working Day APD), RPD (Raw Past Demand), EC (Economic Context), PMI (Purchasing Managers' Index), GDP (Gross Domestic Product), UE (Unemployment Rate). Features A-M are defined in Table 2.

Table 2

Sample feature vector and keywords considered for the use case. We use the following abbreviations: *sp* (sales plan), *GDP* (Gross Domestic Product), *PMI* (Purchasing Managers' Index), *wdp* (monthly demand averaged per amount of working days in that month), *demand* (demand in a given month), *lagNm* (N months lag), *pastwavg* (weighted average for a given month in past years), *scaled* (value scaled between 0–1).

Actionable?	Feature ID	Feature definition	MERS query keywords
YES	A	$sp \cdot \frac{demand_{pastwavg}}{sp_{pastwavg}}$	car sales demand new car sales vehicle sales car demand automotive industry
	B	$wdp_{lag12m} \cdot \frac{sp}{sp_{pastwavg}}$	Same as A
	C	<i>sp</i>	Same as A
	D	$sp \cdot \frac{GDP_{lag3m}}{GDP_{lag15m}}$	global GDP projection global economic outlook economic forecast + keywords listed in A
NO	E	$demand_{lag3m-scaled}$	Same as A
	F	wdp_{lag3m}	Same as A
	G	wdp_{lag8m}	Same as A
	H	$demand_{lag3m}$	Same as A
	I	$wdp_{lag3m} \cdot \frac{GDP_{lag3m}}{GDP_{lag15m}}$	Same as D
	J	$wdp_{lag5m} \cdot \frac{UE_{lag3m}}{UE_{lag15m}}$	unemployment rate unemployment numbers unemployment report employment growth long-term unemployment + keywords listed in A
	K	$wdp_{lag12m} \cdot \frac{PMI_{lag3m}}{PMI_{lag4m}}$	purchase managers' index + keywords listed in A
	L	$demand_{lag3m} \cdot \frac{GDP_{lag3m}}{GDP_{lag15m}}$	Same as D
	M	$demand_{lag4m} \cdot \frac{UE_{lag3m}}{UE_{lag15m}}$	Same as J

5. Case study

Among the related research presented in Section 2, we did not find scientific contributions focused on the explainability of AI demand forecasting models. The architecture we present in this research aims to provide the main building blocks required to provide adequate explanations for demand forecasts at a local level. Furthermore, it has been noted that XAI is foreseen to play a critical role in preserving confidentiality and that its ability to extract knowledge from AI models can be used to discover new relevant sources of knowledge [32].

We tackle both challenges, preserving confidentiality by obfuscating features relevance information behind a domain-specific ontology and using XAI insights to search for databases that could enrich the existing dataset.

To build and test the architecture, we worked with real-world data related to the automotive industry, provided by partners from European Horizon 2020 FACTLOG and STAR projects. The partners provided data for 56 different materials. Since time series forecasting can be framed as a regression problem [23], we did so to take advantage of exogenous sources and leverage data fusion to create a single dataset. Among

Table 3
Data sources grouped by their purpose.

Purpose	Data	Source
Demand Forecasting	Car Sales	International Organization of Motor Vehicle Manufacturers
	Copper Price	London Metal Exchange
	Crude Oil Price	World Bank
	Demand (deliveries)	Provided by EU H2020 FACTLOG and STAR partners
	Gross Domestic Product	World Bank
	Purchasing Managers' Index (PMI)	Institute of Supply Chain Management
	Sales Plan	Provided by EU H2020 FACTLOG and STAR partners
	Unemployment Rate	World Bank
Potentially relevant datasets	Databases Metadata	EU Open Data Portal
Media events relevant to XAI	Media Event Entries	Event Registry

the exogenous sources, we considered three relevant to the automotive sector (GDP, unemployment rates, and oil price), and three suggested by experts (Purchasing Managers' Index (PMI), copper prices, and sales plans). The PMI index summarizes purchasing managers' expectations regarding how the market will behave in the future (e.g., if it will expand, contract, or stay the same, and how strong the growth or contraction are expected to be). Copper prices are relevant to this specific demand forecasting scenario since the product prices are tied to the copper price, and thus they indirectly influence the products' demand. Finally, we incorporated the sales plans created on a quarterly and yearly basis by the strategic sales department, given the experts' suggestions and claims in the scientific literature on how do purchase intentions contribute to the forecasts' accuracy [102,103]. We summarize the data sources by their purpose in Table 3.

We considered a window of three years of data to build demand forecasting models for them. We issued predictions over the last six months of data and built explanations for all materials in the last three months. Our demand forecasting models use the Support Vector Regression algorithm [104]. We used nested cross-validation [105] to evaluate the models and make sure to use the latest data available for each target month. The models achieve an R^2_{adjusted} of 0,9212 and a Mean Absolute Scaled Error [106] of 0,2600. Finally, we used LIME [50] to implement the *Prediction Analyzer*.

The *Prediction Analyzer* provides a ranking of features for each forecast. For each of those features, we consider the query keywords presented in Table 2 to query the Media Events Retrieval System and retrieve media events that provide context to feature values. We used Event Registry [107], a well-established Media Events Retrieval System that has monitored mainstream media since 2014. Our queries consider the keywords that describe each feature and the point-in-time in which each feature component took place. E.g., suppose a feature considers (I) GDP values three and fifteen months before the forecasting horizon and (II) demand four months before the forecasting horizon. In that case, the query will ask (I) for media events related to GDP at three and fifteen months before the forecasting horizon and (II) media events regarding demand four months before the forecasting horizon.

We considered only media events reported in the English language. Once obtained, we processed them to identify and lemmatize nouns that provide insights on the most important facts related to the features. We also use these nouns to query for open datasets that may enrich the current model. In particular, we queried the EU Open Data Portal [108], a portal that provides access to open data published by EU institutions and bodies. For each matching external dataset metadata entry (provided or translated to the English language), we processed their title and description to create a word embedding with the Word2Vec algorithm [109], using Google News 300 pre-trained model [110]. Next, we create another embedding with the feature keywords from Table 2, and the keywords obtained from the media events we retrieved. To rank datasets metadata relevance, we compute the word movers distance [111] between both embeddings. Finally, to increase the diversity of displayed media event entries and datasets' metadata, we random sample them among top candidates for each case.

To create the ontology, we used Protege [112]. The ontology served as a guideline to implement virtual mapping functions to map data to the knowledge graph. We implemented the knowledge graph in Neo4j [113].

We present a sample demand forecast explanation in Fig. 4. The explanations display the following information:

- forecasted demand and associated uncertainty, for a given material at a certain point in time;
- main factors driving the forecast, which are obtained considering most important features to the forecast as provided by the *Prediction Analyzer*, and abstracted using domain knowledge from the *Knowledge Graph*;
- a highlight regarding an actionable aspect, if it exists;
- media events associated with the most important features of the particular forecast instance;
- media events' keywords frequently found in media events related to the forecast instance;
- external dataset that may be used to enrich the existing demand forecasting model.

While the user interface lists the primary factors driving the forecast, sorted based on the features' relevance and the abstractions' occurrence frequency, it does not provide insights on their relative impact on the forecast. However, the abstractions' relative impact can be computed considering the features' relative impact while considering the features' correlation and the abstractions' occurrence frequency. Computing and informing the abstractions' relative impact on the forecast remains a subject of future work.

6. Evaluation

In our research, we develop an architecture that aims to provide explanations for demand forecasting. Since media events play an essential role in our forecast explanations, we were interested in assessing (A) if media events describe features context and (B) if retrieved external datasets were related to the features for which we queried them. Regarding the explanations, we assessed metrics related to listed external datasets, events, and keywords to understand if the provided entries were accurate and diverse. To measure diversity, we computed the ratio of diverse entries (RDE) defined in Eq. (1). It is important to note that media events, keywords, and external datasets listed on explanations do not repeat themselves in a single explanation. However, the same media event, keyword, or external dataset reference may be found in another explanation. We consider that while explanations for similar data points are expected to be similar (e.g., provide similar feature ranking), multiple media events likely report a specific aspect (e.g., economic growth or job losses), or many datasets can be good candidates for data enrichment. Thus, we consider the best possible case is to find completely different but accurate media events, keywords, and external datasets across all explanations. Such a scenario would maximize end-users' learning.

In particular, for listed external datasets, we computed the accuracy and RDE. At the same time, we measured the average precision@K

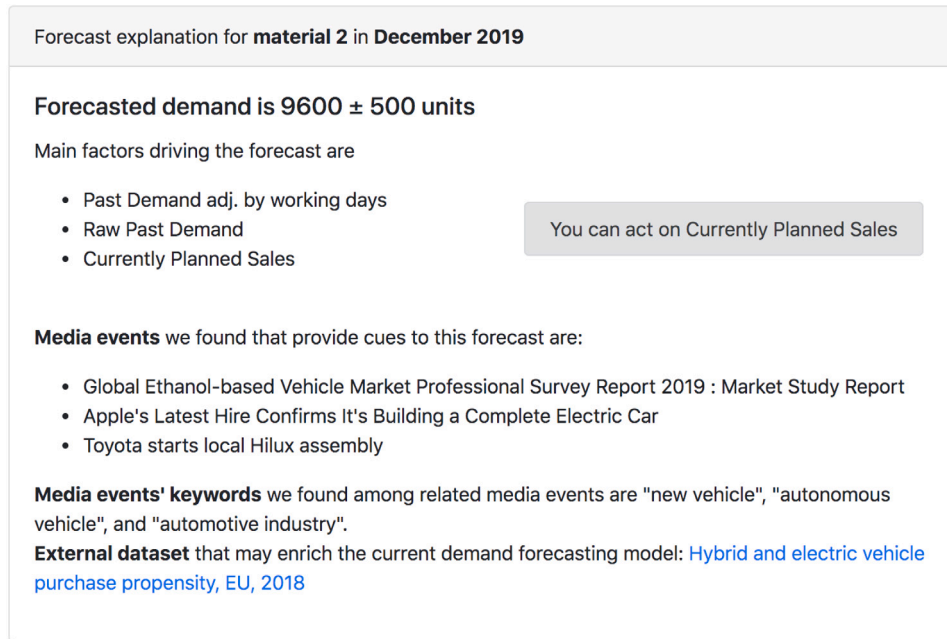


Fig. 4. An example of a demand forecast explanation that is displayed to the end user based on data retrieved from the *Service API*.

and RDE@K with K=1,3 over all of the 168 forecast explanations for media events and related media events keywords. We selected K=1,3 since three media events and media events' keywords are listed in each explanation. The choice regarding listing at most three media events and media events per explanation was made to avoid cluttering the user interface with too many features. The limit is arbitrary and could be changed if such a need arises. Average precision@K is computed as the average of the precision@K we computed for each forecast explanation. RDE@K is computed as RDE across all 168 explanation entries, considering only the first K listed media event or media events' keywords entries.

$$RDE = \frac{\text{Unique Entries}}{\text{Total Listed Entries}} \quad (1)$$

We evaluated these aspects by manually annotating a random sample of retrieved media events, retrieved external datasets metadata, and all media events, keywords, and external datasets proposed in demand forecast explanations.

We retrieved 128226 media events related to the features of interest and manually annotated a random sample of 528 media events. Regarding (A), we found that 69% of retrieved media events were meaningful to features explaining a specific forecast. In addition, all media events were processed to search and retrieve information regarding external datasets, as described in Section 5. From the 401 external dataset entries we retrieved and annotated, we found that (B) 64% were relevant to our use case.

We manually annotated 168 forecast explanations collected over three months. Each forecast explanation listed three media events, three media events keywords, and an external dataset recommendation. We manually annotated all of them. Results of their evaluation are presented in Table 4.

We compiled relevant features to each forecast, associated media events, their annotated subset, and the annotated suggested datasets into a dataset made publicly available at [114].

When annotating the explanations, we found some interesting cases. One of them was the suggestion of the ozone pollution dataset to enrich the automotive demand forecasting model. After some research, we found that while ozone is not emitted directly by automobiles, it is formed in the atmosphere due to a complex set of chemical reactions involving hydrocarbons, oxides of nitrogen, and sunlight. Another interesting case was a news media entry describing wireless charging.

Table 4

Results we obtained from the analysis of forecast explanations issued for 56 products over three months. Media Events, Media Events' Keywords, and External Datasets correspond to contextual information displayed for each forecast explanation.

	Metric	Value
Media Events	average precision@1	0,97
	average precision@3	0,97
	RDE@1	0,30
	RDE@3	0,11
Media Events' Keywords	average precision@1	0,77
	average precision@3	0,78
	RDE@1	0,14
	RDE@3	0,09
External Datasets	accuracy	0,56
	RDE	0,41

This media event seemed unrelated to our use case, but we confirmed that electric vehicles' demand drives research and development of a segment of wireless charging solutions. Finally, the most frequently listed media event among our forecast explanations was "Number of people in work surges again to a new record high of 32.3m". We consider this event to relate to the economic context and be relevant to our demand forecasting case.

The results and examples presented above show that the current architecture and knowledge fusion procedures provide comprehensive explanations to the users while preserving confidentiality. Nevertheless, it must be noted that such an approach requires extracting relevant concepts and associating them to the specific features used to generate the forecasts, as described in Section 3.2.

7. Limitations and improvement opportunities

The architecture and knowledge fusion techniques presented in this paper assume that relevant concepts present in the features that are used to generate the forecasts are identified, mapped, and encoded into a knowledge graph. In addition, a set of keywords characterizing such concepts must be provided, as they are used to obtain additional information by querying datasets and media events. This approach requires that the features used to train the model have a particular meaning, a criterion usually satisfied by handcrafted features. Further

research is required to understand how such an approach can be tailored to models where features are not handcrafted, like in deep learning models. In particular, promising advances have been made regarding the extraction of deep learning models' features semantic meaning [115,116], the incorporation of semantic aspects to deep learning models [117–119], and the use of deep learning for automated ontology development and alignment [117]. Such advancements must be considered to enable the integration of deep learning models into the architecture presented in this work.

We have identified some improvement opportunities from the experience we obtained through the experiments and results described in this paper. While the method we have used to create local explanations and find the most meaningful features to a particular forecast LIME is widely adopted, it is not deterministic. Changes in the feature ranking computations can result in different forecast interpretations. Replacing LIME for some deterministic variant (e.g., DLIME), would provide a more consistent experience to the end-user and enhance scenarios reproducibility for researchers and engineers. Another possible improvement is to highlight current events related to the factors that are driving the forecast. Insights into past and current events would provide the user an enhanced understanding of the overall context, having insights into the past events that affect the models' prediction and relevant current events that can enhance judgment and derive better decision-making.

8. Conclusion

This research work presented an architecture that supports building semantically enhanced explanations for AI models' demand forecasts. We evaluated the architecture and explanations on a real-world use case developed with data obtained from European Horizon 2020 projects FACTLOG and STAR.

Current explanations provide demand forecast values, the associated uncertainty, high-level description of features that influenced the prediction, related media events, and a reference to some external dataset, providing context and means to improve the demand forecasting model. In addition, the high-level description of features informs the main factors influencing the forecast and avoids exposing sensitive details regarding the demand forecasting models.

Our future work will focus on further pre-processing media events, media events' keywords, and external datasets, to filter out those not relevant to the demand forecasting context and increase the forecast's explanation quality. At least three additional research directions can be pursued. First, explore means to increase the diversity of recommended media events and associated keywords displayed on forecast explanations. Second, evaluate the impact of incorporating suggested external datasets to the performance of existing demand forecasting models. Third, we envision enhancing the explanations by including meaningful current events reported by the media. This way, in addition to a good understanding of the past context, the explanations will provide information on events that are likely to influence future demand so that the user can gain a better judgment and perspective when making a decision. Finally, we would like to research the integration of semantic databases and federated ontologies, and how explanations can be enhanced with them.

CRedit authorship contribution statement

Jože M. Rožanec: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Resources, Data curation, Writing – original draft, Writing – review & editing, Visualization. **Blaž Fortuna:** Resources, Supervision, Project administration, Funding acquisition. **Dunja Mladenčić:** Resources, Writing – review & editing, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the Slovenian Research Agency, Slovenia and the European Union's Horizon 2020 program project FACT-LOG under grant agreement H2020-869951, and European Union's Horizon 2020 program project STAR under grant agreement number H2020-956573.

References

- [1] M. Gaur, K. Faldu, A. Sheth, Semantics of the Black-Box: Can knowledge graphs help make deep learning systems more interpretable and explainable? 2020, arXiv preprint [arXiv:2010.08660](https://arxiv.org/abs/2010.08660).
- [2] C. Panigutti, A. Perotti, D. Pedreschi, Doctor XAI: an ontology-based approach to black-box sequential data classification explanations, in: Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency, 2020, pp. 629–639.
- [3] J.M. Rožanec, B. Kažič, M. Škrjanc, B. Fortuna, D. Mladenčić, Automotive OEM demand forecasting: A comparative study of forecasting algorithms and strategies, *Appl. Sci.* 11 (2021) 6787.
- [4] F. Lolli, R. Gamberini, A. Regattieri, E. Balugani, T. Gatos, S. Gucci, Single-hidden layer neural networks for forecasting intermittent demand, *Int. J. Prod. Econ.* 183 (2017) 116–128.
- [5] J.J. Bergman, J.S. Noble, R.G. McGarvey, R.L. Bradley, A Bayesian approach to demand forecasting for new equipment programs, *Robot. Comput.-Integr. Manuf.* 47 (2017) 17–21.
- [6] M. Babai, A. Tsadiras, C. Papadopoulos, On the empirical performance of some new neural network methods for forecasting intermittent demand, *IMA J. Manag. Math.* (2020).
- [7] T. Benbarrad, M. Salhaoui, S.B. Kenitar, M. Arioua, Intelligent machine vision model for defective product inspection based on machine learning, *J. Sensor Actuator Netw.* 10 (2021) 7.
- [8] R. Rajkumar, I. Lee, L. Sha, J. Stankovic, Cyber-physical systems: the next computing revolution, in: Design Automation Conference, IEEE, 2010, pp. 731–736.
- [9] M.W. Grieves, Virtually intelligent product systems: digital and physical twins, 2019.
- [10] P. Tubaro, A.A. Casilli, Micro-work, artificial intelligence and the automotive industry, *J. Ind. Bus. Econ.* 46 (2019) 333–345.
- [11] R.D. Raut, A. Gotmare, B.E. Narkhede, U.H. Govindarajan, S.U. Bokade, Enabling technologies for Industry 4.0 manufacturing and supply chain: concepts, current status, and adoption challenges, *IEEE Eng. Manag. Rev.* 48 (2020) 83–102.
- [12] F. Tao, Q. Qi, A. Liu, A. Kusiak, Data-driven smart manufacturing, *J. Manuf. Syst.* 48 (2018) 157–169.
- [13] B. Brühl, M. Hülsmann, D. Borscheid, C.M. Friedrich, D. Reith, A sales forecast model for the german automobile market based on time series analysis and data mining methods, in: Industrial Conference on Data Mining, Springer, 2009, pp. 146–160.
- [14] F.-K. Wang, K.-K. Chang, C.-W. Tzeng, Using adaptive network-based fuzzy inference system to forecast automobile sales, *Expert Syst. Appl.* 38 (2011) 10587–10593.
- [15] A. Vahabi, S.S. Hosseini, M. Alborzi, A sales forecasting model in automotive industry using adaptive neuro-fuzzy inference system (anfis) and genetic algorithm (GA), *Management* 1 (2016) 2.
- [16] J. Gao, Y. Xie, X. Cui, H. Yu, F. Gu, Chinese automobile sales forecasting using economic indicators and typical domestic brand automobile sales data: A method based on econometric model, *Adv. Mech. Eng.* 10 (2018) 1687814017749325.
- [17] N.Z. Ubaidillah, A study of car demand and its interdependency in sarawak, *Int. J. Bus. Soc.* 21 (2020).
- [18] T. Williams, Stock control with sporadic and slow-moving demand, *J. Oper. Res. Soc.* 35 (1984) 939–948.
- [19] F. Johnston, J.E. Boylan, Forecasting for items with intermittent demand, *J. Oper. Res. Soc.* 47 (1996) 113–121.
- [20] A.A. Syntetos, J.E. Boylan, J. Croston, On the categorization of demand patterns, *J. Oper. Res. Soc.* 56 (2005) 495–503.
- [21] A.A. Syntetos, M.Z. Babai, N. Altay, On the demand distributions of spare parts, *Int. J. Prod. Res.* 50 (2012) 2101–2117.

- [22] D. Lengu, A.A. Syntetos, M.Z. Babai, Spare parts management: Linking distributional assumptions to demand classification, *European J. Oper. Res.* 235 (2014) 624–635.
- [23] R. Saluja, A. Malhi, S. Knapic, K. Främling, C. Cavdar, Towards a rigorous evaluation of explainability for multivariate time series, 2021, arXiv preprint [arXiv:2104.04075](https://arxiv.org/abs/2104.04075).
- [24] A. Dwivedi, M. Niranjana, K. Sahu, A business intelligence technique for forecasting the automobile sales using adaptive intelligent systems (ANFIS and ANN), *Int. J. Comput. Appl.* 74 (2013).
- [25] X. Wang, D. Zeng, H. Dai, Y. Zhu, Making the right business decision: Forecasting the binary NPD strategy in Chinese automotive industry with machine learning methods, *Technol. Forecast. Soc. Change* 155 (2020) 120032.
- [26] D.S. Farahani, M. Momeni, N.S. Amiri, Car sales forecasting using artificial neural networks and analytical hierarchy process, *DATA ANALYTICS* 2016 (2016) 69.
- [27] K.K. Chandriah, R.V. Naraganahalli, RNN/LSTM with modified Adam optimizer in deep learning approach for automobile spare parts demand forecasting, *Multimedia Tools Appl.* (2021) 1–15.
- [28] R. Fildes, P. Goodwin, Stability in the inefficient use of forecasting systems: A case study in a supply chain company, *Int. J. Forecast.* 37 (2021) 1031–1046.
- [29] P. Corredor, E. Ferrer, R. Santamaria, Is cognitive bias really present in analyst forecasts? The role of investor sentiment, *Int. Bus. Rev.* 23 (2014) 824–837.
- [30] R.M. Hogarth, S. Makridakis, Forecasting and planning: An evaluation, *Manage. Sci.* 27 (1981) 115–138.
- [31] J.H. Barnes Jr., Cognitive biases and their impact on strategic planning, *Strat. Manag. J.* 5 (1984) 129–137.
- [32] A.B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, et al., Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI, *Inf. Fusion* 58 (2020) 82–115.
- [33] O. Biran, C. Cotton, Explanation and justification in machine learning: A survey, in: *IJCAI-17 Workshop on Explainable AI (XAI)*, vol. 8, 2017, pp. 8–13.
- [34] A. Malhi, S. Knapic, K. Främling, Explainable agents for less bias in human-agent decision making, in: *International Workshop on Explainable, Transparent Autonomous Agents and Multi-Agent Systems*, Springer, 2020, pp. 129–146.
- [35] X. Wang, M. Yin, Are explanations helpful? A comparative study of the effects of explanations in AI-assisted decision-making, in: *26th International Conference on Intelligent User Interfaces*, 2021, pp. 318–328.
- [36] J.J. Ferreira, M. Monteiro, The human-AI relationship in decision-making: AI explanation to support people on justifying their decisions, 2021, arXiv preprint [arXiv:2102.05460](https://arxiv.org/abs/2102.05460).
- [37] O. Biran, K.R. McKeown, Human-centric justification of machine learning predictions, in: *IJCAI*, vol. 2017, 2017, pp. 1461–1467.
- [38] A. Adadi, M. Berrada, Peeking inside the black-box: a survey on explainable artificial intelligence (XAI), *IEEE Access* 6 (2018) 52138–52160.
- [39] O. Loyola-Gonzalez, Black-box vs. white-box: Understanding their advantages and weaknesses from a practical point of view, *IEEE Access* 7 (2019) 154096–154113.
- [40] A. Das, P. Rad, Opportunities and challenges in explainable artificial intelligence (xai): A survey, 2020, arXiv preprint [arXiv:2006.11371](https://arxiv.org/abs/2006.11371).
- [41] Q.V. Liao, D. Gruen, S. Miller, Questioning the AI: informing design practices for explainable AI user experiences, in: *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, 2020, pp. 1–15.
- [42] P.P. Angelov, E.A. Soares, R. Jiang, N.I. Arnold, P.M. Atkinson, Explainable artificial intelligence: an analytical review, *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 11 (2021) e1424.
- [43] T. Rojat, R. Puget, D. Filliat, J. Del Ser, R. Gelin, N. Díaz-Rodríguez, Explainable artificial intelligence (XAI) on TimeSeries data: A survey, 2021, arXiv preprint [arXiv:2104.00950](https://arxiv.org/abs/2104.00950).
- [44] S.A. Siddiqui, D. Mercier, M. Munir, A. Dengel, S. Ahmed, Tsviz: Demystification of deep learning models for time-series analysis, *IEEE Access* 7 (2019) 67027–67040.
- [45] K. Kashiparekh, J. Narwariya, P. Malhotra, L. Vig, G. Shroff, ConvTimeNet: A pre-trained deep convolutional neural network for time series classification, in: *2019 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2019, pp. 1–8.
- [46] S. Tonekaboni, S. Joshi, D. Duvenaud, A. Goldenberg, Explaining time series by counterfactuals, 2019.
- [47] O. Ozyegen, I. Ilic, M. Cevik, Evaluation of local explanation methods for multivariate time series forecasting, 2020, arXiv preprint [arXiv:2009.09092](https://arxiv.org/abs/2009.09092).
- [48] D. Mercier, A. Dengel, S. Ahmed, P2exnet: Patch-based prototype explanation network, in: *International Conference on Neural Information Processing*, Springer, 2020, pp. 318–330.
- [49] B. Lim, S.O. Arık, N. Loeff, T. Pfister, Temporal fusion transformers for interpretable multi-horizon time series forecasting, *Int. J. Forecast.* (2021).
- [50] M.T. Ribeiro, S. Singh, C. Guestrin, “Why should I trust you?” Explaining the predictions of any classifier, in: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 1135–1144.
- [51] P. Hall, N. Gill, M. Kurka, W. Phan, Machine learning interpretability with h2o driverless ai, 2017, H2O. Ai. URL: <http://docs.h2o.ai/driverless-ai/latest-stable/docs/booklets/MLIBooklet.pdf>.
- [52] M.R. Zafar, N.M. Khan, DLIME: A deterministic local interpretable model-agnostic explanations approach for computer-aided diagnosis systems, 2019, arXiv preprint [arXiv:1906.10263](https://arxiv.org/abs/1906.10263).
- [53] K. Sokol, P. Flach, LIMETree: Interactively customisable explanations based on local surrogate multi-output regression trees, 2020, arXiv preprint [arXiv:2005.01427](https://arxiv.org/abs/2005.01427).
- [54] M.T. Ribeiro, S. Singh, C. Guestrin, Anchors: High-precision model-agnostic explanations, in: *AAAI*, 18, 2018, pp. 1527–1535.
- [55] J. van der Waa, M. Robeer, J. van Diggelen, M. Brinkhuis, M. Neerinx, Contrastive explanations with local foil trees, 2018, arXiv preprint [arXiv:1806.07470](https://arxiv.org/abs/1806.07470).
- [56] R. Guidotti, A. Monreale, S. Ruggieri, D. Pedreschi, F. Turini, F. Giannotti, Local rule-based explanations of black box decision systems, 2018, arXiv preprint [arXiv:1805.10820](https://arxiv.org/abs/1805.10820).
- [57] J. Rožanec, E. Trajkova, K. Kenda, B. Fortuna, D. Mladenčić, Explaining bad forecasts in global time series models, 2021.
- [58] W. Samek, K.-R. Müller, Towards explainable artificial intelligence, in: *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, Springer, 2019, pp. 5–22.
- [59] R. Srinivasan, A. Chander, Explanation perspectives from the cognitive sciences-a survey, in: *IJCAI*, 2020, pp. 4812–4818.
- [60] D. Pedreschi, F. Giannotti, R. Guidotti, A. Monreale, L. Pappalardo, S. Ruggieri, F. Turini, Open the black box data-driven explanation of black box decision systems, 2018, arXiv preprint [arXiv:1806.09936](https://arxiv.org/abs/1806.09936).
- [61] S. Verma, J. Dickerson, K. Hines, Counterfactual explanations for machine learning: A review, 2020, arXiv preprint [arXiv:2010.10596](https://arxiv.org/abs/2010.10596).
- [62] D. Doran, S. Schulz, T.R. Besold, What does explainable AI really mean? A new conceptualization of perspectives, 2017, arXiv preprint [arXiv:1710.00794](https://arxiv.org/abs/1710.00794).
- [63] M. Chromik, Human-Centric Explanation Facilities, (Ph.D. thesis), lmu, 2021.
- [64] M. El-Assady, W. Jentner, R. Kehlbeck, U. Schlegel, R. Sevastjanova, F. Sperrle, T. Spinner, D. Keim, Towards XAI: Structuring the processes of explanations, in: *ACM Workshop on Human-Centered Machine Learning*, 2019.
- [65] M.T. Keane, B. Smyth, Good counterfactuals and where to find them: A case-based technique for generating counterfactuals for explainable AI (XAI), in: *International Conference on Case-Based Reasoning*, Springer, 2020, pp. 163–178.
- [66] M.T. Keane, E.M. Kenny, E. Delaney, B. Smyth, If only we had better counterfactual explanations: five key deficits to rectify in the evaluation of counterfactual XAI techniques, 2021, arXiv preprint [arXiv:2103.01035](https://arxiv.org/abs/2103.01035).
- [67] F. Ameri, D. Dutta, An upper ontology for manufacturing service description, in: *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, vol. 42578, 2006, pp. 651–661.
- [68] S. Lemaignan, A. Siadat, J.-Y. Dantan, A. Semenenko, MASON: A proposal for an ontology of manufacturing domain, in: *IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications (DIS'06)*, IEEE, 2006, pp. 195–200.
- [69] M. Courtot, F. Gibson, A.L. Lister, J. Malone, D. Schober, R.R. Brinkman, A. Ruttenberg, MIREOT: The minimum information to reference an external ontology term, *Appl. Ontol.* 6 (2011) 23–33.
- [70] M. Cannataro, C. Comito, A data mining ontology for grid programming, in: *Proc. 1st Int. Workshop on Semantics in Peer-To-Peer and Grid Computing*, Citeseer, 2003, pp. 113–134.
- [71] P. Panov, S. Džeroski, L. Soldatova, OntoDM: An ontology of data mining, in: *2008 IEEE International Conference on Data Mining Workshops*, IEEE, 2008, pp. 752–760.
- [72] P. Panov, L. Soldatova, S. Džeroski, Ontology of core data mining entities, *Data Min. Knowl. Discov.* 28 (2014) 1222–1265.
- [73] C. Diamantini, D. Potena, E. Storti, Kddonto: An ontology for discovery and composition of kdd algorithms, in: *Third Generation Data Mining: Towards Service-Oriented Knowledge Discovery (SoKD'09)*, Citeseer, 2009, pp. 13–24.
- [74] V. Ermolayev, N. Keberle, W.-E. Matzke, An ontology of environments, events, and happenings, in: *2008 32nd Annual IEEE International Computer Software and Applications Conference*, IEEE, 2008, pp. 539–546.
- [75] S. Gottschalk, E. Demidova, Eventkg: A multilingual event-centric temporal knowledge graph, in: *European Semantic Web Conference*, Springer, 2018, pp. 272–287.
- [76] R. Mizoguchi, YAMATO: yet another more advanced top-level ontology, in: *Proceedings of the Sixth Australasian Ontology Workshop*, 2010, pp. 1–16.
- [77] M. Uschold, Building ontologies: Towards a unified methodology, in: *Proceedings of 16th Annual Conference of the British Computer Society Specialists Group on Expert Systems*, Citeseer, 1996.
- [78] M. Uschold, M. King, Towards a Methodology for Building Ontologies, Citeseer, 1995.
- [79] M. Fernández-López, A. Gómez-Pérez, N. Juristo, Methontology: from ontological art towards ontological engineering, 1997.

- [80] H.M. Kim, M.S. Fox, M. Gruninger, An ontology of quality for enterprise modelling, in: *Proceedings 4th IEEE Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises (WET ICE'95)*, IEEE, 1995, pp. 105–116.
- [81] F. Ameri, C. Urbanovsky, C. McArthur, A systematic approach to developing ontologies for manufacturing service modeling, in: *Proceedings of the Workshop on Ontology and Semantic Web for Manufacturing*, vol. 14, 2012.
- [82] X. Chang, R. Rai, J. Terpeny, Development and utilization of ontologies in design for manufacturing, *J. Mech. Des.* 132 (2010).
- [83] R. Weber, M. Shrestha, A.J. Johs, Knowledge-based XAI through CBR: There is more to explanations than models can tell, 2021, arXiv preprint [arXiv: 2108.10363](https://arxiv.org/abs/2108.10363).
- [84] S.R. Islama, W. Eberleb, Implications of combining domain knowledge in explainable artificial intelligence, 2021.
- [85] R. Sharma, A.K. Sinha, Sales forecast of an automobile industry, *Int. J. Comput. Appl.* 53 (2012).
- [86] D. Salinas, V. Flunkert, J. Gasthaus, T. Januschowski, DeepAR: Probabilistic forecasting with autoregressive recurrent networks, *Int. J. Forecast.* 36 (2020) 1181–1191.
- [87] M. Nentwig, M. Hartung, A.-C. Ngonga Ngomo, E. Rahm, A survey of current link discovery frameworks, *Semantic Web* 8 (2017) 419–436.
- [88] M.A. Khoudja, M. Fareh, H. Bouarfa, Ontology matching using neural networks: survey and analysis, in: *2018 International Conference on Applied Smart Systems (ICASS)*, IEEE, 2018, pp. 1–6.
- [89] J.P. McCrae, P. Buitelaar, Linking datasets using semantic textual similarity, *Cybernet. Inform. Technol.* 18 (2018) 109–123.
- [90] E. Thiéblin, O. Haemmerlé, N. Hernandez, C. Trojahn, Survey on complex ontology matching, *Semantic Web* 11 (2020) 689–727.
- [91] L.-W. Chen, Extending wikification: Nominal discovery, nominal linking, and the grounding of nouns, 2017.
- [92] J. Brank, G. Leban, M. Grobelnik, Annotating documents with relevant wikipedia concepts, in: *Proceedings of SIKDD*.
- [93] R.C. Fernandez, E. Mansour, A.A. Qahtan, A. Elmagarmid, I. Ilyas, S. Madden, M. Ouzzani, M. Stonebraker, N. Tang, Seeping semantics: Linking datasets using word embeddings for data discovery, in: *2018 IEEE 34th International Conference on Data Engineering (ICDE)*, IEEE, 2018, pp. 989–1000.
- [94] R.J. Miller, Open data integration, *Proc. VLDB Endow.* 11 (2018) 2130–2139.
- [95] X.L. Dong, T. Rekatsinas, Data integration and machine learning: A natural synergy, in: *Proceedings of the 2018 International Conference on Management of Data*, 2018, pp. 1645–1650.
- [96] E. Demidova, A. Zaveri, E. Simperl, Matching domain and top-level ontologies exploring word sense disambiguation and word embedding, in: *Emerging Topics in Semantic Technologies: ISWC 2018 Satellite Events*, vol. 36, IOS Press, 2018, p. 27.
- [97] A. Assi, H. Mcheick, A. Karawash, W. Dhifli, Context-aware instance matching through graph embedding in lexical semantic space, *Knowl.-Based Syst.* 186 (2019) 104925.
- [98] J.z.M. Rožanec, D. Mladenčić, Ontology for contextualized demand forecasting explanations, 2020, <http://dx.doi.org/10.7910/DVN/GIREAY>.
- [99] P. Leitão, F. Restivo, ADACOR: A holonic architecture for agile and adaptive manufacturing control, *Comput. Ind.* 57 (2006) 121–130.
- [100] S. Borgo, P. Leitão, Foundations for a core ontology of manufacturing, in: *Ontologies*, Springer, 2007, pp. 751–775.
- [101] G. Kourtis, E. Kavakli, R. Sakellariou, A rule-based approach founded on description logics for Industry 4.0 smart factories, *IEEE Trans. Ind. Inf.* 15 (2019) 4888–4899.
- [102] J.S. Armstrong, V.G. Morwitz, V. Kumar, Sales forecasts for existing consumer products and services: Do purchase intentions contribute to accuracy? *Int. J. Forecast.* 16 (2000) 383–397.
- [103] V.G. Morwitz, J.H. Steckel, A. Gupta, When do purchase intentions predict sales? *Int. J. Forecast.* 23 (2007) 347–364.
- [104] H. Drucker, C.J. Burges, L. Kaufman, A.J. Smola, V. Vapnik, Support vector regression machines, in: *Advances in Neural Information Processing Systems*, 1997, pp. 155–161.
- [105] M. Stone, Cross-validated choice and assessment of statistical predictions, *J. R. Stat. Soc. Ser. B Stat. Methodol.* 36 (1974) 111–133.
- [106] R.J. Hyndman, et al., Another look at forecast-accuracy metrics for intermittent demand, *Int. J. Appl. Forecast.* 4 (2006) 43–46.
- [107] G. Leban, B. Fortuna, J. Brank, M. Grobelnik, Event registry: learning about world events from news, in: *Proceedings of the 23rd International Conference on World Wide Web*, 2014, pp. 107–110.
- [108] P.O. of the European Union, EU open data portal: The official portal for European data, 2020, <https://data.europa.eu> (Accessed 15 December 2020).
- [109] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: *Advances in Neural Information Processing Systems*, 2013, pp. 3111–3119.
- [110] Google, Word2Vec: an efficient implementation of the continuous bag-of-words and skip-gram architectures for computing vector representations of words, 2015, <https://code.google.com/archive/p/word2vec> (Accessed 15 December 2020).
- [111] M. Kusner, Y. Sun, N. Kolkin, K. Weinberger, From word embeddings to document distances, in: *International Conference on Machine Learning*, 2015, pp. 957–966.
- [112] N.F. Noy, M. Crubézy, R.W. Fergerson, H. Knublauch, S.W. Tu, J. Vendetti, M.A. Musen, Protégé-2000: an open-source ontology-development and knowledge-acquisition environment, in: *AMIA... Annual Symposium Proceedings. AMIA Symposium*, 2003, p. 953.
- [113] Neo4j, Neo4j: The graph database platform for today's intelligent applications, 2020, <https://neo4j.com> (Accessed 15 December 2020).
- [114] J.z.M. Rožanec, D. Mladenčić, Data for contextualized demand forecasting explanations, 2020, <http://dx.doi.org/10.7910/DVN/FPXYCM>.
- [115] A. Gudi, Recognizing semantic features in faces using deep learning, 2015, arXiv preprint [arXiv:1512.00743](https://arxiv.org/abs/1512.00743).
- [116] F.D. de Souza, S. Sarkar, G. Cámara-Chávez, Building semantic understanding beyond deep learning from sound and vision, in: *2016 23rd International Conference on Pattern Recognition (ICPR)*, IEEE, 2016, pp. 2097–2102.
- [117] X. Huang, C. Zanni-Merk, B. Crémilleux, Enhancing deep learning with semantics: an application to manufacturing time series analysis, *Procedia Comput. Sci.* 159 (2019) 437–446.
- [118] E. Coviello, A.B. Chan, G. Lanckriet, Time series models for semantic music annotation, *IEEE Trans. Audio Speech Lang. Process.* 19 (2010) 1343–1359.
- [119] Y. Dong, H. Su, J. Zhu, B. Zhang, Improving interpretability of deep neural networks with semantic information, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4306–4314.



Article

Enriching Artificial Intelligence Explanations with Knowledge Fragments

Jože Rožanec ^{1,2,3,*} , Elena Trajkova ⁴ , Inna Novalija ² , Patrik Zajec ^{1,2} , Klemen Kenda ^{1,2,3} ,
Blaž Fortuna ³ and Dunja Mladenec ²

- ¹ Jožef Stefan International Postgraduate School, Jamova 39, 1000 Ljubljana, Slovenia; patrik.zajec@ijs.si (P.Z.); klemen.kenda@ijs.si (K.K.)
² Jožef Stefan Institute, Jamova 39, 1000 Ljubljana, Slovenia; inna.koval@ijs.si (I.N.); dunja.mladenec@ijs.si (D.M.)
³ Qlector d.o.o., Rovšnikova 7, 1000 Ljubljana, Slovenia; blaz.fortuna@qlector.com
⁴ Faculty of Electrical Engineering, University of Ljubljana, Tržaška c. 25, 1000 Ljubljana, Slovenia; trajkova.elena.00@gmail.com
* Correspondence: joze.rozanec@ijs.si

Abstract: Artificial intelligence models are increasingly used in manufacturing to inform decision making. Responsible decision making requires accurate forecasts and an understanding of the models' behavior. Furthermore, the insights into the models' rationale can be enriched with domain knowledge. This research builds explanations considering feature rankings for a particular forecast, enriching them with media news entries, datasets' metadata, and entries from the Google knowledge graph. We compare two approaches (embeddings-based and semantic-based) on a real-world use case regarding demand forecasting. The embeddings-based approach measures the similarity between relevant concepts and retrieved media news entries and datasets' metadata based on the word movers' distance between embeddings. The semantic-based approach recourses to wikification and measures the Jaccard distance instead. The semantic-based approach leads to more diverse entries when displaying media events and more precise and diverse results regarding recommended datasets. We conclude that the explanations provided can be further improved with information regarding the purpose of potential actions that can be taken to influence demand and to provide "what-if" analysis capabilities.

Keywords: explainable artificial intelligence; human-centric artificial intelligence; smart manufacturing; demand forecasting; Industry 4.0; Industry 5.0



Citation: Rožanec, J.; Trajkova, E.; Novalija, I.; Zajec, P.; Kenda, K.; Fortuna, B.; Mladenec, D. Enriching Artificial Intelligence Explanations with Knowledge Fragments. *Future Internet* **2022**, *14*, 134. <https://doi.org/10.3390/fi14050134>

Academic Editors: Massimo Stella and Giulio Rossetti

Received: 7 April 2022

Accepted: 26 April 2022

Published: 29 April 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The industry is the part of the economy that is concerned with a highly mechanized and automated production of material goods [1]. Since the beginning of industrialization, technological breakthroughs have allowed to achieve increasing manufacturing efficiencies and have led to paradigm shifts. The latest such shifts are known as Industry 4.0 and Industry 5.0 [2,3], and aim to foster and increase the digitalization, networking, and automation of the manufacturing processes. Furthermore, these, in turn, enable tighter integration between physical and cyber domains (e.g., through cyber-physical systems [4], and digital twins [5]), increased deployment of intelligence (e.g., through artificial intelligence solutions [6]), to achieve autonomy and automation. These changes are expected to shorten the development and manufacturing periods, increase the manufacturing efficiency and sustainability [7], and achieve greater flexibility. Furthermore, emphasis on making such technologies and applications safe, trustworthy, and human-centric is a crucial characteristic of the nascent Industry 5.0 paradigm [8,9].

The increasing digitalization and the democratization of artificial intelligence have enabled such models to automate and provide assistance on a wide range of tasks. Machine

learning models can learn from historic data to predict future outcomes (e.g., estimate future demand [10]), or perform certain tasks (e.g., identify manufacturing defects [11] or automate manual tasks with robotic assistance [12]). Nevertheless, it is crucial to establish a synergic and collaborative environment between humans and machines, where humans can flourish in developing their creativity, while delegating monotonous and repetitive tasks to machines [9]. Furthermore, such collaboration requires transparency to develop trust in artificial intelligence applications. Research exploring the models' rationale behind the predictions and how the insights must be provided to the user as an explanation is known as explainable artificial intelligence.

Insights obtained regarding the models' rationale behind a prediction can be enriched with domain knowledge to provide a better-contextualized explanation to the user [13,14]. To encode domain knowledge, researchers have resorted to using semantic technologies, such as ontologies and knowledge graphs. In order to integrate semantic technologies into the explanation crafting process, the structure of the explanation must be defined, and a procedure established to (a) extract semantic meaning from the explanation, (b) use such information to query external sources (e.g., open knowledge graphs), and (c) enrich the explanation with new information and insights obtained from the external sources.

In this research, we extend the work performed by [15,16]. Given a set of demand forecasts, a mapping between features and concepts that define an ontology and a hierarchy of concepts, and a set of keywords associated with each concept, we recourse to a wikification process to extract wiki concepts based on the keywords mentioned above. The wiki concepts are then used to query external sources (e.g., open knowledge graphs or open datasets) and rank query results based on their semantic similarity, computing the Jaccard distance. Our approach was tested in the domain of demand forecasting and validated on a real-world case study, using models we developed as part of the European Horizon 2020 projects FACTLOG (<https://www.factlog.eu/>, (accessed on 27 April 2022)) and STAR (<http://www.star-ai.eu/>, (accessed on 27 April 2022)). We evaluate our results through three metrics (precision@K and Ratio of Diverse Entries (RDE@K), and coverage regarding five questions proposed in [17]) to assess how precise and diverse the knowledge fragments are.

The rest of this paper is structured as follows: Section 2 presents related work, Section 3 describes the use case we used and the implementation we followed to test our concept, and Section 4 provides the results we obtained and their evaluation. Finally, in Section 5, we provide our conclusions and outline future work.

2. Related Work

2.1. Industry 4.0 and Industry 5.0

Industry 4.0 was coined in 2011 by the German government initiative aimed at developing advanced production systems to increase the productivity and efficiency of the manufacturing industry. The initiative was soon followed by national initiatives across multiple countries, such as the USA, UK, France, Canada, Japan, and China, among others [18–20]. Industry 4.0 was conceived as a technological revolution adding value to the whole manufacturing and product lifecycle. Part of such a revolution is the concept of a smart and integrated supply chain, which aims to reduce delivery times and reduce information distortion across suppliers and manufacturers [21]. The benefits mentioned above are achieved by enhancing the demand forecasting and optimizing the organization and management of materials, suppliers, and inventory [22,23]. To that end, digital twins of existing processes can be created to simulate, test *what-if* scenarios, and enhance them without disruption of the physical operations [24,25].

On top of the aforementioned integrated supply chain, the Industry 4.0 paradigm emphasizes the redesign of the human role in manufacturing, leveraging new technological advancements and capabilities [21]. This aspect is further evolved in Industry 5.0. Industry 5.0 is a value-driven manufacturing paradigm that underscores the relevance of research and innovation to develop human-centric approaches when supporting industry

operation [26]. Human-centricity in manufacturing must take into account the skills that are unique to the human workers, such as critical thinking, creativity, and domain knowledge, while leveraging machine strengths (e.g., high efficiency in performing repetitive tasks [3,27,28]). Human-centricity can be realized through (a) a systemic approach focused on forging synergic, two-way relationships between humans and machines, (b) the use of digital twins at a systemic level, and (c) the adoption of artificial intelligence at all levels [9,29].

Artificial intelligence can be essential to achieving human–machine collaboration. In particular, active learning and explainable artificial intelligence can be used to complement each other. Active learning is the sub-field of artificial intelligence, concerned with retrieving specific data and leveraging human knowledge to satisfy a particular learning goal. On the other hand, explainable artificial intelligence is the sub-field of artificial intelligence concerned with providing insights into the inner workings of a model regarding its outcome so that the user can learn about its underlying behavior. This way, the human can act as a teacher to artificial intelligence models and learn from them through explainable artificial intelligence. This two-way relationship can lead to a trusted collaboration [30,31].

2.2. Demand Forecasting

Demand forecasting is a key component of manufacturing companies. Precise demand forecasts allow to set correct inventory levels, price the products and plan future operations. Any improvements in such forecasts directly translate to the supply chain performance [32]. Demand depends on characteristics that are intrinsic to the product [33] (e.g., elasticity or configuration), and external factors [34] (e.g., particular sales conditions).

Researchers have developed multiple schemas to classify demand according to its characteristics. Among them, we find the ABC inventory classification system [35], the XYZ analysis [36], and the quadrant proposed by Syntetos et al. [37]. The ABC inventory classification system classifies the items based on their decreasing order of annual dollar volume. The XYZ analysis classifies items according to their consumption patterns: (X) constant consumption, (Y) fluctuating (usually seasonal) consumption, and (Z) irregular consumption. Finally, Syntetos et al. divide demand into four quadrants based on demand size and demand occurrence variability. It was recognized that artificial intelligence models can provide accurate demand forecasts based on past demand and complementary data. Different methods are appropriate for demands with different characteristics [38]. Demand forecasting models are usually framed as time series forecasting problems using supervised regression models or specialized models to learn patterns and forecast future values. Research related to the automotive industry has reported using complementary sources, such as the unemployment rate [39], inflation rate [40] or gross domestic product [41], and a variety of algorithms, such as multiple linear regression [42], support vector machine [43], and neural networks [44].

The ability to leverage a wide range of complementary data sources is a specific advantage of such models against humans in light of their visual working memory, short-term memory, and capacity to process variables [45]. Furthermore, artificial intelligence models avoid multiple cognitive biases to which humans are prone to [46]. Nevertheless, planners must approach such forecasts with critical thinking since they hold responsibility for decisions based on such forecasts. They must understand the models' rationale behind the forecast [47–49], take into account information that could signal that adjustments to the forecast are needed and make such adjustments when needed [50–52]. Furthermore, while transparency regarding the models' underlying rationale can be, in some cases, required by law [53] (e.g., the general data protection regulation (GDPR) [54] or the artificial intelligence act [55]), it also provides a learning opportunity, which is key to the employees' engagement [56].

2.3. Explainable Artificial Intelligence

Explainable artificial intelligence is a sub-field of artificial intelligence research, concerned with how the models' behavioral aspects can be translated into a human interpretable form to understand causality, enhance trustworthiness, and develop confidence [57,58]. Techniques are usually classified based on their complexity (degree of interpretability), their scope (global or local), and whether they are model-agnostic [47]. Regarding their degree of interpretability, models are usually considered black-box (opaque) or white-box (transparent). The source of models' opacity can be due to the inherent properties of the algorithms, their complexity, or due to the explicit requirement to avoid exposing its inner workings (e.g., trade secret), [59,60].

Demand forecasting can be framed as a regression problem, and thus explainability techniques developed for this kind of supervised learning can be used to unveil the models' inner workings. One such technique is the *Local Interpretable Model-agnostic Explanations* (LIME) [61], which provide a model-agnostic approach to estimate features' relevance for each forecast. LIME creates a linear model to approximate the model's behavior at a particular forecasting point. Then, it estimates the features' relevance by measuring how much predictions change upon features' perturbation. Similar approaches were later developed to ensure that explanations were deterministic [62], or take into account non-linear relationships between the features [63,64]. Other frequently cited methods are the *Shapley additive explanations* (SHAP, which leverages coalitional game theory to estimate the contributions of individual feature values) [65,66], *Local Agnostic attribute Contribution Explanation* (LACE, which leverages SHAP and LIME to provide an explanation through local rules) [67], *Local Rule-based Explanation* (LoRE, which crafts an explanation extracting a decision rule and a set of counterfactual rules) [68], *anchors* [69] (which establishes a set of precise rules that explain the forecast at a local level), and *local foil trees* [70] (which identifies a disjoint set of rules that result in counterfactual outcomes).

Understanding why the model issued a particular forecast is of utmost importance [71]. Explanation crafting can be guided using domain knowledge, enriched with complementary data or insights. An approach to enhance explanations given specific domain knowledge was developed by Confalonieri et al. [72], who demonstrated that decision trees built for explainability were more understandable if built considering domain knowledge. Enrichment with complementary data was researched by Panigutti et al. [73], where high fidelity to the forecasting model was achieved by enriching the explainability rules with semantically encoded domain knowledge. Semantic enrichment or recourse to graph representations to bind multiple insights was explored by several authors [14,15,74]. Finally, Rabold et al. [75] devised means to create enhanced, multimodal explanations by relating visual explanations to logic rules obtained through an inductive logic programming system.

While many metrics have been devised to assess the quality of the outcomes produced by explainability techniques [76–79], explanations must also be evaluated and measured from a human-centric perspective. Attention must be devoted to ensuring the explanations convey meaningful information to different user profiles according to their purpose [80–82] such that they promote curiosity to increase learning and engagement [83], and provide means to develop trust through exploration [84]. Furthermore, it is desired that the explanations are actionable [85,86], and inform conditions that could change the forecast outcome [87]. The aspects mentioned above are frequently evaluated through qualitative interviews and questionnaires, think-aloud, or self-explanations [83,88]. Methods such as tracking participants' eye movement patterns, measuring users' response time to an explanation, the quantification of answers provided, or the number of explanations required by the user to learn have also been proposed [17,83,88,89].

3. Use Case

This research was developed for a demand forecasting use case, exploring how to provide comprehensive forecast explanations. In particular, we consider the most relevant features regarding a particular forecast and enrich it based on domain knowledge and

information retrieved from external sources. The explanations aim to (a) provide context regarding real-world events that could describe feature values observed for a particular forecast, (b) assist in exploring datasets that can be used to enrich the dataset used to train the model, and (c) discover new relevant concepts by integrating with knowledge graphs. Knowledge regarding which factors drive a particular forecast and real-world events provides the user with valuable context for responsible decision making. Exploring complementary datasets is valuable to data scientists and machine learning engineers, who can iterate upon the existing model and enhance it. Furthermore, such information can also be valuable to other profiles, widening the understanding of forces driving the forecast or the influence of such particular demand. Such understanding can also be widened with insights obtained from knowledge graphs.

This research extends work performed in [15,16,90], taking into account four data sources: (i) data provided by a European original equipment manufacturer targeting the global automotive industry market; (ii) news media entries provided by Event Registry [91], a well-established media events retrieval system that provides real-time mainstream media monitoring; (iii) the EU Open Data Portal [92]; and (iv) the Google Knowledge Graph (Google KG) [93]. We detail the procedure in Figure 1.

$$JaccardIndex(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (1)$$

$$JaccardDistance = 1 - JaccardIndex \quad (2)$$

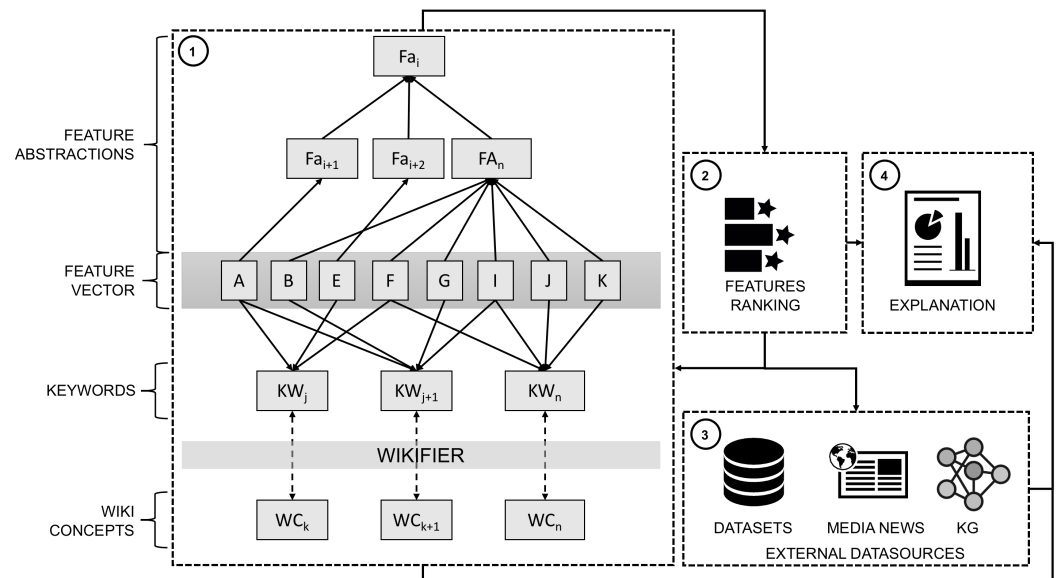


Figure 1. High-level diagram of the components taken into account and the procedure followed to craft the explanations. (1) Given a feature vector, feature abstractions are created, to encode their semantic mining at higher levels. At the same time, keywords are associated to them, whose wiki-concept correspondence is obtained using a wikifier. (2) Given a feature ranking for a particular prediction, (3) complementary data is searched in external datasources to enrich the explanation. (4) Finally, an explanation is assembled based on relevant information gathered concerning the forecast.

In [15], we studied how explanations given to the user could be enriched with external sources, searching for news media entries and metadata regarding datasets. Searches were performed based on keywords that described features' semantic abstractions and new keywords identified from retrieved news media entries. The entries were then ranked based on the word movers distance ([94]) between embeddings. In this research, we adopted a different strategy and (i) computed wiki concepts for the features' abstraction keywords (see Table 1), (ii) ranked news media entries based on the Jaccard distance (see Equation (2))

between relevant reference wiki concepts and the ones obtained by wikifying the news media events [95], (iii) queried and ranked external data sources based on reference wiki concepts and the most important concepts that emerged from news media events, and (iv) further enriched the explanations by adding most relevant entries from the Google KG queried with the most relevant wiki concepts obtained from news media events (Media Events' Keywords and Wiki Concepts (Media Events' K&WC)). However, we found that the most frequent wiki concepts in media news referred to persons and places, which were not informative to the task at hand. Therefore, we decided to filter them out to enhance the quality of the outcomes.

Table 1. Mapping between feature keywords and wiki concepts. Wikification was performed invoking the service available at <https://wikifier.org> (accessed on 27 April 2022). Note that while most concepts were accurate, the wikification of the *Purchasing Managers' Index* produced a wrong concept.

Feature Keywords	Wiki Concepts
Car Sales Demand	Car
	Demand
New Car Sales	Car
	Sales
Vehicle Sales	Vehicle
Car Demand	Car
	Demand
Automotive Industry	Automotive Industry
Global Gross Domestic Product Projection	Gross Domestic Product
	Gross World Product
Global Economic Outlook	Economy
	World economy
Economic Forecast	Forecasting
	Economy
Unemployment Rate	Unemployment
Unemployment Numbers	Unemployment
Unemployment Report	Unemployment
Employment Growth	Employment
Long-term Unemployment	Unemployment
Purchasing Managers' Index	Manager (Gaelic games)

4. Evaluation and Results

Our primary interest in this research was to use semantic technologies and external data sources to enrich the explanations. We consider that explanations can be enriched with (a) information contextual to existing domain knowledge, (b) metadata regarding datasets, which can be used to enrich future models, and (c) relevant concepts obtained by integrating with semantic tools and knowledge graphs. We realized (a) by querying events informed in media news, which could potentially explain a given forecast. Furthermore, we realized (b) by querying open dataset portals, and (c) by retrieving wiki concepts associated with media news pieces of interest and the results obtained from the Google KG.

$$RDE = \frac{\text{Unique Entries}}{\text{Total Listed Entries}} \quad (3)$$

To evaluate the outcomes of such enrichment, we used two metrics to assess whether the entries were precise and diverse: (a) Average Precision@K and the RDE@K (see Equation (3)). The first metric allowed us to measure how much of the information we displayed was related to the underlying model's features. Given that the users rely on trusted automation [96], we consider that exact results are required to increase the users' trust in the underlying application. Furthermore, considering that curiosity is related to the workers' engagement, we consider a diverse set of entries is preferred to foster it. We quantify such diversity through the RDE@K metric. While the entries do not repeat themselves in a single forecast explanation, nothing prevents having the duplicate entries listed through different forecast explanations. We consider that the best case could avoid repeated entries to maximize users' learning. Nevertheless, different strategies could be adopted. One such strategy could be to frame the entries displayed as a recommender system problem and consider users' implicit and explicit feedback to rank and decide whether and where to display them [97–99].

We present our results in Table 2. We compare the results obtained in this research (the *semantics-based approach* described in Section 3) with those obtained in our previous work (*embeddings-based approach* [15]). We found that the former approach had better performance when considering the media events' diversity, trading a slight decay in precision compared with the embeddings-based approach. Regarding the Media Events' K&WC, the embeddings-based approach achieved better performance for both diversity and precision. While differences in precision were contested, they were pronounced regarding diversity. We consider it is natural to have a less numerous set of wiki concepts compared to keywords. Nevertheless, we consider that the metric values could be improved. On the other hand, the semantics-based approach was much more precise when recommending datasets and displayed a slightly higher diversity. Finally, when evaluating the entries related to the Google KG, we observed that the diversity was similar to the one obtained for media news with the embeddings-based approach. The first results were precise, but the precision dropped by 0.30 points when considering $k = 3$. Furthermore, we found that those considered erroneous were mostly related to the economy or automotive industry when analyzing the entries. Nevertheless, they were not useful for the explanations at hand since they referenced banks (e.g., Bank of America or Deutsche Bank) or prominent figures (e.g., Edward Fulton Denison, who pioneered the measurement of the United States' Gross Domestic Product, or Kathleen Wilson-Thompson, independent director at Tesla Motors Inc. at the time of this writing). On the other hand, we considered that the entries were accurate when they referenced companies from the automotive sector (e.g., Faurecia, Rivian Automotive Inc., Polestar or Vinfast) or related to it (e.g., *plug power* which develops hydrogen fuel cell systems to replace conventional batteries, or the *Flinkster* carsharing company).

Considering the results regarding Media Events' K&WC and external datasets, we consider that the embeddings-based approach is best to obtain new keywords and concepts later displayed to the users. In contrast, the semantics-based approach provides the best concepts that lead to better results when searching for external datasets' metadata.

Finally, based on the work of [17], we evaluated the goodness of the overall explanations provided to the user based on a score similar to the degree of explainability, computed as the coverage of five questions related to the main archetypes from the abstract meaning representation theory [100]: (i) Why? (ii) How? (iii) What for? (iv) What if? (v) When? Our explanations achieved a score of 0.6, providing information regarding (i) the factors leading to such a forecast (*why is such a forecast issued?*), (ii) how such an outcome can be changed (relevant, actionable aspect), and (iii) when will the forecast demand take place and when did the relevant events influencing the demand occur. The explanations can be improved with information regarding the purpose of the action (increasing or decreasing the demand) and providing capabilities to explore "what-if" scenarios.

Table 2. Results we obtained by analyzing the forecast explanations created for 56 products over three months. The best results are shown in bold. *Media events*, *Media events' keywords*, *external datasets* and *Google KG* correspond to contextual information displayed for each forecast explanation. *NA* is used for entries regarding the results concerning the Google KG for the embeddings-based approach, given that the research we compare against did not provide such results.

	Metric	Embeddings-Based Approach	Semantics-Based Approach
Media Events	Average Precision@1	0.97	0.95
	Average Precision@3	0.97	0.95
	RDE@1	0.30	0.38
	RDE@3	0.11	0.14
Media Events' K&WC	Average Precision@1	0.77	0.71
	Average Precision@3	0.78	0.72
	RDE@1	0.14	0.01
	RDE@3	0.09	0.01
External Datasets	Average Precision@1	0.56	0.68
	RDE@1	0.41	0.43
Google KG	Average Precision@1	NA	0.76
	Average Precision@3	NA	0.46
	RDE@1	NA	0.15
	RDE@3	NA	0.09

5. Conclusions and Future Work

Along with the increasing adoption of artificial intelligence in manufacturing, explainability techniques must be developed to ensure that users' can learn the models' behaviors. Furthermore, explanations provided to the user can be enriched with additional insights that foster the users' curiosity, resulting in an exploratory dynamic toward artificial intelligence application and domain-specific problems, enabling the development of trust toward such applications. One way to achieve this goal is to enrich the explanations with information obtained from external sources to augment users' knowledge and help them make responsible decisions.

This research explored augmenting explanations by incorporating media news, datasets' metadata, and information queried from open knowledge graphs. Furthermore, we compared two approaches (based on embeddings and wiki concepts) to rank the data sources' entries and retrieve new concepts and keywords from them. We found that results were similar when considering media events but differed for new keywords and concepts (embeddings-based approach was best) and external datasets (semantic-based approach was best). We found that the embeddings-based and semantic-based approaches could complement each other regarding keywords and concepts extracted from media events. While the embeddings-based approach resulted in better scores regarding keywords and concepts shown to the user, the semantics-based approach led to better results when searching for datasets' metadata. Finally, while the integration with the Google KG proved informative, new strategies must be explored to increase the precision of results when showing multiple entries.

We consider that such an approach toward building explanations can be extended to other use cases. Among its strengths, we consider the capability to explore real-world events impacting the forecasting model and the ability to identify complementary data that can be used to enhance the forecasting model. On the other hand, the approach requires manually associating each feature with higher-level concepts and a set of keywords characterizing them. This requires that the features used to train the model have a meaning intelligible to

the human—a requirement always satisfied by handcrafted features. Another limitation is the need to specify a time frame within which feature-related events are expected to happen, given the features' time of occurrence.

In future work, we would like to explore how critical components of an explanation encoded in a graph structure can be used to enhance the explanations and whether users' feedback can lead to better explanations if we frame them as a recommender system problem.

Author Contributions: Conceptualization, J.R.; methodology, J.R.; software, E.T. and J.R.; validation, J.R.; formal analysis, J.R.; investigation, J.R.; resources, J.R., P.Z. and E.T.; data curation, J.R.; writing—original draft preparation, J.R.; writing—review & editing, J.R. and K.K.; visualization, J.R.; supervision, D.M. and B.F.; project administration, K.K. and I.N.; funding acquisition, K.K., B.F. and D.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Slovenian Research Agency and the European Union's Horizon 2020 program projects FACTLOG under grant agreement H2020-869951, and STAR under grant agreement number H2020-956573.

Data Availability Statement: No data were released.

Acknowledgments: This document is the property of the STAR consortium and shall not be distributed or reproduced without the formal approval of the STAR Management Committee. The content of this report reflects only the authors' view. The European Commission is not responsible for any use that may be made of the information it contains.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

GDPR	General Data Protection Regulation
Google KG	Google Knowledge Graph
LACE	Local Agnostic attribute Contribution Explanation
LIME	Local Interpretable Model-agnostic Explanations
LoRE	Local Rule-based Explanation
Media Events' K&WC	Media Events' Keywords and Concepts
RDE	Ratio of Diverse Entries
SHAP	Shapley Additive Explanations

References

1. Lasi, H.; Fettke, P.; Kemper, H.G.; Feld, T.; Hoffmann, M. Industry 4.0. *Bus. Inf. Syst. Eng.* **2014**, *6*, 239–242. [\[CrossRef\]](#)
2. Erro-Garcés, A. Industry 4.0: defining the research agenda. *Benchmarking Int. J.* **2019**, *28*, 1858–1882. [\[CrossRef\]](#)
3. Maddikunta, P.K.R.; Pham, Q.V.; Prabadevi, B.; Deepa, N.; Dev, K.; Gadekallu, T.R.; Ruby, R.; Liyanage, M. Industry 5.0: A survey on enabling technologies and potential applications. *J. Ind. Inf. Integr.* **2021**, *26*, 100257. [\[CrossRef\]](#)
4. Lu, Y. Cyber physical system (CPS)-based industry 4.0: A survey. *J. Ind. Integr. Manag.* **2017**, *2*, 1750014. [\[CrossRef\]](#)
5. Shafto, M.; Conroy, M.; Doyle, R.; Glaessgen, E.; Kemp, C.; LeMoigne, J.; Wang, L. Draft modeling, simulation, information technology & processing roadmap. *Technol. Area* **2012**, *32*, 1–38.
6. Arinez, J.F.; Chang, Q.; Gao, R.X.; Xu, C.; Zhang, J. Artificial intelligence in advanced manufacturing: Current status and future outlook. *J. Manuf. Sci. Eng.* **2020**, *142*, 110804. [\[CrossRef\]](#)
7. Ghobakhloo, M. Industry 4.0, digitization, and opportunities for sustainability. *J. Clean. Prod.* **2020**, *252*, 119869. [\[CrossRef\]](#)
8. Martynov, V.V.; Shavaleeva, D.N.; Zaytseva, A.A. Information technology as the basis for transformation into a digital society and industry 5.0. In Proceedings of the 2019 International Conference “Quality Management, Transport and Information Security, Information Technologies” (IT&QM&IS), Sochi, Russia, 23–27 September 2019; pp. 539–543.
9. Rožanec, J.M.; Novalija, I.; Zajec, P.; Kenda, K.; Tavakoli, H.; Suh, S.; Veliou, E.; Papamartzivanos, D.; Giannetsos, T.; Menesidou, S.A.; et al. Human-Centric Artificial Intelligence Architecture for Industry 5.0 Applications. *arXiv* **2022**, arXiv:2203.10794.
10. Rožanec, J.M.; Kažič, B.; Škrjanc, M.; Fortuna, B.; Mladenčić, D. Automotive OEM demand forecasting: A comparative study of forecasting algorithms and strategies. *Appl. Sci.* **2021**, *11*, 6787. [\[CrossRef\]](#)
11. Trajkova, E.; Rožanec, J.M.; Dam, P.; Fortuna, B.; Mladenčić, D. Active Learning for Automated Visual Inspection of Manufactured Products. *arXiv* **2021**, arXiv:2109.02469.
12. Bhatt, P.M.; Malhan, R.K.; Shembekar, A.V.; Yoon, Y.J.; Gupta, S.K. Expanding capabilities of additive manufacturing through use of robotics technologies: A survey. *Addit. Manuf.* **2020**, *31*, 100933. [\[CrossRef\]](#)

13. Dhanorkar, S.; Wolf, C.T.; Qian, K.; Xu, A.; Popa, L.; Li, Y. Who needs to know what, when?: Broadening the Explainable AI (XAI) Design Space by Looking at Explanations Across the AI Lifecycle. In Proceedings of the Designing Interactive Systems Conference 2021, Virtual Event, 28 June–2 July 2021; pp. 1591–1602.
14. Dragoni, M.; Donadello, I. *A Knowledge-Based Strategy for XAI: The Explanation Graph*; IOS Press: Amsterdam, The Netherlands, 2022.
15. Rožanec, J.M.; Fortuna, B.; Mladenčić, D. Knowledge graph-based rich and confidentiality preserving Explainable Artificial Intelligence (XAI). *Inf. Fusion* **2022**, *81*, 91–102. [CrossRef]
16. Rožanec, J.M.; Zajec, P.; Kenda, K.; Novalija, I.; Fortuna, B.; Mladenčić, D. XAI-KG: knowledge graph to support XAI and decision-making in manufacturing. In *Proceedings of the International Conference on Advanced Information Systems Engineering*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 167–172.
17. Sovrano, F.; Vitali, F. An Objective Metric for Explainable AI: How and Why to Estimate the Degree of Explainability. *arXiv* **2021**, arXiv:2109.05327.
18. Majstorovic, V.D.; Mitrovic, R. Industry 4.0 programs worldwide. In *Proceedings of the International Conference on the Industry 4.0 Model for Advanced Manufacturing*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 78–99.
19. Bogoviz, A.V.; Osipov, V.S.; Chistyakova, M.K.; Borisov, M.Y. Comparative analysis of formation of industry 4.0 in developed and developing countries. In *Industry 4.0: Industrial Revolution of the 21st Century*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 155–164.
20. Raj, A.; Dwivedi, G.; Sharma, A.; de Sousa Jabbour, A.B.L.; Rajak, S. Barriers to the adoption of industry 4.0 technologies in the manufacturing sector: An inter-country comparative perspective. *Int. J. Prod. Econ.* **2020**, *224*, 107546. [CrossRef]
21. Frank, A.G.; Dalenogare, L.S.; Ayala, N.F. Industry 4.0 technologies: Implementation patterns in manufacturing companies. *Int. J. Prod. Econ.* **2019**, *210*, 15–26. [CrossRef]
22. Ghobakhloo, M. The future of manufacturing industry: a strategic roadmap toward Industry 4.0. *J. Manuf. Technol. Manag.* **2018**, *29*, 910–936. [CrossRef]
23. Zheng, T.; Ardolino, M.; Bacchetti, A.; Perona, M. The applications of Industry 4.0 technologies in manufacturing context: A systematic literature review. *Int. J. Prod. Res.* **2021**, *59*, 1922–1954. [CrossRef]
24. Qi, Q.; Tao, F.; Zuo, Y.; Zhao, D. Digital twin service towards smart manufacturing. *Procedia Cirp* **2018**, *72*, 237–242. [CrossRef]
25. Rožanec, J.M.; Lu, J.; Rupnik, J.; Škrjanc, M.; Mladenčić, D.; Fortuna, B.; Zheng, X.; Kiritsis, D. Actionable cognitive twins for decision making in manufacturing. *Int. J. Prod. Res.* **2022**, *60*, 452–478. [CrossRef]
26. Xu, X.; Lu, Y.; Vogel-Heuser, B.; Wang, L. Industry 4.0 and Industry 5.0—Inception, conception and perception. *J. Manuf. Syst.* **2021**, *61*, 530–535. [CrossRef]
27. Nahavandi, S. Industry 5.0—A human-centric solution. *Sustainability* **2019**, *11*, 4371. [CrossRef]
28. Demir, K.A.; Döven, G.; Sezen, B. Industry 5.0 and human-robot co-working. *Procedia Comput. Sci.* **2019**, *158*, 688–695. [CrossRef]
29. Industry 5.0: Towards More Sustainable, Resilient and Human-Centric Industry. Available online: <https://op.europa.eu/en/publication-detail/-/publication/468a892a-5097-11eb-b59f-01aa75ed71a1/> (accessed on 15 March 2022).
30. Weitz, K.; Schiller, D.; Schlagowski, R.; Huber, T.; André, E. “Do you trust me?” Increasing user-trust by integrating virtual agents in explainable AI interaction design. In Proceedings of the 19th ACM International Conference on Intelligent Virtual Agents, Paris, France, 2–5 July 2019; pp. 7–9.
31. Honeycutt, D.; Nourani, M.; Ragan, E. Soliciting human-in-the-loop user feedback for interactive machine learning reduces user trust and impressions of model accuracy. In Proceedings of the AAAI Conference on Human Computation and Crowdsourcing, Virtul, 26–28 October 2020; Volume 8, pp. 63–72.
32. Moroff, N.U.; Kurt, E.; Kamphues, J. Machine Learning and statistics: A Study for assessing innovative demand forecasting models. *Procedia Comput. Sci.* **2021**, *180*, 40–49. [CrossRef]
33. Purohit, D.; Srivastava, J. Effect of manufacturer reputation, retailer reputation, and product warranty on consumer judgments of product quality: A cue diagnosticity framework. *J. Consum. Psychol.* **2001**, *10*, 123–134. [CrossRef]
34. Callon, M.; Méadel, C.; Rabeharisoa, V. The economy of qualities. *Econ. Soc.* **2002**, *31*, 194–217. [CrossRef]
35. Teunter, R.H.; Babai, M.Z.; Syntetos, A.A. ABC classification: service levels and inventory costs. *Prod. Oper. Manag.* **2010**, *19*, 343–352. [CrossRef]
36. Scholz-Reiter, B.; Heger, J.; Meinecke, C.; Bergmann, J. Integration of demand forecasts in ABC-XYZ analysis: Practical investigation at an industrial company. *Int. J. Product. Perform. Manag.* **2012**, *61*, 445–451. [CrossRef]
37. Syntetos, A.A.; Boylan, J.E.; Croston, J. On the categorization of demand patterns. *J. Oper. Res. Soc.* **2005**, *56*, 495–503. [CrossRef]
38. Rožanec, J.M.; Mladenčić, D. Reframing demand forecasting: a two-fold approach for lumpy and intermittent demand. *arXiv* **2021**, arXiv:2103.13812.
39. Brühl, B.; Hülsmann, M.; Borscheid, D.; Friedrich, C.M.; Reith, D. A sales forecast model for the german automobile market based on time series analysis and data mining methods. In *Proceedings of the Industrial Conference on Data Mining*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 146–160.
40. Vahabi, A.; Hosseininia, S.S.; Alborzi, M. A Sales Forecasting Model in Automotive Industry using Adaptive Neuro-Fuzzy Inference System (Anfis) and Genetic Algorithm (GA). *Management* **2016**, *1*, 1–7. [CrossRef]
41. Ubaidillah, N.Z. A study of car demand and its interdependency in sarawak. *Int. J. Bus. Soc.* **2020**, *21*, 997–1011. [CrossRef]
42. Dwivedi, A.; Niranjana, M.; Sahu, K. A business intelligence technique for forecasting the automobile sales using Adaptive Intelligent Systems (ANFIS and ANN). *Int. J. Comput. Appl.* **2013**, *74*, 1–7. [CrossRef]

43. Wang, X.; Zeng, D.; Dai, H.; Zhu, Y. Making the right business decision: Forecasting the binary NPD strategy in Chinese automotive industry with machine learning methods. *Technol. Forecast. Soc. Chang.* **2020**, *155*, 120032. [\[CrossRef\]](#)
44. Chandriah, K.K.; Naraganahalli, R.V. RNN/LSTM with modified Adam optimizer in deep learning approach for automobile spare parts demand forecasting. *Multimed. Tools Appl.* **2021**, *80*, 26145–26159. [\[CrossRef\]](#)
45. Halford, G.S.; Baker, R.; McCredden, J.E.; Bain, J.D. How many variables can humans process? *Psychol. Sci.* **2005**, *16*, 70–76. [\[CrossRef\]](#) [\[PubMed\]](#)
46. Barnes, J.H., Jr. Cognitive biases and their impact on strategic planning. *Strateg. Manag. J.* **1984**, *5*, 129–137. [\[CrossRef\]](#)
47. Adadi, A.; Berrada, M. Peeking inside the black-box: A survey on explainable artificial intelligence (XAI). *IEEE Access* **2018**, *6*, 52138–52160. [\[CrossRef\]](#)
48. Arrieta, A.B.; Díaz-Rodríguez, N.; Del Ser, J.; Bannetot, A.; Tabik, S.; Barbado, A.; García, S.; Gil-López, S.; Molina, D.; Benjamins, R.; et al. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Inf. Fusion* **2020**, *58*, 82–115. [\[CrossRef\]](#)
49. Confalonieri, R.; Coba, L.; Wagner, B.; Besold, T.R. A historical perspective of explainable Artificial Intelligence. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2021**, *11*, e1391. [\[CrossRef\]](#)
50. Davydenko, A.; Fildes, R.A.; Trapero Arenas, J. *Judgmental Adjustments to Demand Forecasts: Accuracy Evaluation and Bias Correction*; The Department of Management Science, Lancaster University: Lancaster, UK, 2010.
51. Davydenko, A.; Fildes, R. Measuring forecasting accuracy: The case of judgmental adjustments to SKU-level demand forecasts. *Int. J. Forecast.* **2013**, *29*, 510–522. [\[CrossRef\]](#)
52. Alvarado-Valencia, J.; Barrero, L.H.; Önköl, D.; Dennerlein, J.T. Expertise, credibility of system forecasts and integration methods in judgmental demand forecasting. *Int. J. Forecast.* **2017**, *33*, 298–313. [\[CrossRef\]](#)
53. Goodman, B.; Flaxman, S. European Union regulations on algorithmic decision-making and a “right to explanation”. *AI Mag.* **2017**, *38*, 50–57. [\[CrossRef\]](#)
54. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the Protection of Natural Persons with Regard to the Processing of Personal Data and on the Free Movement of such Data, and Repealing Directive 95/46/EC (General Data Protection Regulation). Available online: <https://eur-lex.europa.eu/eli/reg/2016/679/oj> (accessed on 15 March 2022).
55. Proposal for a Regulation of The European Parliament and of The Council Laying Down Harmonised Rules on Artificial Intelligence (Artificial Intelligence Act) and Amending Certain Union Legislative Acts. Available online: <https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:52021PC0206> (accessed on 15 March 2022).
56. Anitha, J. Determinants of employee engagement and their impact on employee performance. *Int. J. Product. Perform. Manag.* **2014**, *63*, 308–323.
57. Emmert-Streib, F.; Yli-Harja, O.; Dehmer, M. Explainable artificial intelligence and machine learning: A reality rooted perspective. *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.* **2020**, *10*, e1368. [\[CrossRef\]](#)
58. Schwalbe, G.; Finzel, B. XAI Method Properties: A (Meta-) study. *arXiv* **2021**, arXiv:2105.07190.
59. Chan, L. Explainable AI as Epistemic Representation. Available online: https://aisb.org.uk/wp-content/uploads/2021/04/AISB21_Opacity_Proceedings.pdf#page=9 (accessed on 25 April 2022).
60. Müller, V.C. Deep Opacity Undermines Data Protection and Explainable Artificial Intelligence. Available online: http://explanations.ai/symposium/AISB21_Opacity_Proceedings.pdf#page=20 (accessed on 25 April 2022).
61. Ribeiro, M.T.; Singh, S.; Guestrin, C. “Why should i trust you?” Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 1135–1144.
62. Zafar, M.R.; Khan, N.M. DLIME: A deterministic local interpretable model-agnostic explanations approach for computer-aided diagnosis systems. *arXiv* **2019**, arXiv:1906.10263.
63. Hall, P.; Gill, N.; Kurka, M.; Phan, W. Machine Learning Interpretability with H₂O Driverless AI. 2017. Available online: <http://docs.h2o.ai/driverless-ai/latest-stable/docs/booklets/MLIBooklet.pdf> (accessed on 25 April 2022).
64. Sokol, K.; Flach, P. LIMETree: Interactively Customisable Explanations Based on Local Surrogate Multi-output Regression Trees. *arXiv* **2020**, arXiv:2005.01427.
65. Lundberg, S.M.; Lee, S.I. A unified approach to interpreting model predictions. Available online: <https://proceedings.neurips.cc/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf> (accessed on 25 April 2022).
66. Strumbelj, E.; Kononenko, I. An efficient explanation of individual classifications using game theory. *J. Mach. Learn. Res.* **2010**, *11*, 1–18.
67. Pastor, E.; Baralis, E. Explaining black box models by means of local rules. In Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing, Limassol, Cyprus, 8–12 April 2019; pp. 510–517.
68. Guidotti, R.; Monreale, A.; Ruggieri, S.; Pedreschi, D.; Turini, F.; Giannotti, F. Local rule-based explanations of black box decision systems. *arXiv* **2018**, arXiv:1805.10820.
69. Ribeiro, M.T.; Singh, S.; Guestrin, C. Anchors: High-Precision Model-Agnostic Explanations. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 18, pp. 1527–1535.
70. Van der Waa, J.; Robeer, M.; van Diggelen, J.; Brinkhuis, M.; Neerinx, M. Contrastive explanations with local foil trees. *arXiv* **2018**, arXiv:1806.07470.

71. Rožanec, J.; Trajkova, E.; Kenda, K.; Fortuna, B.; Mladenčić, D. Explaining Bad Forecasts in Global Time Series Models. *Appl. Sci.* **2021**, *11*, 9243. [CrossRef]
72. Confalonieria, R.; Galliana, P.; Kutza, O.; Porellob, D.; Righettia, G.; Troquarda, N. Towards Knowledge-driven Distillation and Explanation of Black-box Models. In Proceedings of the International Workshop on Data meets Applied Ontologies in Explainable AI (DAO-XAI 2021), Bratislava, Slovakia, 18–19 September 2021.
73. Panigutti, C.; Perotti, A.; Pedreschi, D. Doctor XAI: An ontology-based approach to black-box sequential data classification explanations. In Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency, Barcelona, Spain, 27–30 January 2020; pp. 629–639.
74. Lécué, F.; Abeloos, B.; Anctil, J.; Bergeron, M.; Dalla-Rosa, D.; Corbeil-Letourneau, S.; Martet, F.; Pommellet, T.; Salvan, L.; Veilleux, S.; et al. Thales XAI Platform: Adaptable Explanation of Machine Learning Systems-A Knowledge Graphs Perspective. In Proceedings of the ISWC Satellites, Auckland, New Zealand, 26–30 October 2019; pp. 315–316.
75. Rabold, J.; Deininger, H.; Siebers, M.; Schmid, U. Enriching visual with verbal explanations for relational concepts—combining LIME with Aleph. In *Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 180–192.
76. Lakkaraju, H.; Kamar, E.; Caruana, R.; Leskovec, J. Interpretable & explorable approximations of black box models. *arXiv* **2017**, arXiv:1707.01154.
77. Nguyen, A.p.; Martínez, M.R. On quantitative aspects of model interpretability. *arXiv* **2020**, arXiv:2007.07584.
78. Rosenfeld, A. Better metrics for evaluating explainable artificial intelligence. In Proceedings of the 20th international Conference on Autonomous Agents and Multiagent Systems, Virtual, 3–7 May 2021; pp. 45–50.
79. Amparore, E.; Perotti, A.; Bajardi, P. To trust or not to trust an explanation: using LEAF to evaluate local linear XAI methods. *PeerJ Comput. Sci.* **2021**, *7*, e479. [CrossRef]
80. Samek, W.; Müller, K.R. Towards explainable artificial intelligence. In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 5–22.
81. Pedreschi, D.; Giannotti, F.; Guidotti, R.; Monreale, A.; Pappalardo, L.; Ruggieri, S.; Turini, F. Open the black box data-driven explanation of black box decision systems. *arXiv* **2018**, arXiv:1806.09936.
82. El-Assady, M.; Jentner, W.; Kehlbeck, R.; Schlegel, U.; Sevastjanova, R.; Sperrle, F.; Spinner, T.; Keim, D. Towards XAI: Structuring the Processes of Explanations. In Proceedings of the ACM Workshop on Human-Centered Machine Learning, Glasgow, UK, 4 May 2019.
83. Hsiao, J.H.w.; Ngai, H.H.T.; Qiu, L.; Yang, Y.; Cao, C.C. Roadmap of designing cognitive metrics for explainable artificial intelligence (XAI). *arXiv* **2021**, arXiv:2108.01737.
84. Hoffman, R.R.; Mueller, S.T.; Klein, G.; Litman, J. Metrics for explainable AI: Challenges and prospects. *arXiv* **2018**, arXiv:1812.04608.
85. Keane, M.T.; Smyth, B. Good counterfactuals and where to find them: A case-based technique for generating counterfactuals for explainable AI (XAI). In *Proceedings of the International Conference on Case-Based Reasoning*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 163–178.
86. Keane, M.T.; Kenny, E.M.; Delaney, E.; Smyth, B. If only we had better counterfactual explanations: five key deficits to rectify in the evaluation of counterfactual XAI techniques. *arXiv* **2021**, arXiv:2103.01035.
87. Verma, S.; Dickerson, J.; Hines, K. Counterfactual Explanations for Machine Learning: A Review. *arXiv* **2020**, arXiv:2010.10596.
88. Mohseni, S.; Zarei, N.; Ragan, E.D. A multidisciplinary survey and framework for design and evaluation of explainable AI systems. *Acm Trans. Interact. Syst. (Tiis)* **2021**, *11*, 1–45. [CrossRef]
89. Lage, I.; Ross, A.S.; Kim, B.; Gershman, S.J.; Doshi-Velez, F. Human-in-the-loop interpretability prior. *arXiv* **2018**, arXiv:1805.11571.
90. Rozanec, J.M. Explainable demand forecasting: A data mining goldmine. In Proceedings of the Companion Proceedings of the Web Conference 2021, Ljubljana, Slovenia, 19–23 April 2021; pp. 723–724.
91. Leban, G.; Fortuna, B.; Brank, J.; Grobelnik, M. Event registry: Learning about world events from news. In Proceedings of the 23rd International Conference on World Wide Web, Seoul, Korea, 7–11 April 2014; pp. 107–110.
92. Publications Office of the European Union. EU Open Data Portal: The Official Portal for European Data. Available online: <https://data.europa.eu> (accessed on 15 December 2020).
93. Noy, N.; Gao, Y.; Jain, A.; Narayanan, A.; Patterson, A.; Taylor, J. Industry-scale knowledge graphs: Lessons and challenges. *Queue* **2019**, *17*, 48–75. [CrossRef]
94. Kusner, M.; Sun, Y.; Kolkin, N.; Weinberger, K. From word embeddings to document distances. In Proceedings of the International Conference on Machine Learning, Lille, France, 6–11 July 2015; pp. 957–966.
95. Brank, J.; Leban, G.; Grobelnik, M. Annotating Documents with Relevant Wikipedia Concepts. Available online: https://ailab.ijs.si/Dunja/SiKDD2017/Papers/Brank_Wikifier.pdf (accessed on 25 April 2022).
96. Lee, J.D.; See, K.A. Trust in automation: Designing for appropriate reliance. *Hum. Factors* **2004**, *46*, 50–80. [CrossRef]
97. Kilani, Y.; Alhijawi, B.; Alsarhan, A. Using artificial intelligence techniques in collaborative filtering recommender systems: Survey. *Int. J. Adv. Intell. Paradig.* **2018**, *11*, 378–396. [CrossRef]
98. Karimi, M.; Jannach, D.; Jugovac, M. News recommender systems—Survey and roads ahead. *Inf. Process. Manag.* **2018**, *54*, 1203–1227. [CrossRef]

-
99. Sidana, S.; Trofimov, M.; Horodnytskyi, O.; Laclau, C.; Maximov, Y.; Amini, M.R. User preference and embedding learning with implicit feedback for recommender systems. *Data Min. Knowl. Discov.* **2021**, *35*, 568–592. [[CrossRef](#)]
 100. Michael, J.; Stanovsky, G.; He, L.; Dagan, I.; Zettlemoyer, L. Crowdsourcing question-answer meaning representations. *arXiv* **2017**, arXiv:1711.05885.

4.2 Explaining Bad Forecasts in Global Time Series Models

In our work *Automotive OEM Demand Forecasting: A Comparative Study of Forecasting Algorithms and Strategies* [36] (published in MDPI Applied Sciences in 2022 (Q2 - IF 2.838)), we noticed that global models could produce erroneous demand forecasts based on patterns learned from the many products considered when training a machine learning model. Therefore, to trust such models, we must identify whether the prediction is reasonable for a given product or could be wrong and why it could be considered anomalous. To that end, we envisioned an application that would perform anomaly detection considering multiple criteria and inform the user whether the forecasted value could be regarded as anomalous in the context of past observed values for a given time series and the criteria under which an anomaly detector considers it anomalous. While it is important to detect anomalous forecasts and warn the end user about them, a step further can be taken to provide additional information to the machine learning engineers to iterate and correct the model. With a view to this purpose, we analyzed relevant features to a particular forecast, retrieved training instances that could have influenced the model to learn the way a forecast was observed, and provided counterfactual examples of how changes in a feature vector can lead to a different example.

While the work has shown promising results, we have not used it in a real-world setting. We consider user feedback would be critical to assess its usefulness and determine how the information displayed could be useful to persons consuming machine learning forecasts and to machine learning engineers developing and debugging machine learning models.

Article

Explaining Bad Forecasts in Global Time Series Models

Jože Rožanec ^{1,2,3,*} , Elena Trajkova ^{1,4} , Klemen Kenda ^{1,2,3} , Blaž Fortuna ^{1,2}  and Dunja Mladenec ¹

¹ Jožef Stefan Institute, Jamova 39, 1000 Ljubljana, Slovenia; trajkova.elena.00@gmail.com (E.T.); klemen.kenda@ijs.si (K.K.); blaz.fortuna@ijs.si (B.F.); dunja.mladenec@ijs.si (D.M.)

² Qlector d.o.o., Rovšnikova 7, 1000 Ljubljana, Slovenia

³ Jožef Stefan International Postgraduate School, Jamova 39, 1000 Ljubljana, Slovenia

⁴ Faculty of Electrical Engineering, University of Ljubljana, Tržaška 25, 1000 Ljubljana, Slovenia

* Correspondence: joze.rozanec@ijs.si

Featured Application: The outcomes of this work can be applied to understand better when and why global time series forecasting models issue incorrect predictions and iteratively groom the dataset to enhance the models' performance.

Abstract: While increasing empirical evidence suggests that global time series forecasting models can achieve better forecasting performance than local ones, there is a research void regarding when and why the global models fail to provide a good forecast. This paper uses anomaly detection algorithms and explainable artificial intelligence (XAI) to answer when and why a forecast should not be trusted. To address this issue, a dashboard was built to inform the user regarding (i) the relevance of the features for that particular forecast, (ii) which training samples most likely influenced the forecast outcome, (iii) why the forecast is considered an outlier, and (iv) provide a range of counterfactual examples to understand how value changes in the feature vector can lead to a different outcome. Moreover, a modular architecture and a methodology were developed to iteratively remove noisy data instances from the train set, to enhance the overall global time series forecasting model performance. Finally, to test the effectiveness of the proposed approach, it was validated on two publicly available real-world datasets.

Keywords: explainable artificial intelligence; XAI; time series forecasting; global time series models; machine learning; artificial intelligence



Citation: Rožanec, J.; Trajkova, E.; Kenda, K.; Fortuna, B.; Mladenec, D. Explaining Bad Forecasts in Global Time Series Models. *Appl. Sci.* **2021**, *11*, 9243. <https://doi.org/10.3390/app11199243>

Academic Editor: Tobias Meisen

Received: 9 August 2021

Accepted: 24 September

Published: 4 October 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Time series forecasting is a relevant problem with application in many domains [1], gaining further relevance with the increasing availability of historical data [2]. Historically, much research focused on time series models trained on a single time series [3]. Though global machine learning models issued good results in the past, they gained new attention with the advent of deep learning [4], and recent success on the M4 and M5 time series forecasting competitions [5,6]. Such models can learn patterns shared across multiple time series, enhancing the overall forecasting performance. The usage of multiple time series to train the model can be considered a source of explainability: each forecast can be explained not only through past behavior on a single time series, but similar patterns can be found in other time series, providing a different perspective and complementary insights [7]. The ability to develop global time series models implicates scaling advantages too: it reduces the number of forecasting models, and thus the amount of human supervision required to build them, and fewer deployments, monitoring, and maintenance [8].

While authors have shown that global time series machine learning forecasting models (GTSMFLM) can achieve better performance overall, “understanding when and why global forecasting models work, is arguably the most important open problem currently in time series forecasting” [3]. It is thus crucial to develop Explainable Artificial Intelligence (XAI)

approaches to answer those questions. The approaches can differ based on the goal they aim to tackle (e.g., increase trust in the model or provide insights that can be used to improve the model), the user profile they target and focus (if the explanations provided are either local or global) [9].

The paper focuses on understanding when and why GTSMFLMs work, providing relevant information to the end-users and the machine learning engineers. The end-users must understand if a particular forecast instance can be relied upon (e.g., detect if it can be considered anomalous) and what features do influence the forecast. If the forecast is anomalous, the user can be interested in getting to know some counterfactual examples. While machine learning engineers will appreciate this information, they can also find which instances from the train set most likely influenced the models' learning to issue an unlikely forecast. To provide such an understanding to the users, we use local features relevance, anomaly detection models, and develop a novel approach to provide counterfactual explanations for regression methods. Furthermore, we integrate the insights into a dashboard that can serve the end-users and the machine learning engineers, to understand better when and why global time series forecasting models work.

This research provides several contributions, with which we address the gap of providing means to answer when and why a GTSMFLM prediction should be considered a bad forecast [3]. First, we make use of anomaly detection algorithms to identify potentially wrong forecasts. Then, we compute the models' feature attribution for those data instances, identify similar data instances in the train set, and create counterfactual examples. Furthermore, we develop a dashboard integrating the insights mentioned above and a line plot showing the time series, with a relevance heatmap overlay to inform the influence of data points on a given forecast. While such relevant heatmaps have been widely applied for deep learning models, they are not frequently used for other time series models. In addition, such a dashboard enables a visual inspection of time series and forecasts—a valuable tool for end-users and machine learning engineers. Finally, based on our experience, we outline an architecture and a methodology that can be followed to enhance the dataset and GTSMFLMs iteratively.

To evaluate our approach, we conduct a series of experiments and perform a quantitative evaluation to measure the impact of the insights are developed when engineering the GTSMFLM. In particular, we measure the Mean Absolute Scaled Error (MASE) [10], the number of outliers observed in the test set, and the number of instances removed from the train set, based on the detected anomalous predictions.

We organized the remainder of this paper as follows. In Section 2, we review related scientific works; in Section 3, we introduce the proposed architecture, while in Section 4, we provide details on the methodology that we followed to treat the dataset and train new GTSMFLMs iteratively. In Section 5, we describe two different time series datasets that we used to test our approach, detailing the preprocessing steps and features that we created. In Section 6, we describe the experiments that we performed, provide a more detailed overview regarding the metrics used to measure the results, and the results we obtained, while in Section 7 we address the limitations of this research. Finally, in Section 8, we conclude by summarizing this research, and outline future work.

2. Review of Related Scientific Works

2.1. Forecasting Time Series: Local vs. Global Approach

For many decades, most research on time series forecasting assumed that the time series are generated by independent processes and can be tackled as a regression problem, creating a single model per time series (local models) [8,11]. Growing empirical evidence suggests that creating a single model to forecast multiple time series (under the assumption of forecasts' independence for different time series) known as global models can outperform local ones. Seminal research on using global models developed decades ago [12], and the concept was further explored by many researchers afterwards [13]. The first approaches towards global models considered pooling similar time series, which improves the overall

models' accuracy. By having more training data of similar time series, algorithms can better distinguish common data patterns, while also minimizing distortions introduced by outlier data points [14]. On the other side, this approach requires defining some time series grouping criteria, which can lead to suboptimal groupings [15]. Among pooling strategies, we can find model-based clustering [14,16,17], random clustering [8], grouping based on similarity measures [18–20], or expert judgement [21]. Recent research has explored creating global models considering all available time series, regardless of their heterogeneity, obtaining promising results [3,8,22]. Furthermore, it has been demonstrated that, for every dataset, some global model exists that can equal or outperform a local model, regardless of how heterogeneous the data may be [8]. Such models make the strong assumption that some relationship exists between time series, though the forecasts are independent of each other [11].

These insights, the good results obtained by applying global neural network models for time series forecasting [11,23–27], and success of global models at the M4 and M5 competitions [5,6], have renewed the research interest on global models for time series forecasting [3]. In such models, the relationship between time series is not very well understood, and understanding why and when do global time series forecasting models work remains an open research topic [3]. We envision that anomaly detection algorithms can be used to detect when the GTSMFLM provides accurate forecasts or not, and XAI approaches can provide insights to understand better factors affecting the forecasts, and provide prototype (local) explanations [28], and counterfactual examples.

2.2. Time Series Anomaly Detection

One of the open research questions regarding GTSMFLM is, when do such models provide acceptable forecasts [3]? Such response can be obtained from anomaly detection algorithms and models, which can alert on point anomalies. In this section, we provide an overview of anomaly detection techniques, focusing on the ones developed and applied in a time series setting. We use the terms anomaly, outlier, or deviant interchangeably [29], to denote “observations that deviate so much from others as to arouse suspicion that it was generated by another mechanism” [30]. In particular, outliers related to time series data must also consider the behavior across time [31].

Outliers can be characterized into many types. The first characterization of this type can be found in [32], who introduced the concepts of two time series' outlier types: those (a) that affect a single observation (*type I*), and those (b) that affect an observation and the subsequent ones (*type II*). More recently, [31] distinguishes between *point outliers* (single point in the time series, which has an unusual value when compared to the whole time series or the neighboring points) and *subsequence outliers* (points which may not be outliers by themselves, but the sequence arrangement is anomalous). *Subsequence outliers* are further classified into *contextual anomalies* (they are anomalous in the context of the surrounding observations) or *pattern anomalies* (they are anomalous regardless of the surrounding observations) [33]. In this research, we limit ourselves to point outliers.

Anomaly detection techniques are frequently classified into three categories: statistical, distance-based approaches, and model-based approaches [34]. The first anomaly detection techniques were developed in statistics and remained among the most frequently used ones. Non-parametric techniques usually allow fast computations and are adopted where such speed is of primary importance. Among them, we find the histogram-based approaches, which assume feature independence and determine the outliers based on the histogram distribution [35–38]. Other non-parametric approaches are bitmap time series anomaly detectors, which compute the relative frequency of its features to create a bitmap and identify anomalous time series [39,40], and statistical methods that allow estimating outliers based on a kernel density estimation by using a kernel function. Such kernel functions can provide a probability estimate, given that function is a probability density function [34,41–43]. Among the parametric methods, we find the Gaussian methods, such

as box-plot anomaly detection [44], the Gaussian process [45,46], or regression approaches such as least squares regression [47–49].

Statistical anomaly detection methods cannot be applied on datasets with an unknown distribution [38]. Different approaches were developed to overcome this issue. When considering the distance between data points, we find the k-nearest-neighbors (kNN), which determines the outliers based on the kNN distance. While approaches were developed to determine the best k parameter [50], some techniques attempted to avoid dependence on the k-value. One such method is the local outlier factor (LOF) method, which computes the distance from a point to all other points in the dataset [51]. A similar approach was followed at the outlier detection using indegree number (ODIN), which computes the number of instances that contain a given point in their neighborhood. However, a parameter is required to determine the outlier threshold [52]. A different method was developed by [53], who introduced the multi-granularity deviation factor to identify local density variations that lead to isolated outliers or outlying clusters. Variations to the LOF method were developed to solve some of its shortcomings. For example, the connectivity-based outlier factor (COF) aims to capture better clusters where the data points are distributed in a linear manner [54], while influenced outlierness (INFLO) attempts to better discriminate points nearby two clusters with different densities [55]. Finally, to account for the different feature importance, [56] developed an alternative anomaly detection algorithm, using a weighted kNN.

Model-based techniques can be divided into (i) models that learn and predict whether the value is anomalous and (ii) models that compare the potential outlier with expected values drawn from a generative model or data distribution. Since model-based techniques require labeled data, active learning can be utilized to minimize the labeling effort [57]. Among the models of the first group, we find the SVM-based models, such as the one-class support vector machine (OC-SVM), which was introduced by [58], and later enhanced by many authors [59,60]. Since the regular SVM algorithm can provide poor generalization on an imbalanced dataset, the authors suggested representing the anomalous classes with the high dimensional space origin and mapping anomalous instances close to it. Other SVM variants used for anomaly detection include support vector data description (SVDD) [61], and SVM-SVDD [62]. A different intuition is followed in the Isolation Forest. This tree-based model is based on the principle that the fewer instances of anomalies generate a smaller number of partitions, and thus are likely to have short paths in the tree structure [63]. Other models reported in the literature involve the use of random forests [64], gradient boosted machines [65], artificial neural networks [66], or voting ensembles [67]. Models from the second group have multiple configurations, varying the generative methods and outlier detection criteria. Some examples are the use of ARIMA models to predict future time series values and mark incoming readings as anomalies if they exceed a certain threshold when compared with the forecast [68,69]. Other approaches fused statistical methods and ANNs [70], or used ANNs alone [31,71].

Anomaly detection algorithms can identify anomalous forecasts in the context of a particular time series. Based on the algorithm type, insights can be gained on why the point forecast is considered anomalous. Additional insights can be obtained through XAI to understand which features were most influential to such forecast and provide counterfactual examples, highlighting value changes to those features that would produce a better outcome.

2.3. Explainable Artificial Intelligence

The increasing adoption of artificial intelligence demands understanding the logic beneath the forecasts so that a decision can be made as to whether such forecasts can be trusted or not [72]. The sub-field of artificial intelligence devoted to research on obtaining and providing such understanding is called explainable artificial intelligence (XAI). Authors identify two sources of model opacity [73]. The first one is the complexity of the formal structure of the model, which can be beyond human comprehension, or alien to human

reasoning. When the opacity cannot be removed even by human experts, we speak about deep opacity [74]. The second source of opacity is the intentionally induced opaqueness to avoid revealing sensitive model details (e.g., due to their proprietary nature).

Researchers developed multiple approaches to provide black-box explanations of forecasting models. Among the most frequently cited, we find LIME [75] and its variants (e.g., k-LIME [76], DLIME [77], and LIMETree [78]), Anchors [79], Local Foil Trees [80], or LoRE [81]. These approaches build surrogate models for each prediction sample, learning the reference model's behavior on the particular case of interest by introducing perturbations to the feature vector variables. The Shapley additive explanations take a different approach (SHAP) [82,83], which are grounded in cooperative game theory. The feature relevance is computed based on the approximate computation of Shapley values. Shapley values are also used to explain features relevance in a time series setting. In particular, the TimeSHAP implementation measures which features and past events are most relevant to a recurrent model [84].

Research regarding XAI for time series has focused mainly on explainability for deep learning models. One of the first of such methods was introduced by [85], who computed feature attributions by taking the partial derivative of the output class with respect to the input. This method was later improved in the *Gradient*Input* method, which computes neuron and filter activations for a specific instance by multiplying the input by the partial derivative of a layer with respect to the input [86]. Similar approaches followed, such as the Deep Learning Important FeaTures (DeepLIFT) [87], *integrated gradients* [88], or *Smooth-Grad* [89]. The introduction of attention mechanisms to deep learning models was also envisioned as a source of explainability, since it provides insights regarding which points in time are relevant to the forecast [90]. Such information is frequently presented in heatmaps [91,92], saliency maps [85,93], and custom visualizations [94]. Finally, several feature perturbation methods were developed to measure the features' contribution to the forecasted value when such features are removed [95], or masked [96–98].

Methods such as Shapley values [99] and region partition trees [100] have been successfully applied to explain detected anomalies. More systemic approaches have been developed too, such as Exathlon [101], EXAD [102], and others [103]. While EXAD focuses on explanation discovery for each anomaly, Exathlon crafts the explanations, providing two pieces of information to the user: why the data point was identified as an anomaly and the root causes of such anomaly.

Along with feature relevance, it is sometimes important to know how values should change to achieve a different forecast. Such examples are known as counterfactual explanations. When counterfactual explanations are provided as actionable advice, they are known as directive explanations. Directive explanations can be either directive-specific (suggest a concrete action to change the forecasted value), or directive-generic (suggest a generic action to alter the forecasted value) [104]. While most frequently applied to classification models, counterfactual explanations are also applied to regression models. While such counterfactual explanations most frequently use some threshold [105], other implementations were developed based on potentials [106], or satisfiability modulo theories solvers [107], among others.

While counterfactual examples provide value to understand how to change a certain target state, sometimes it is useful to visualize prototypical instances similar to the feature vector used to issue a prediction. Such examples are known as prototype explanations, and allow us to understand better which instances influenced a certain forecast [108].

As described above, multiple methods exist to obtain relevant information that can explain the underlying reasoning of an artificial intelligence model. Along with them, it is also important to consider how such information is presented to the users. Good explanations should convey meaningful information, resemble a logic explanation [109], target a specific user profile [110], focus on actionability, and if possible, provide some counterfactual examples [111]. An explanation should take into account relevant context, which can be captured in three elements: the target user profile, the explanation goals

(e.g., improve the model, or enhance trust in the system), and the focus (if the explanations provided are meant at a global or local level).

When developing our architecture and dashboard, we considered different intuitive ways to present the information, ranging from plots and tables to human-readable sentences. When the forecasts are considered anomalous, it is helpful to the user to understand why such a forecast is considered anomalous, which are the most relevant features to that forecast, provide counterfactual examples, and examples from the training set that could have influenced the forecast.

3. The Proposed Architecture

We propose a modular architecture to provide forecast explanations for global time series forecasting models. We combine anomaly detection and explainability methods to enhance the forecasting precision and provide valuable contextual information to the end-users. The anomaly detection module provides a means to identify potentially bad forecasts. In such cases, we can fall back to a local statistical model, or alert the user that a given forecast should not be trusted given past time series' behavior. The feature relevance informs the user on which variables exercised the most influence on the forecast. We use them as an input when computing the counterfactual examples to understand better what value changes on those variables would produce an outcome that is no longer considered an outlier. Finally, we provide insight to the user on which data instances could have influenced the forecasting models to provide such a forecast by identifying them in the train set based on their similarity regarding the forecasted one. We consider that this information helps investigate possible patterns learned by the model and how to engineer a better model learning in the future.

The architecture (see Figure 1) comprises the following components:

- **Data Module:** provides a dataset to train machine learning GTSMLFMs. The dataset comprises time series data, either considering their raw values and derivative features or a refined version where specific instances that could cause outlier forecasts were treated. The module wraps a set of strategies to find and treat data instances that are similar to the ones producing outlier forecasts, identified by the anomaly detection module.
- **Forecasting Module:** comprises a machine learning GTSMLFM, and a set of local statistical models to forecast the time series. The machine learning GTSMLFM is created based on a dataset obtained from the Data Module. The Forecasting Module makes use of the input provided by the Anomaly Detection Module to decide whether the outcome should be the forecast obtained from a GTSMLFM, or a local statistical model.
- **Anomaly Detection Module:** leverages algorithms and models to analyze the forecast in the context of a time series and classify it as an anomaly or not. It interacts with the Forecasting Module and the XAI Module, providing feedback on whether a forecast can be considered anomalous or not.
- **XAI Module:** uses various XAI algorithms and models to craft forecast explanations for the user. In particular, we envision that this module (i) indicates if the forecast is anomalous, (ii) crafts a text explanation highlighting most relevant features influencing a specific forecast, (iii) shows a sample of n data instances found in the train set that most likely influenced the GTSMLFM towards the given forecast, and (iv) shows a set of counterfactual examples created considering (a) the relevant features to that specific forecast, and (b) past values observed for that particular time series. We consider that this information provides the user a good insight into whether the forecast can be trusted and understand the behavior of the underlying model.
- **API:** a standard Application Programming Interface (API) endpoint can be used to serve the user as a front-facing interface, masking the structure, complexity, and deployment configuration of each of the modules mentioned above.

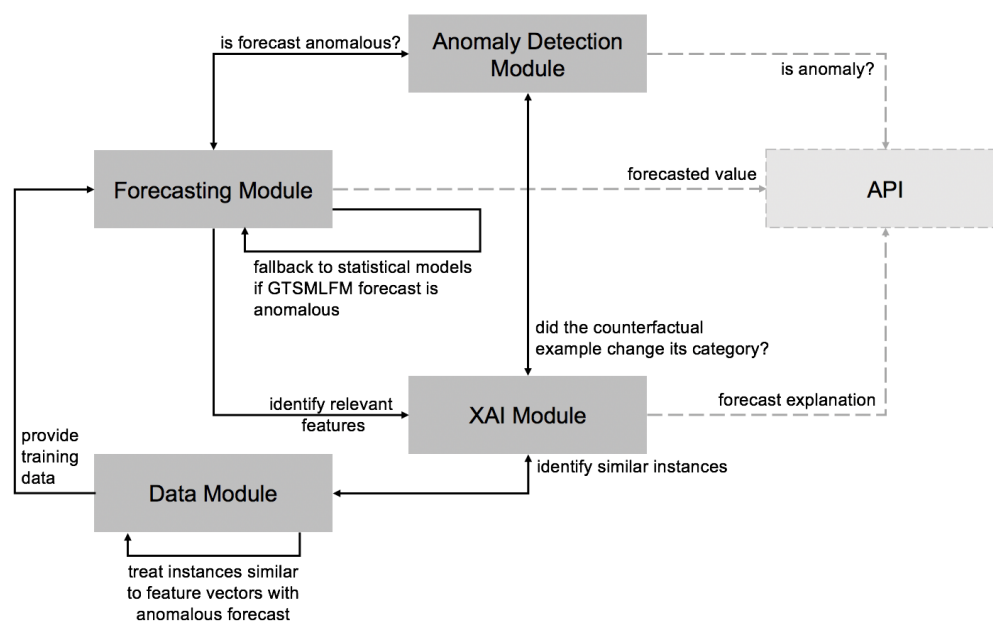


Figure 1. Architecture overview.

The architecture was implemented in a prototype application for the purpose of this paper, and not deployed into production. The backend was built using the Python 3.6 language, and leveraging multiple open-source data processing and machine learning libraries (Numpy [112], Pandas [113], Scikit-Learn [114], Scipy [115], Lime [75], Pyod [116], LightGBM [117]), and exposing relevant data through an HTTP REST API, using the Flask framework [118]. To build the frontend, we used HTML to layout the core of the dashboard, the Bootstrap framework to embellish the user interface, Plotly-js [119] to draw the plot, and jQuery [120] to interface with the HTTP REST API, retrieving the data and serving the content to the web page.

The interaction between the modules embodies the methodology we followed to enhance the GTSMLFM performance iteratively and provide an insight to the user regarding when and why does a GTSMLFM model provides an adequate forecast or not. We detail the methodology in Section 4.

4. Methodology

In this section, we describe how we leveraged the information provided by a prototype application built following the architecture described in Section 3, to enhance a GTSMLFM. To that end, we developed an iterative methodology inspired by work done by several authors, and summarized in Figure 2. Scientific literature on GTSMLFMs agrees that pooling similar time series does provide an advantage over local models [3,4,13,21], while pooling and time series similarity criteria remain an arbitrary choice [15]. Furthermore, some research indicates that GTSMLFMs can be trained over disparate time series and still obtain good forecasting results. We propose training an initial model over all the time series and measuring its performance. We then identify anomalous forecasted values. We consider that these values are a consequence of a subset of train data instances with a similar feature vector but different target values. To identify such instances, we first compute the feature relevance for a particular forecast. Given the N most relevant features to that forecast, we search for similar instances in the train set computing the cosine distance across the feature vectors, considering only the subset of the aforementioned N features. To avoid distortions due to different feature magnitudes, we scaled the features between zero and one. To decide which train instances to remove, we set an arbitrary similarity threshold. It is important to consider that the GTSMLFM performance can be affected by train data instances that lead to learning inaccurate forecasting patterns and

the amount of data available, which can eventually lead to better predictions. Thus, setting the right threshold requires a compromise between both factors and can be subject to trial and error. To conclude an iteration, we assess the GTSMLFM quality. In particular, we decided to measure it through three metrics (see Section 6): (i) mean absolute scaled error (MASE) [10], (ii) the number of outliers detected in the test set, and (iii) the number of train instances removed from the dataset to train that particular GTSMLFM. Following the Equation (1) [28], we argue that, while local statistical models do not match the overall performance of a good GTSMLFM, they can sometimes provide a better prediction when the GTSMLFM provides an anomalous forecast. To retain the scaling advantages, such as ease of deployment, and avoid dedicating human resources to developing and maintaining such models, simple heuristics such as the naïve forecast, simple moving average, or exponential smoothing can be used.

$$Data = Global Model + Local Models + Noise \quad (1)$$

The equation represents that local models can make better forecasts when the global model fails to predict reasonably. The equation was reproduced from [28].

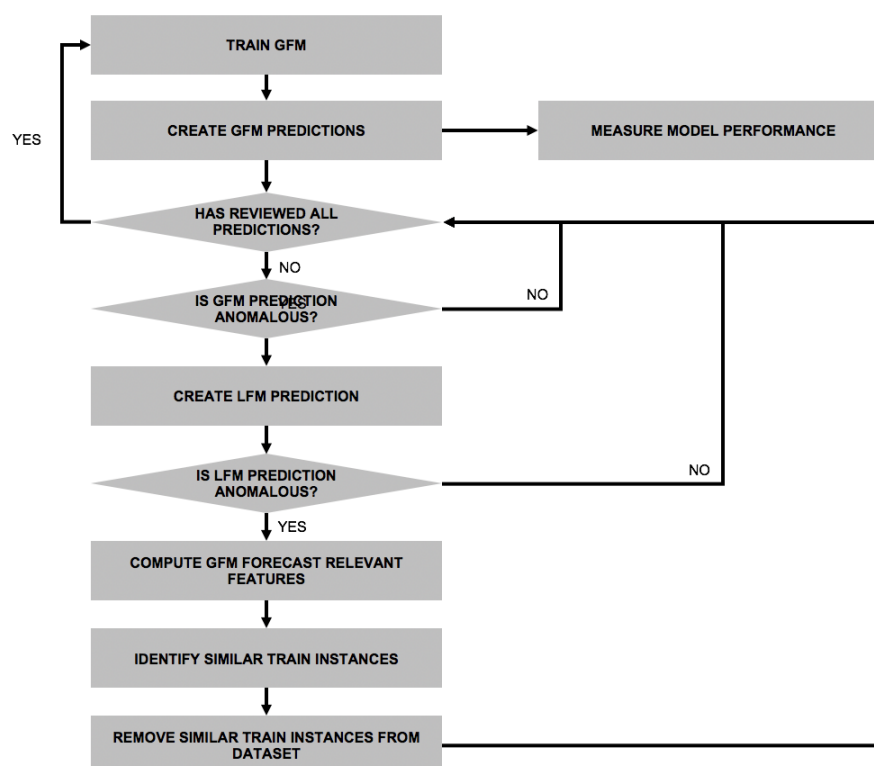


Figure 2. Fluxogram detailing the methodology that we applied to identify anomalous forecasts, most similar instances in the train set, and their treatment to enhance the performance of future GFM.

Once we finalize an iteration, we start a new one by retraining the GTSMLFM on the new dataset, following the steps mentioned above. Iterative model retraining is based on the RemOve And Retrain (ROAR) method [121], developed to identify features relevance measuring the model performance change between iterations when a feature is nullified. In our case, we do not measure the impact of the features but of a subset of training instances when removed from the dataset. Furthermore, inspiration to look into training instances to understand their impact on a given forecast was obtained from [122], who did so using influence functions.

5. Case Study

In this research, we considered two open datasets that are widely used in time series forecasting research: M4 competition dataset (M4CD) [5], and the Kaggle Wikipedia Web Traffic forecasting competition dataset (KWWTFCD) [123]. To ensure that the performance of the global models does not depend on the knowledge of a particular domain or characteristics of the time series data, we defined the same set of generic features for both cases. In this section, we describe the characteristics of each dataset, provide details on how we sampled instances from them, and the preprocessing steps we performed before training the GTSMLFM.

The M4CD comprises 100,000 time series selected from the ForeDeCk database, corresponding to multiple business domains, such as industries, government, transport, household, and natural resources. While the dataset includes time series at different frequencies (from hourly to yearly frequency), we focused on those provided monthly, which account for 48,000 time series. When analyzing their metadata and the actual time series length, we found some discrepancies. We clarified them with the authors from [124], who researched how representative it is of the reality. In private correspondence, the authors confirmed that such discrepancies existed, attributing them to the original public sources from which they were obtained. Since such time series represented only a tiny proportion of the total dataset, we ignored them.

The KWWTFCD was provided by Google and introduced in a Kaggle competition in 2017. The dataset comprises time series accounting for the daily views of approximately 145,000 Wikipedia articles, starting from 1 July 2015, until 11 September 2017. The dataset distinguishes between zero and missing values.

We randomly sampled 2000 time series for each dataset and selected only a subsequence to minimize the number of missing values (first 45 values for M4CD and first 56 values for the KWWTFCD). We considered 2000 time series as a big enough sample to evaluate diverse time series and provide meaningful results, while avoiding processing the whole dataset. The need for sampling is grounded in the fact that a feature vector is created per time step for each time series, requiring extensive computational resources and time to train the models.

When designing the experiment, we considered that a global model is more likely to produce anomalous forecasts if trained over a dataset that contains two types of time series: (i) the ones whose values remain in the same order of magnitude, and (ii) the ones whose values comprehend different orders of magnitude (TSDM). We thus analyzed the proportion of time series with such properties in each dataset and ensured that the sampling respected those proportions. The new datasets comprised 903 of type (i) and 1097 of type (ii) time series for the M4CD, and 512 type (i) and 1488 type (ii) time series for the KWWTFCD. We provide further details on the original and reduced dataset in Table 1, and describe the target values distribution in Figure 3.

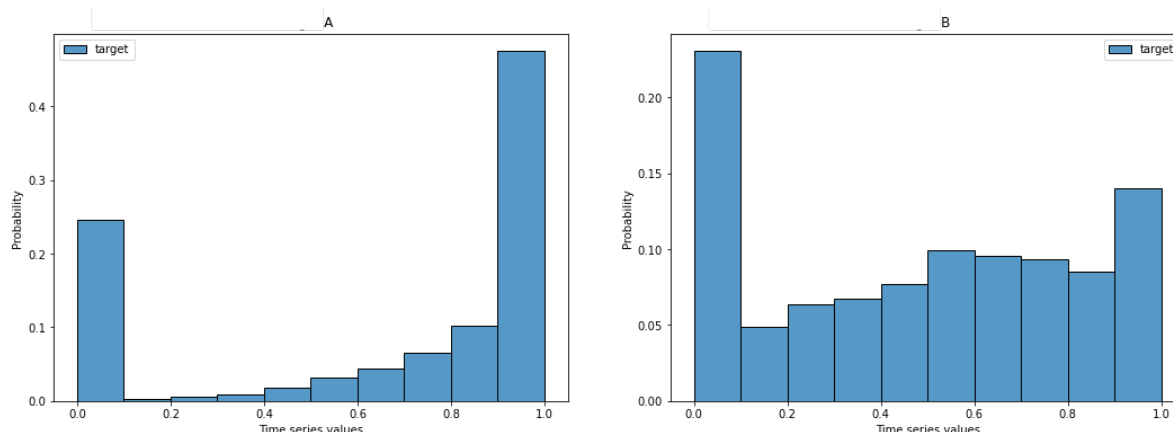


Figure 3. Target values distribution. (A) describes values for M4CD, while (B) describes values for KWWTFCD.

Table 1. Descriptive data for the M4CD and KWWTFCD datasets. TSDM is used as an abbreviation for time series with values of different orders of magnitude, while TS abbreviates time series.

Dataset	Original		Reduced Dataset						Test Window
	# TS	% TSDM	# TS	# TSDM	TS Datapoints	# Instances	Mean (Target)	Std (Target)	
M4CD	47,983	55	2000	1097	45	90,000	0.6598	0.4036	12
KWWTFCD	145,063	74	2000	1488	56	112,000	0.4782	0.3442	14

To describe the time series, we created a set of features, presented in Table 2. We considered three types of features: (i) the features that describe the values observed for a given time series (e.g., minimum, mean, median values, along with the standard deviation), (ii) the features that describe the time series shape (e.g., skew, kurtosis, the number of peaks we observed, and the number of values above the mean), and (iii) the features that describe the context close to the forecasted value (e.g., last observed value—which can be used as a naïve forecast). We compute thirty-three features for each dataset. For the global models, we do not perform feature selection, given that in all cases, we observe the number of features satisfies the Equation (2), as suggested in [125]. For the local machine learning models, we perform feature selection selecting top K features based on their mutual information [126].

$$k \leq \sqrt{N} \quad (2)$$

k represents the maximum number of features used to train a model to avoid overfitting. N represents the number of data instances available in the train set. The equation is based on research done by [125].

Several kinds of preprocessing have been tried for GTSMLFMs in the scientific literature. Smyl [127] applied on the fly preprocessing to remove level and seasonality components. Rabanser et al. [128] found that binning proved to be useful in almost all the cases they analyzed. Duncan et al. [13] understands that differences in magnitudes and variances can be removed, either by standardizing the time series or using dimensionless dependent variables. Hewamalage et al. [129] describe applying a local normalization, deseasonalization, and log transformation to the features. Finally, Salinas et al. [27] describes scaling the features by their average value. In our case, we opted to scale the values of a feature vector between zero and one by dividing the entries by the maximum value observed in the period of interest. Such scaling forced most feature values to a standard interval between zero and one, helping the model learn similar patterns, regardless of the original time series magnitudes. To further ease the learning process to the model, we scaled the target values too.

Table 2. Description of features we created for each dataset. We used $n = 3, 5, 7, 12$.

Feature	Data Type	Description
min_ n	Double	Minimum value in rolling window of last n observations.
mean_ n	Double	Mean of values in rolling window of last n observations.
std_ n	Double	Standard deviation for values in rolling window of last n observations.
median_ n	Double	Median value in rolling window of last n observations.
skew_ n	Double	Skew value in rolling window of last n observations.
kurt_ n	Double	Kurtosis value in rolling window of last n observations.
peaks_ n	Integer	Count the number of peaks for a rolling window of last n observations.
above_mean_ n	Double	Count the number of datapoints above the mean, for a rolling window of last n observations. The value is normalized by windows length.
$n - 1$	Double	Last observed target value, normalized by maximum value observed in time window $n = 12$.

Outlier detection was performed considering a time series comprised of the original time series data points up to the start of the forecasting horizon and then adding the forecasted value, since we only performed one step ahead forecasts (see Figure 4). Doing so enables us to use multiple anomaly detection approaches to evaluate aspects under which a particular value can be considered an outlier within the time series.

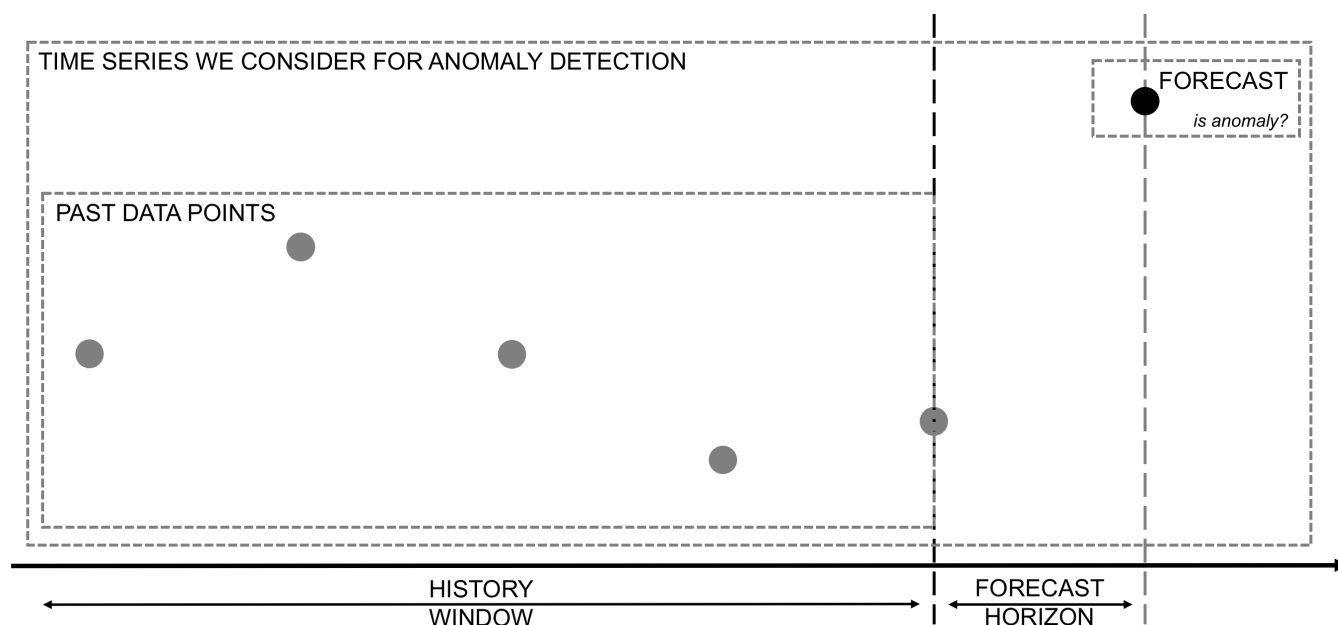


Figure 4. Diagram describing the data that we provide to build the time series context for anomaly detection.

6. Experiments, Results, and Analysis

To validate the architecture and methodology described in Sections 3 and 4, we conducted a series of experiments (summarized in Table 3) comparing four models (described in Table 4) on the reduced version of two open datasets presented in Section 5. We built our local and global forecasting model with a gradient boosted machine regressor (GBMR) [117], considering that all but one of the top five solutions of the M5 forecasting competition were based on it [6]. We configured the GBMR model to be deterministic, have a maximum depth of five, and at most a hundred estimators. All GBMR models were instantiated with the same random seed (using the value 744) and an L2 loss. For every time series, we also created two simple local statistical models: a simple moving average, based on the last three points of data, and a naïve forecast, providing a forecast based on the last observed value. On top of the models we described, we built an additional model, which considered the predictions issued by the GM(GBMR) model, and used the naïve model to issue fallback predictions when detecting an anomalous prediction was given by the global model. We opted to use a naïve model as fallback, since it is considered a baseline model in most of timeseries literature. In addition, the naïve model is cheap to compute, and requires no previous knowledge regarding specific timeseries, making it a more scalable solution. We evaluated the models performing a nested cross-validation [130] over the last twelve data points (a year of data monthly) for M4CD and the last fourteen data points (two weeks of data daily) for KWWTFCD.

To evaluate the experiments, we considered three groups of metrics. First, we used MASE [10] to measure the model's performance. The MASE metric provides a magnitude agnostic estimate of the forecasting precision achieved by the model, comparing the model's performance against a naïve forecast. We measured the MASE values for the MA(3), GM(GBMR), GM(GBMR)+naïve, and LM(GBMR) models. Second, we assessed the anomalies detected in the test set. We measured (i) how many GM(GBMR) predictions were considered anomalous when the target values were not (GM(GBMR) vs. Target

discrepancy column in Table 5); (ii) how many target values were identified as anomalous (Target column in Table 5); (iii) how many GM(GBMR) predictions were identified as anomalous, and their overlap regarding the GM(GBMR) column under Anomalies in test in Table 5); and (iv) how many GM(GBMR)+naïve predictions were identified as anomalous, and their overlap regarding the GM(GBMR)+naïve column under Anomalies in test in Table 5). Finally, we measured how many instances were erased from the train set based on their similarity to feature vectors producing anomalous forecasts. Through these sets of metrics, we could assess the model's performance when changing the experimental conditions, understand if the detected outliers correspond to true or false positives, and how successfully does the global model behave in such cases, and if there is any correlation between the number of train instances removed and the global model's performance.

Table 3. Description of the experiments that we performed.

Experiment	Description
Experiment 1	Compare the performance of GFM and LFM.
Experiment 2	Remove instances similar to the cases where the forecast was considered anomalous. The similarity of the train instances is measured on relevant features of the forecast feature vector.
Experiment 3	Remove only instances similar to the cases where the fallback was considered anomalous. The similarity of the train instances is measured on relevant features of the forecast feature vector.
Experiment 4	Remove instances similar to the cases where the forecast was considered anomalous. The similarity of the train instances is measured on relevant features of the forecast feature vector and target value.
Experiment 5	Remove only instances similar to the cases where the fallback was considered anomalous. The similarity of the train instances is measured on relevant features of the forecast feature vector and target value.

Table 4. Description of the models we evaluated through the experiments that we performed.

Model name	Description
MA(3)	Moving average over last three time steps.
naïve	Last time step actual is used as the forecast value.
GM(GBMR)	Global model built with GBMR.
LM(GBMR)	Local model built with GBMR.
GM(GBMR)+naïve	Forecasts are issued from GM(GBMR), except when the forecasted value is considered anomalous. In such cases, it falls back to a naïve forecast.

Table 5. Results obtained for the experiments. For columns GM(GBMR) and GM(GBMR)+naïve under anomalies in test, we use the following convention to present the results: A/B (C), where A denotes the number of datapoints considered anomalies both, in the prediction and effective target value; B denotes the number of forecasts issued by the model that were considered anomalous; and C provides the ratio between A and B . ETI is used as an abbreviation for erased train instances. TD is used as an abbreviation for target discrepancy.

Experiment	Dataset	MASE				GM(GBMR) vs. TD	Anomalies in Test			ETI	
		MA(3)	GM(GBMR)	GM(GBMR)+naïve	LM(GBMR)		Target	GM(GBMR)	GM(GBMR)+naïve	# ETI	Ratio ETI
Experiment 1	M4CD	1.8473	0.7387	0.7644	4.3261	304	4337	1553/1857 (0.84)	1555/2249 (0.69)	NA	NA
	KWWTFCD	1.4871	0.6828	0.7216	1.0360	407	2750	991/1398 (0.71)	924/1916 (0.48)	NA	NA
Experiment 2	M4CD	1.8473	0.7392	3.0213	4.3261	290	4337	1567/1857 (0.84)	1555/2249 (0.69)	1857	0.0281
	KWWTFCD	1.4871	0.6832	1.5915	1.0360	374	2714	919/1293 (0.71)	869/1808 (0.48)	1293	0.0154
Experiment 3	M4CD	1.8473	0.7361	3.0228	4.3261	283	4337	1588/1871 (0.85)	1555/2249 (0.69)	1365	0.0207
	KWWTFCD	1.4871	0.6829	1.5912	1.0360	375	2714	925/1300 (0.71)	869/1808 (0.48)	813	0.0097
Experiment 4	M4CD	1.8473	0.7392	3.0213	4.3261	290	4337	1567/1857 (0.84)	1555/2249 (0.69)	1857	0.0281
	KWWTFCD	1.4871	0.6832	1.5915	1.0360	374	2714	919/1293 (0.71)	869/1808 (0.48)	1293	0.0154
Experiment 5	M4CD	1.8473	0.7361	3.0229	4.3261	283	4337	1588/1871 (0.85)	1555/2249 (0.69)	1365	0.0207
	KWWTFCD	1.4871	0.6828	1.5911	1.0360	374	2714	925/1299 (0.71)	869/1808 (0.48)	813	0.0097

6.1. Premise Validation (Experiment 1)

We devoted the first experiment to validate the premise of this work: that the global model outperformed the local ones (GM(GBMR) vs. MA(3) and LM(GBMR) in Experiment 1), and even the GM(GBMR)+naïve, showing that the fallback to the naïve forecast did not provide the expected performance improvements. This was confirmed

in the rest of the experiments. Our intuition behind the result is that fallback forecasts replacing the original prediction in false positives issued by the anomaly detector (for which the GM(GBMR) model would have a better prediction) degrade the overall performance and lead to suboptimal results. We then performed four additional experiments to understand if removing instances from the train set similar to the reference vector generating an anomalous forecast could improve the performance of the global model.

It must be noted that the anomaly detection is run on forecasted values. Thus, while anomalies can be subjective, we consider that objective criteria can be established to identify suspicious data points (or forecasts) that meet those criteria, and are most likely to be considered outliers (or anomalous).

6.2. Dealing with Anomalous GTSMLFM Forecasts (Experiment 2 and Experiment 4)

Once the GTSMLFM was trained, we proceeded to identify outliers produced by the GTSMLFM using an ensemble of time series anomaly detectors (COPOD [131], ABOD [132], and KNN [133]), and considering a given forecast anomalous only when all the detectors agreed it should be considered as such. Next, we used the LIME [75] to compute features relevance for each anomalous forecast.

The most similar instances between the feature vector generating an anomalous prediction and data available in the training set were obtained, computing the cosine distance over a subset of the five most relevant features to each prediction. In Experiment 2 and Experiment 4, we computed such similarity against the data instances in the train set for all GTSMLFM anomalous forecasts. For Experiment 2, we considered a vector built with the most relevant features to a specific forecast, while in Experiment 4, we enlarged that vector with the target value for the train set instances and the predicted value for the reference vector. The intuition behind including such value in the vector is that instances most likely influence the model with similar input values and teach a target value close to the observed prediction.

While we experimented with multiple similarity thresholds, we finally adopted a threshold of 0.999999, for which, in most cases, just a handful of train instances were identified as similar to the feature vectors producing anomalous forecasts. However, due to some exceptional cases resulting in a relatively high number of similar instances despite the tight threshold, we decided to provide an additional bounding criterion, collecting, at most, ten top train instances for each anomalous forecast. This limit was established considering the relationship between the size of the test set, the number of anomalies, and an upper bound to the number of instances to be removed. While such a procedure helped us identify and remove instances that distorted the GTSMLFM's training, future work will develop a better procedure to determine the parameters and identify such instances.

When analyzing the results, we observed, in both cases, that the model's performance decayed (MASE measurements for GM(GBMR) and GM(GBMR)+naïve), while the discrepancy between GM(GBMR) predictions and target values that were considered anomalous reduced, indicating a better adjustment to the real data behavior for unexpected values. While the MASE improvement was slight, it must be noted that such improvement was consistent and thus is improbable that it could be attributed to models' variance. Furthermore, we expect that the forecast improvements are minor in terms of MASE: reducing the number of anomalies across a test set of 24,000 records for dataset M4CD, and 28,000 for dataset KWWTFCD, should not have a substantial impact on the overall MASE. While the MASE is indicative of the overall models' forecast adjustment, the improvement regarding anomalous forecasts should be considered in terms of a reduction of the target discrepancy (the model better adjusts to the observed target values in the test set). We observe that the number of anomalous forecasts is reduced in both cases, showing that the new model better adjusts to the time series behavior.

6.3. Dealing with Anomalous Fallback Forecasts (Experiment 3 and Experiment 5)

Assuming that more data is beneficial to global model performance and that the naïve forecast can provide a reasonable estimate when the GM(GBMR) issues an anomalous forecast, we conducted two additional experiments (Experiment 3 and Experiment 5), removing only the train instances that were similar to feature vectors for which both, the GM(GBMR) and the naïve forecast provided an anomalous forecast. For Experiment 5, we enlarged the reference vector with the predicted value (or the target value for the vectors in the train set). In both experiments, we observed an improved MASE score regarding Experiment 2 and Experiment 4 and a reduced discrepancy between values considered anomalous for GM(GBMR) predictions and target values. In particular, we found that the best results for all metrics in both datasets were obtained for Experiment 5, reducing outliers discrepancy regarding the original model for at least seven percent in both datasets.

From the results that we obtained, we consider the vector structure used to measure the similarity between the instances (using the most relevant features either with the target (or predicted) value or not), did not influence the results. We confirmed that improving the global model performance is possible by removing particular data instances from the train set. When searching for such instances, it is crucial to ensure that the least possible data is removed. Since the M4CD dataset comprises a wide variety of time series and the same architecture and methodology were used to replicate the findings on the KWWTFCD dataset, we expect that they can be generalized to other domains. Moreover, though the research was applied to time series datasets, we consider that the findings can be ported to regression problems in general, due to how we formulated the forecasting problem.

6.4. Dashboard

To inform the user when and why a given forecast should be trusted, we built a dashboard (see Figure 5). The dashboard has six sections, each of which provides specific information to the user. At the top of the dashboard (Figure 5A), we provide information regarding the time series identifier, the dataset they belong to, the forecast date and value, if considered an outlier, and the most relevant features to that particular forecast. We then created a line plot of the time series, where the last value corresponds to the forecast (Figure 5B). We use a gray dashed vertical line (see the Figure 5(B2) arrow) to indicate the start of the forecasting horizon, and a red dashed vertical line (see the Figure 5(B3) arrow) to highlight the occurrence of an anomalous forecast. We provide a short explanation regarding the criteria applied to the forecast to determine if it is anomalous or not (Figure 5C).

Given that the machine learning models that we consider assume manually crafted features, insights regarding the time dimension can be conveyed only through the features and their metadata. Inspired by related work [91,93], we map features metadata regarding the time dimension and compute how strongly that the referenced time windows and points-in-time are from the most relevant features to a particular forecast overlay. Based on that information and the values of the time series data points within each range considered, we overlaid a heatmap to the time series line plot, showing how influential those data points are to the forecast. In particular, in Figure 5(B1), we show a heatmap with a heat scale ranging from yellow (low relevance) to red (high relevance). The relevance heatmap's area is determined by the time windows of the relevant features and the values of the data points that fall within the aforementioned time window.

In Figure 5D, we display a set of training instances that probably influenced the forecast (prototype local explanations), followed by two counterfactual examples: a set of instances that would result in a non-anomalous forecast (Figure 5E). We created our utility code to compute the counterfactual examples (see algorithm's pseudocode in Listing 1). To compute them, we restricted the values search to the most significant features identified by LIME and ensured that the values were plausible. To decide whether there was a significant change in the forecasted value when computing counterfactual examples, we created a feature vector with the mean values of all the features, except for the meaningful

ones identified through LIME, which received values of their own by ninety thousand values from a normal distribution. The normal distribution was parameterized with a mean equal to the mean of past time series values, and a standard deviation equal to three times the standard deviation measured in the past for that same time series. We then used the GTSMFLM to issue a forecast and the anomaly detector to determine if the proposed sample instance qualified as an outlier or not, and randomly selected a subset of them to show them to the user.

Listing 1: Algorithm used to compute counterfactual examples.

```
# X_train: dataset train instances
# feature_vector: feature vector used to issue the forecast
# relevant_features: list of relevant features to the given forecast
# model: forecasting model
# anomaly_detector: some anomaly detector
# n_samples: number of samples to draw from the Normal distribution
given X_train, feature_vector, relevant_features, model, anomaly_detector
synthetic_samples = new dictionary()

for each feature in relevant_features:
    feature_mean = mean(X_train[feature])
    feature_std = standard_deviation(X_train[feature])
    feature_perturbed_values = normal(feature_mean, 3*feature_std, n_samples)
    synthetic_samples[feature]=feature_perturbed_values

# create new dataset, merging data from non relevant features, and perturbed ones
new_dataset = create_dataset(X_train, synthetic_samples)

counterfactual_examples = new list()
for each feature_vector in new_dataset:
    y_pred = model.predict(feature_vector)
    if not anomaly_detector.is_anomaly(y_pred):
        counterfactual_examples.append(feature_vector)
return counterfactual_examples
```

The last dashboard square is devoted to providing a set of values expected to make a better forecast (Figure 5F). To obtain them, we drew ten thousand values following a normal distribution, with a mean equal to the mean of past time series values and a standard deviation equal to three times the standard deviation measured in the past for that same time series. The values were then filtered by performing the same anomaly detection procedure as for the GTSMFLM forecasts, keeping only those not considered anomalous. Figure 5F displays just a subset of them: the minimum, median, and maximum values, and additional four random samples drawn from them.

In this work, we used a particular selection of models and algorithms to create the forecasts, detect anomalies, and craft the explanations. However, given the architecture's modular structure, these can be replaced as black boxes, impacting the quality of the content displayed in the dashboard.



Figure 5. Dashboard screenshot. We highlight different areas devoted to explaining a given forecast, providing a context within the time series, indicating the most relevant features to the forecast, train instances that most likely influenced the forecast, counterfactual examples, and alternative values expected to make a good forecast.

7. Limitations and Improvement Opportunities

We identify two significant limitations of the research described above. It must first be noted that, while the methodology and dashboard that we developed are generic, they target global time series models constrained to machine learning algorithms whose input features are intelligible to the human. Furthermore, many of the explanations assume the features convey a particular meaning to the user—which is, in our view, a criterion that handcrafted features can only meet. Due to this limitation, we consider this approach is not suitable for deep learning models. In addition, given how the features are framed for such machine learning models, insights regarding the time dimension can be conveyed through the features and their metadata. Second, the current approach to selecting train data instances similar to the data instances creating the anomalous forecast requires some tuning to find an appropriate instances' similarity threshold and set proper constraints towards the maximum number of instances to consider for each case. While this could be replaced by an automatic procedure searching for the best similarity measure, threshold, and instances' limit, such a procedure would come at a high cost, since it would require computing the similarities across all instances for each run. Further research is required to understand better alternative procedures to identify similar train data instances that drive the forecast towards an anomalous value.

From experience that we obtained through the experiments and results described above, we identified improvement opportunities. Following research done by [122], finding the most influential instances in the train set for an anomalous forecast can be done using influence functions. Other possible enhancements would be to use a more stable model to estimate feature relevance for each prediction. In particular, we could replace LIME for DLIME, since LIME is not deterministic, and changes in feature ranking computations can

affect the instance selection based on the similarity to the detected anomalous forecasts. Another improvement can be made regarding the anomaly detection module. Reducing the number of false positives correlates to the number of instances removed from the train set. Furthermore, removing only instances similar to the ones where the global model and fallback provide anomalous forecasts has been shown to improve the performance of the model trained without such instances. Following this intuition, we can use our current anomaly detector for unsupervised anomalies labeling. By labeling instances as anomalous only when the current anomaly detector predicts the actual value is not anomalous when the forecast was, we can later train more precise supervised machine learning models to detect outliers.

8. Conclusions

In this work, we developed a modular architecture, a methodology, and a dashboard, that provide insights when a GTSMLFM forecast can be trusted or not and the reasons behind anomalous forecasts. The architecture, methodology, and dashboard support the development and engineering of GTSMLFMs, providing means to enhance their performance. We evaluated our approach through a series of experiments conducted on a reduced version of two publicly available datasets and measuring the performance improvements of the GTSMLFM when following the methodology described in Section 4. Our research confirmed that removing particular instances from the train set can lead to a better GTSMLFM performance and compared several approaches to achieve the best outcome.

As future work, we envision extending the current application to support explainable anomaly detection algorithms and include a semantic model to enrich the explainability of any anomaly detection model. Such a semantic model can provide additional insights based on domain knowledge regarding the inner workings of the anomaly detection model and the data of a particular time series and forecast. Finally, we would like to explore different policies to deal with problematic train instances. In particular, we are interested in studying if replacing the values of a subset of features of interest with some imputation criteria can be an effective alternative to removing such data instances. Such a technique would retain the advantages of keeping all the data to train a global model, while removing patterns that create outlier forecasts. Moreover, to avoid losing valuable information in the noisy instances, we could leverage generative adversarial sampling to enrich the dataset with synthetic train instances that resemble noisy ones. Such enrichment could help the model better learn the decision boundaries, increasing the overall model's performance.

Author Contributions: Conceptualization, J.R.; methodology, J.R.; software, J.R. and E.T.; validation, J.R. and E.T.; formal analysis, J.R. and K.K.; investigation, J.R.; resources, J.R., E.T., K.K. and B.F.; data curation, J.R. and E.T.; writing—original draft preparation, J.R.; writing—review and editing, J.R., K.K., B.F. and D.M.; visualization, J.R.; supervision, K.K., B.F. and D.M.; project administration, B.F. and D.M.; funding acquisition, B.F. and D.M. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Slovenian Research Agency and the European Union's Horizon 2020 program project STAR under grant agreement number H2020-956573.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AF	Anomalous Forecast
COF	Connectivity based Outlier Factor
ETI	Erased Train Instances
GBMR	Gradient Boosted Machine Regressor
GFM	Global Forecasting Model
GTSMLFM	Global Time Series Machine Learning Forecasting Model
INFLO	Influenced Outlierness
kNN	n-Nearest Neighbor
KWWTFC	Kaggle Wikipedia Web Traffic Forecasting Competition Dataset
LFM	Local Forecasting Model
LOF	Local Outlier Factor
M4CD	M4 Competition Dataset
MASE	Mean Absolute Scaled Error
ODIN	Outlier Detection using Indegree Number
ROAR	RemOve And Retrain
TD	Target Discrepancy
TS	Time Series
TSDM	Time Series with Different Magnitude values
XAI	Explainable Artificial Intelligence

References

- Sen, R.; Yu, H.F.; Dhillon, I. Think globally, act locally: A deep neural network approach to high-dimensional time series forecasting. *arXiv* **2019**, arXiv:1905.03806.
- Bontempi, G.; Taieb, S.B.; Le Borgne, Y.A. Machine learning strategies for time series forecasting. In *European Business Intelligence Summer School*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 62–77.
- Hewamalage, H.; Bergmeir, C.; Bandara, K. Global models for time series forecasting: A simulation study. *arXiv* **2020**, arXiv:2012.12485.
- Petropoulos, F.; Apiletti, D.; Assimakopoulos, V.; Babai, M.Z.; Barrow, D.K.; Bergmeir, C.; Bessa, R.J.; Boylan, J.E.; Browell, J.; Carnevale, C.; et al. Forecasting: Theory and practice. *arXiv* **2020**, arXiv:2012.03854.
- Makridakis, S.; Spiliotis, E.; Assimakopoulos, V. The M4 Competition: 100,000 time series and 61 forecasting methods. *Int. J. Forecast.* **2020**, *36*, 54–74. [\[CrossRef\]](#)
- Makridakis, S.; Spiliotis, E.; Assimakopoulos, V. The M5 accuracy competition: Results, findings and conclusions. *Int. J. Forecast.* **2020**, in press. [\[CrossRef\]](#)
- Rojat, T.; Puget, R.; Filliat, D.; Del Ser, J.; Gelin, R.; Díaz-Rodríguez, N. Explainable Artificial Intelligence (XAI) on TimeSeries Data: A Survey. *arXiv* **2021**, arXiv:2104.00950.
- Montero-Manso, P.; Hyndman, R.J. Principles and algorithms for forecasting groups of time series: Locality and globality. *Int. J. Forecast.* **2021**, *37*, 1632–1653. [\[CrossRef\]](#)
- Henin, C.; Le Métayer, D. A multi-layered approach for tailored black-box explanations. In Proceedings of the ICPR 2020—Workshop Explainable Deep Learning, Virtual, 10–15 January 2021.
- Hyndman, R.J. Another look at forecast-accuracy metrics for intermittent demand. *Foresight* **2006**, *4*, 43–46.
- Januschowski, T.; Gasthaus, J.; Wang, Y.; Salinas, D.; Flunkert, V.; Bohlke-Schneider, M.; Callot, L. Criteria for classifying forecasting methods. *Int. J. Forecast.* **2020**, *36*, 167–177. [\[CrossRef\]](#)
- Zellner, A. An efficient method of estimating seemingly unrelated regressions and tests for aggregation bias. *J. Am. Stat. Assoc.* **1962**, *57*, 348–368. [\[CrossRef\]](#)
- Duncan, G.T.; Gorr, W.L.; Szczypula, J. Forecasting analogous time series. In *Principles of Forecasting*; Springer: Boston, MA, USA, 2001; pp. 195–213.
- Bandara, K.; Bergmeir, C.; Smyl, S. Forecasting across time series databases using recurrent neural networks on groups of similar series: A clustering approach. *Expert Syst. Appl.* **2020**, *140*, 112896. [\[CrossRef\]](#)
- Godahehwa, R.; Bandara, K.; Webb, G.I.; Smyl, S.; Bergmeir, C. Ensembles of localised models for time series forecasting. *arXiv* **2020**, arXiv:2012.15059.
- Mori, H.; Yuihara, A. Deterministic annealing clustering for ANN-based short-term load forecasting. *IEEE Trans. Power Syst.* **2001**, *16*, 545–551. [\[CrossRef\]](#)
- Manojlović, I.; Švenda, G.; Erdeljan, A.; Gavrić, M. Time series grouping algorithm for load pattern recognition. *Comput. Ind.* **2019**, *111*, 140–147. [\[CrossRef\]](#)
- Marinazzo, D.; Liao, W.; Pellicoro, M.; Stramaglia, S. Grouping time series by pairwise measures of redundancy. *Phys. Lett. A* **2010**, *374*, 4040–4044. [\[CrossRef\]](#)
- Romanov, A.; Perfilieva, I.; Yarushkina, N. Time series grouping on the basis of F 1-transform. In Proceedings of the 2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Beijing, China, 6–11 July 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 517–521.

20. Li, Y.; Liu, R.W.; Liu, Z.; Liu, J. Similarity grouping-guided neural network modeling for maritime time series prediction. *IEEE Access* **2019**, *7*, 72647–72659. [\[CrossRef\]](#)
21. Fildes, R.; Beard, C. Forecasting systems for production and inventory control. *Int. J. Oper. Prod. Manag.* **1992**. [\[CrossRef\]](#)
22. Verdes, P.; Granitto, P.; Navone, H.; Ceccatto, H. Forecasting chaotic time series: Global vs. local methods. *Novel Intell. Autom. Control. Syst.* **1998**, *1*, 129–145.
23. Långkvist, M.; Karlsson, L.; Loutfi, A. A review of unsupervised feature learning and deep learning for time-series modeling. *Pattern Recognit. Lett.* **2014**, *42*, 11–24. [\[CrossRef\]](#)
24. Wen, R.; Torkkola, K.; Narayanaswamy, B.; Madeka, D. A multi-horizon quantile recurrent forecaster. *arXiv* **2017**, arXiv:1711.11053.
25. Laptev, N.; Yosinski, J.; Li, L.E.; Smyl, S. Time-series extreme event forecasting with neural networks at uber. In Proceedings of the International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; Volume 34, pp. 1–5.
26. Rangapuram, S.S.; Seeger, M.W.; Gasthaus, J.; Stella, L.; Wang, Y.; Januschowski, T. Deep state space models for time series forecasting. *Adv. Neural Inf. Process. Syst.* **2018**, *31*, 7785–7794.
27. Salinas, D.; Flunkert, V.; Gasthaus, J.; Januschowski, T. DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *Int. J. Forecast.* **2020**, *36*, 1181–1191. [\[CrossRef\]](#)
28. Burkart, N.; Huber, M.F. A survey on the explainability of supervised machine learning. *J. Artif. Intell. Res.* **2021**, *70*, 245–317. [\[CrossRef\]](#)
29. Blázquez-García, A.; Conde, A.; Mori, U.; Lozano, J.A. A Review on outlier/Anomaly Detection in Time Series Data. *ACM Comput. Surv. (CSUR)* **2021**, *54*, 1–33. [\[CrossRef\]](#)
30. Hawkins, D.M. *Identification of Outliers*; Chapman and Hall: London, UK, 1980; Volume 11.
31. Munir, M.; Siddiqui, S.A.; Dengel, A.; Ahmed, S. DeepAnT: A deep learning approach for unsupervised anomaly detection in time series. *IEEE Access* **2018**, *7*, 1991–2005. [\[CrossRef\]](#)
32. Fox, A.J. Outliers in time series. *J. R. Stat. Soc. Ser. B (Methodol.)* **1972**, *34*, 350–363. [\[CrossRef\]](#)
33. Cook, A.A.; Mısırlı, G.; Fan, Z. Anomaly detection for IoT time-series data: A survey. *IEEE Internet Things J.* **2019**, *7*, 6481–6494. [\[CrossRef\]](#)
34. Latecki, L.J.; Lazarevic, A.; Pokrajac, D. Outlier detection with kernel density functions. In Proceedings of the International Workshop on Machine Learning and Data Mining in Pattern Recognition, Leipzig, Germany, 18–20 July 2007; Springer: Berlin/Heidelberg, Germany, 2007; pp. 61–75.
35. Sheng, B.; Li, Q.; Mao, W.; Jin, W. Outlier detection in sensor networks. In Proceedings of the 8th ACM International Symposium on Mobile ad Hoc Networking and Computing, Montréal, QC, Canada, 9–14 September 2007; pp. 219–228.
36. Goldstein, M.; Dengel, A. Histogram-based outlier score (hbos): A fast unsupervised anomaly detection algorithm. In *KI-2012: Poster and Demo Track*; Citeseer: Princeton, NJ, USA, 2012; pp. 59–63.
37. Sathe, S.; Aggarwal, C.C. Subspace histograms for outlier detection in linear time. *Knowl. Inf. Syst.* **2018**, *56*, 691–715. [\[CrossRef\]](#)
38. Smiti, A. A critical overview of outlier detection methods. *Comput. Sci. Rev.* **2020**, *38*, 100306. [\[CrossRef\]](#)
39. Wei, L.; Kumar, N.; Lolla, V.N.; Keogh, E.J.; Lonardi, S.; Ratanamahatana, C.A. Assumption-Free Anomaly Detection in Time Series. In Proceedings of the 17th International Conference on Scientific and Statistical Database Management, SSDBM 2005, Santa Barbara, CA, USA, 27–29 June 2005; Volume 5, pp. 237–242.
40. Kumar, N.; Lolla, V.N.; Keogh, E.; Lonardi, S.; Ratanamahatana, C.A.; Wei, L. Time-series bitmaps: A practical visualization tool for working with large time series databases. In Proceedings of the 2005 SIAM International Conference on Data Mining, SIAM, Newport Beach, CA, USA, 21–23 April 2005; pp. 531–535.
41. Ahmed, T. Online anomaly detection using KDE. In Proceedings of the GLOBECOM 2009—2009 IEEE Global Telecommunications Conference, Honolulu, HI, USA, 30 November–4 December 2009; IEEE: Piscataway, NJ, USA, 2009; pp. 1–8.
42. Kim, J.; Scott, C.D. Robust kernel density estimation. *J. Mach. Learn. Res.* **2012**, *13*, 2529–2565.
43. Zhang, L.; Lin, J.; Karim, R. Adaptive kernel density-based anomaly detection for nonlinear systems. *Knowl.-Based Syst.* **2018**, *139*, 50–63. [\[CrossRef\]](#)
44. Laurikkala, J.; Juhola, M.; Kentala, E.; Lavrac, N.; Miksch, S.; Kavsek, B. Informal identification of outliers in medical data. In Proceedings of the Fifth International Workshop on Intelligent Data Analysis in Medicine and Pharmacology, Berlin, Germany, 22 August 2000; Citeseer: Princeton, NJ, USA, 2000; Volume 1, pp. 20–24.
45. Pang, J.; Liu, D.; Liao, H.; Peng, Y.; Peng, X. Anomaly detection based on data stream monitoring and prediction with improved Gaussian process regression algorithm. In Proceedings of the 2014 International Conference on Prognostics and Health Management, Fort Worth, TX, USA, 29 September–2 October 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 1–7.
46. Pandit, R.K.; Infield, D. SCADA-based wind turbine anomaly detection using Gaussian process models for wind turbine condition monitoring purposes. *IET Renew. Power Gener.* **2018**, *12*, 1249–1255. [\[CrossRef\]](#)
47. Rousseeuw, P.J. Least median of squares regression. *J. Am. Stat. Assoc.* **1984**, *79*, 871–880. [\[CrossRef\]](#)
48. Rousseeuw, P.J.; Van Driessen, K. Computing LTS regression for large data sets. *Data Min. Knowl. Discov.* **2006**, *12*, 29–45. [\[CrossRef\]](#)
49. Salibian-Barrera, M.; Yohai, V.J. A fast algorithm for S-regression estimates. *J. Comput. Graph. Stat.* **2006**, *15*, 414–427. [\[CrossRef\]](#)
50. Ning, J.; Chen, L.; Zhou, C.; Wen, Y. Parameter k search strategy in outlier detection. *Pattern Recognit. Lett.* **2018**, *112*, 56–62. [\[CrossRef\]](#)

51. Breunig, M.M.; Kriegel, H.P.; Ng, R.T.; Sander, J. LOF: identifying density-based local outliers. In Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, Dallas, TX, USA, 16–18 May 2000; pp. 93–104.
52. Hautamaki, V.; Karkkainen, I.; Franti, P. Outlier detection using k-nearest neighbour graph. In Proceedings of the 17th International Conference on Pattern Recognition—ICPR 2004, Cambridge, UK, 26 August 2004; Volume 3, pp. 430–433.
53. Papadimitriou, S.; Kitagawa, H.; Gibbons, P.B.; Faloutsos, C. Loci: Fast outlier detection using the local correlation integral. In Proceedings 19th International Conference on Data Engineering (Cat. No. 03CH37405), Bangalore, India, 5–8 March 2003; IEEE: Bangalore, India, 2003; pp. 315–326.
54. Tang, J.; Chen, Z.; Fu, A.W.C.; Cheung, D.W. Enhancing effectiveness of outlier detections for low density patterns. In Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining, Taipei, Taiwan, 6–8 May 2002; Springer-Verlag: Berlin/Heidelberg, Germany, 2002; pp. 535–548.
55. Jin, W.; Tung, A.K.; Han, J.; Wang, W. Ranking outliers using symmetric neighborhood relationship. In Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining, Singapore, 9–12 April 2006; Springer: Berlin/Heidelberg, Germany, 2006; pp. 577–593.
56. Ma, Y.; Zhao, X. POD: A Parallel Outlier Detection Algorithm Using Weighted kNN. *IEEE Access* **2021**, *9*, 81765–81777. [[CrossRef](#)]
57. Trittenbach, H.; Englhardt, A.; Böhm, K. An overview and a benchmark of active learning for outlier detection with one-class classifiers. *Expert Syst. Appl.* **2020**, *168*, 114372. [[CrossRef](#)]
58. Schölkopf, B.; Platt, J.C.; Shawe-Taylor, J.; Smola, A.J.; Williamson, R.C. Estimating the support of a high-dimensional distribution. *Neural Comput.* **2001**, *13*, 1443–1471. [[CrossRef](#)] [[PubMed](#)]
59. Li, K.L.; Huang, H.K.; Tian, S.F.; Xu, W. Improving one-class SVM for anomaly detection. In Proceedings of the 2003 International Conference on Machine Learning and Cybernetics (IEEE Cat. No. 03EX693), Xi'an, China, 5 November 2003; IEEE: Piscataway, NJ, USA, 2003; Volume 5, pp. 3077–3081.
60. Ji, M.; Xing, H.J. Adaptive-weighted one-class support vector machine for outlier detection. In Proceedings of the 2017 29th Chinese Control and Decision Conference (CCDC), Chongqing, China, 28–30 May 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 1766–1771.
61. Tax, D.M.; Duin, R.P. Support vector data description. *Mach. Learn.* **2004**, *54*, 45–66. [[CrossRef](#)]
62. Wang, Z.; Zhao, Z.; Weng, S.; Zhang, C. Solving one-class problem with outlier examples by SVM. *Neurocomputing* **2015**, *149*, 100–105. [[CrossRef](#)]
63. Liu, F.T.; Ting, K.M.; Zhou, Z.H. Isolation forest. In Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, Pisa, Italy, 15–19 December 2008; IEEE: Piscataway, NJ, USA, 2008; pp. 413–422.
64. Primartha, R.; Tama, B.A. Anomaly detection using random forest: A performance revisited. In Proceedings of the 2017 International Conference on Data and Software Engineering (ICoDSE), Palembang, Indonesia, 1–2 November; IEEE: Piscataway, NJ, USA, 2017; pp. 1–6.
65. Tama, B.A.; Rhee, K.H. An in-depth experimental study of anomaly detection using gradient boosted machine. *Neural Comput. Appl.* **2019**, *31*, 955–965. [[CrossRef](#)]
66. Kieu, T.; Yang, B.; Jensen, C.S. Outlier detection for multidimensional time series using deep neural networks. In Proceedings of the 2018 19th IEEE International Conference on Mobile Data Management (MDM), Aalborg, Denmark, 25–28 June 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 125–134.
67. Thomas, R.; Judith, J. Voting-Based Ensemble of Unsupervised Outlier Detectors. In *Advances in Communication Systems and Networks*; Springer: Berlin/Heidelberg, Germany, 2020; pp. 501–511.
68. Yu, Q.; Jibin, L.; Jiang, L. An improved ARIMA-based traffic anomaly detection algorithm for wireless sensor networks. *Int. J. Distrib. Sens. Netw.* **2016**, *12*, 9653230. [[CrossRef](#)]
69. Zhou, Y.; Qin, R.; Xu, H.; Sadiq, S.; Yu, Y. A data quality control method for seafloor observatories: The application of observed time series data in the East China Sea. *Sensors* **2018**, *18*, 2628. [[CrossRef](#)]
70. Munir, M.; Siddiqui, S.A.; Chattha, M.A.; Dengel, A.; Ahmed, S. FuseAD: Unsupervised anomaly detection in streaming sensors data by fusing statistical and deep learning models. *Sensors* **2019**, *19*, 2451. [[CrossRef](#)] [[PubMed](#)]
71. Ibrahim, B.I.; Nicolae, D.C.; Khan, A.; Ali, S.I.; Khattak, A. VAE-GAN based zero-shot outlier detection. In Proceedings of the 2020 4th International Symposium on Computer Science and Intelligent Control, Newcastle upon Tyne, UK, 17–19 November 2020; pp. 1–5.
72. Xu, F.; Uszkoreit, H.; Du, Y.; Fan, W.; Zhao, D.; Zhu, J. Explainable AI: A brief survey on history, research areas, approaches and challenges. In Proceedings of the CCF International Conference on Natural Language Processing and Chinese Computing, Dunhuang, China, 9–14 October 2019; Springer: Berlin/Heidelberg, Germany, 2019; pp. 563–574.
73. Chan, L. Explainable AI as Epistemic Representation. In Proceedings of the AISB 2021 Symposium Overcoming Opacity in Machine Learning, London, UK, 7–9 April 2021; p. 7.
74. Müller, V.C. Deep Opacity Undermines Data Protection and Explainable Artificial Intelligence. In Proceedings of the AISB 2021 Symposium Overcoming Opacity in Machine Learning, London, UK, 7–9 April 2021; p. 18.
75. Ribeiro, M.T.; Singh, S.; Guestrin, C. “Why should I trust you?” Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; ACM: New York, NY, USA, 2016; pp. 1135–1144.

76. Hall, P.; Gill, N.; Kurka, M.; Phan, W. Machine Learning Interpretability with H₂O Driverless AI. H₂O. AI. 2017. Available online: <http://docs.h2o.ai/driverless-ai/latest-stable/docs/booklets/MLIBooklet.pdf> (accessed on 5 August 2021).
77. Zafar, M.R.; Khan, N.M. DLIME: A deterministic local interpretable model-agnostic explanations approach for computer-aided diagnosis systems. *arXiv* **2019**, arXiv:1906.10263.
78. Sokol, K.; Flach, P. LIMETree: Interactively Customisable Explanations Based on Local Surrogate Multi-output Regression Trees. *arXiv* **2020**, arXiv:2005.01427.
79. Ribeiro, M.T.; Singh, S.; Guestrin, C. Anchors: High-Precision Model-Agnostic Explanations. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LO, USA, 2–7 February 2018; Volume 18, pp. 1527–1535.
80. van der Waa, J.; Robeer, M.; van Diggelen, J.; Brinkhuis, M.; Neerincx, M. Contrastive explanations with local foil trees. *arXiv* **2018**, arXiv:1806.07470.
81. Guidotti, R.; Monreale, A.; Ruggieri, S.; Pedreschi, D.; Turini, F.; Giannotti, F. Local rule-based explanations of black box decision systems. *arXiv* **2018**, arXiv:1805.10820.
82. Štrumbelj, E.; Kononenko, I. Explaining prediction models and individual predictions with feature contributions. *Knowl. Inf. Syst.* **2014**, *41*, 647–665. [CrossRef]
83. Lundberg, S.; Lee, S.I. A unified approach to interpreting model predictions. *arXiv* **2017**, arXiv:1705.07874.
84. Bento, J.; Saleiro, P.; Cruz, A.F.; Figueiredo, M.A.; Bizarro, P. TimeSHAP: Explaining recurrent models through sequence perturbations. *arXiv* **2020**, arXiv:2012.00073.
85. Simonyan, K.; Vedaldi, A.; Zisserman, A. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv* **2013**, arXiv:1312.6034.
86. Shrikumar, A.; Greenside, P.; Shcherbina, A.; Kundaje, A. Not just a black box: Learning important features through propagating activation differences. *arXiv* **2016**, arXiv:1605.01713.
87. Shrikumar, A.; Greenside, P.; Kundaje, A. Learning important features through propagating activation differences. In Proceedings of the International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; pp. 3145–3153.
88. Sundararajan, M.; Taly, A.; Yan, Q. Axiomatic attribution for deep networks. In Proceedings of the International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; pp. 3319–3328.
89. Smilkov, D.; Thorat, N.; Kim, B.; Viégas, F.; Wattenberg, M. Smoothgrad: Removing noise by adding noise. *arXiv* **2017**, arXiv:1706.03825.
90. Vinayavekhin, P.; Chaudhury, S.; Munawar, A.; Agravante, D.J.; De Magistris, G.; Kimura, D.; Tachibana, R. Focusing on what is relevant: Time-series learning and understanding using attention. In Proceedings of the 2018 24th International Conference on Pattern Recognition (ICPR), Beijing, China, 20–24 August 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 2624–2629.
91. Viton, F.; Elbattah, M.; Guérin, J.L.; Dequen, G. Heatmaps for Visual Explainability of CNN-Based Predictions for Multivariate Time Series with Application to Healthcare. In Proceedings of the 2020 IEEE International Conference on Healthcare Informatics (ICHI), Virtual Conference, 30 November–3 December 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 1–8.
92. Schlegel, U.; Arnout, H.; El-Assady, M.; Oelke, D.; Keim, D.A. Towards a rigorous evaluation of xai methods on time series. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), Seoul, Korea, 27–28 October 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 4197–4201.
93. Pan, Q.; Hu, W.; Zhu, J. Series Saliency: Temporal Interpretation for Multivariate Time Series Forecasting. *arXiv* **2020**, arXiv:2012.09324.
94. Hsieh, T.Y.; Wang, S.; Sun, Y.; Honavar, V. Explainable Multivariate Time Series Classification: A Deep Neural Network Which Learns to Attend to Important Variables as Well as Time Intervals. In Proceedings of the 14th ACM International Conference on Web Search and Data Mining, Virtual Event, Israel, 8–12 March 2021; pp. 607–615.
95. Robnik-Šikonja, M.; Kononenko, I. Explaining classifications for individual instances. *IEEE Trans. Knowl. Data Eng.* **2008**, *20*, 589–600. [CrossRef]
96. Zeiler, M.D.; Fergus, R. Visualizing and understanding convolutional networks. In *Computer Vision—ECCV 2014, Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 818–833.
97. Fong, R.C.; Vedaldi, A. Interpretable explanations of black boxes by meaningful perturbation. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 3429–3437.
98. Petsiuk, V.; Das, A.; Saenko, K. Rise: Randomized input sampling for explanation of black-box models. *arXiv* **2018**, arXiv:1806.07421.
99. Antwarg, L.; Miller, R.M.; Shapira, B.; Rokach, L. Explaining anomalies detected by autoencoders using SHAP. *arXiv* **2019**, arXiv:1903.02407.
100. Park, C.H.; Kim, J. An explainable outlier detection method using region-partition trees. *J. Supercomput.* **2021**, *77*, 3062–3076. [CrossRef]
101. Jacob, V.; Song, F.; Stiegler, A.; Diao, Y.; Tatbul, N. AnomalyBench: An Open Benchmark for Explainable Anomaly Detection. *arXiv* **2020**, arXiv:2010.05073.
102. Song, F.; Diao, Y.; Read, J.; Stiegler, A.; Bifet, A. EXAD: A system for explainable anomaly detection on big data traces. In Proceedings of the 2018 IEEE International Conference on Data Mining Workshops (ICDMW), Singapore, 17–20 November 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1435–1440.

103. Amarasinghe, K.; Kenney, K.; Manic, M. Toward explainable deep neural network based anomaly detection. In Proceedings of the 2018 11th International Conference on Human System Interaction (HSI), Gdańsk, Poland, 4–6 July 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 311–317.
104. Singh, R.; Dourish, P.; Howe, P.; Miller, T.; Sonenberg, L.; Velloso, E.; Vetere, F. Directive explanations for actionable explainability in machine learning applications. *arXiv* **2021**, arXiv:2102.02671.
105. Artelt, A.; Hammer, B. On the computation of counterfactual explanations—A survey. *arXiv* **2019**, arXiv:1911.07749.
106. Spooner, T.; Dervovic, D.; Long, J.; Shepard, J.; Chen, J.; Magazzeni, D. Counterfactual Explanations for Arbitrary Regression Models. *arXiv* **2021**, arXiv:2106.15212.
107. Karimi, A.H.; Barthe, G.; Balle, B.; Valera, I. Model-agnostic counterfactual explanations for consequential decisions. In Proceedings of the International Conference on Artificial Intelligence and Statistics, Online, 26–28 August 2020; pp. 895–905.
108. Rüping, S. Learning Interpretable Models 2006. H₂O. AI. 2017. Available online: https://eldorado.tu-dortmund.de/bitstream/2003/23008/1/dissertation_rueping.pdf (accessed on 1 August 2021).
109. Pedreschi, D.; Giannotti, F.; Guidotti, R.; Monreale, A.; Pappalardo, L.; Ruggieri, S.; Turini, F. Open the black box data-driven explanation of black box decision systems. *arXiv* **2018**, arXiv:1806.09936.
110. Samek, W.; Müller, K.R. Towards explainable artificial intelligence. In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 5–22.
111. Verma, S.; Dickerson, J.; Hines, K. Counterfactual Explanations for Machine Learning: A Review. *arXiv* **2020**, arXiv:2010.10596.
112. Harris, C.R.; Millman, K.J.; van der Walt, S.J.; Gommers, R.; Virtanen, P.; Cournapeau, D.; Wieser, E.; Taylor, J.; Berg, S.; Smith, N.J.; et al. Array programming with NumPy. *Nature* **2020**, *585*, 357–362. [[CrossRef](#)] [[PubMed](#)]
113. McKinney, W. Data Structures for Statistical Computing in Python. In Proceedings of the 9th Python in Science Conference, Austin, TX, USA, 28 June–3 July 2010; pp. 56–61. [[CrossRef](#)]
114. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
115. Virtanen, P.; Gommers, R.; Oliphant, T.E.; Haberland, M.; Reddy, T.; Cournapeau, D.; Burovski, E.; Peterson, P.; Weckesser, W.; Bright, J.; et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nat. Methods* **2020**, *17*, 261–272. [[CrossRef](#)] [[PubMed](#)]
116. Zhao, Y.; Nasrullah, Z.; Li, Z. PyOD: A Python Toolbox for Scalable Outlier Detection. *J. Mach. Learn. Res.* **2019**, *20*, 1–7.
117. Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.Y. Lightgbm: A highly efficient gradient boosting decision tree. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 3146–3154.
118. Grinberg, M. *Flask Web Development: Developing Web Applications with Python*; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2018.
119. Plotly. *Collaborative Data Science*; P.T. Inc.: Montreal, QC, Canada, 2015.
120. jQuery: A Fast, Small, and Feature-Rich JavaScript Library. Available online: <https://jquery.com/> (accessed on 11 September 2021).
121. Hooker, S.; Erhan, D.; Kindermans, P.J.; Kim, B. A benchmark for interpretability methods in deep neural networks. *arXiv* **2018**, arXiv:1806.10758.
122. Koh, P.W.; Liang, P. Understanding black-box predictions via influence functions. In Proceedings of the International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; pp. 1885–1894.
123. Petluri, N.; Al-Masri, E. Web traffic prediction of wikipedia pages. In Proceedings of the 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 10–13 December 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 5427–5429.
124. Spiliotis, E.; Kouloumos, A.; Assimakopoulos, V.; Makridakis, S. Are forecasting competitions data representative of the reality? *Int. J. Forecast.* **2020**, *36*, 37–53. [[CrossRef](#)]
125. Hua, J.; Xiong, Z.; Lowey, J.; Suh, E.; Dougherty, E.R. Optimal number of features as a function of sample size for various classification rules. *Bioinformatics* **2005**, *21*, 1509–1515. [[CrossRef](#)] [[PubMed](#)]
126. Kraskov, A.; Stögbauer, H.; Grassberger, P. Erratum: Estimating mutual information [Phys. Rev. E 69, 066138 (2004)]. *Phys. Rev. E* **2011**, *83*, 019903. [[CrossRef](#)]
127. Smyl, S. A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting. *Int. J. Forecast.* **2020**, *36*, 75–85. [[CrossRef](#)]
128. Rabanser, S.; Januschowski, T.; Flunkert, V.; Salinas, D.; Gasthaus, J. The effectiveness of discretization in forecasting: An empirical study on neural time series models. *arXiv* **2020**, arXiv:2005.10111.
129. Hewamalage, H.; Bergmeir, C.; Bandara, K. Recurrent neural networks for time series forecasting: Current status and future directions. *Int. J. Forecast.* **2021**, *37*, 388–427. [[CrossRef](#)]
130. Stone, M. Cross-validatory choice and assessment of statistical predictions. *J. R. Stat. Soc. Ser. B (Methodol.)* **1974**, *36*, 111–133. [[CrossRef](#)]
131. Li, Z.; Zhao, Y.; Botta, N.; Ionescu, C.; Hu, X. COPOD: Copula-based outlier detection. In Proceedings of the 2020 IEEE International Conference on Data Mining (ICDM), Sorrento, Italy, 17–20 November 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 1118–1123.

-
132. Kriegel, H.P.; Schubert, M.; Zimek, A. Angle-based outlier detection in high-dimensional data. In Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, NV, USA, 24–27 August 2008; pp. 444–452.
 133. Fix, E.; Hodges, J.L. Discriminatory analysis. Nonparametric discrimination: Consistency properties. *Int. Stat. Rev. Int. Stat.* **1989**, *57*, 238–247. [[CrossRef](#)]

Chapter 5

Conclusions and Future Work

This thesis studies how machine learning can be applied to enhance demand forecasts. In particular, for smooth and erratic demands, we study how the forecasting error in global models can be constrained by pooling product data based on the past demand magnitude. For lumpy and intermittent demands, we propose a novel forecasting approach and a set of metrics to evaluate the quality of such forecasts. Given the sparse demand event occurrence in lumpy and intermittent demands, forecasting such events is paramount. When forecasting them, we are interested in two goals: (a) creating models that can discriminate between demand and non-demand events optimally, and (b) probabilistically calibrated so that the output scores convey information about how probable the demand event can be and therefore inform decision-making. While multiple metrics have been proposed to assess calibration quality, they have several shortcomings. In this research, we propose a set of novel metrics that overcome some shortcomings of the widely adopted Expected Calibration Error metric.

While accurate demand forecasting is of key importance, explainability of the forecasting models cannot be neglected. In our research, we have shown how explainable artificial intelligence approaches can be used to inform the user on relevant aspects affecting a given forecast. Furthermore, we considered semantic abstractions of the machine learning features to create human-friendly explanations while also leveraging such abstractions to preserve confidentiality, inform the user regarding possible events of interest, and suggest datasets for data enrichment. The role of semantic technologies in the context of demand forecasting is not limited to machine learning explainability. The forecasting models and semantic technologies are some of the key enablers of cognitive digital twins [48]. While forecasts can provide a glimpse into the future, the semantic technologies can connect a diverse set of data sources and models enabling the use of factual, inductive, and deductive knowledge and thus drive greater value to the end users.

The research performed regarding demand forecasting and explainability remained at a prototype level. Ensuring readiness of such approaches in production environment requires considering multiple scalability aspects, such as the amount of manual work to be performed, the magnitude of the data and requests processed, tolerated response times, and hardware requirements. We consider certain amount of manual work inevitable. When creating semantic technologies, such effort can be decreased by the use of proper tooling and by building a comprehensive ontology over time. When building machine learning models, the first models will require learning about the domain and ensure such expert knowledge is translated into features. Nevertheless, when such models are applied to new clients, semantic mappings can be created to identify data with same semantic meaning across different sources. Furthermore, semantic descriptions of the machine learning models can be used to automate training of existing and new models. Finally, best practices

regarding machine learning operations must be applied to automate model training, assessment, and deployment when new data is available. To deal with the magnitude of data, there is no upfront response: different types of data and use case requirements require vastly different solutions. Constraints regarding the velocity at which new data must be considered, the level of consistency required, and whether the data must be kept or not will influence decisions regarding data storage, processing and the type of machine learning models considered (e.g., streaming or batch). When considering requests latency, depending on the load and use case requirements, different software engineering approaches can be applied, leveraging asynchronous processing, multiple cache levels, load balancers, and autoscalers, among others.

5.1 Contributions to Science

The main scientific contributions of the thesis are discussed below.

SC 1. Development of a novel approach to forecasting intermittent and lumpy demand. We have shown that while intermittent and lumpy demand are characterized by sparse demand occurrence, most scientific work considered it a regression problem. While many authors understood that information regarding demand events occurrence should be included in the model, they mostly did so to hint at a regression model of whether the demand would be zero or greater than zero. We propose a novel approach by separating lumpy and intermittent demand forecasting into two separate problems: (a) classification, to determine whether the demand takes place for a given fixed forecasting horizon, and (b) regression, to determine the amount of product that would be demanded in case of demand event. The proposed approach has shown state-of-the-art performance: while most approaches described in the literature could not determine whether a demand event would take place, and the best model did so with an AUC ROC of 0.6955, our models could predict demand occurrence with an AUC ROC of 0.9700 (Rožanec et al. [34]).

SC 2. Definition of a novel set of metrics for assessing the quality of forecasting intermittent and lumpy demands. Through an extensive metrics analysis of the related work reporting research on lumpy and intermittent demand forecasting, we show that the authors considered only regression and inventory metrics to evaluate the forecasts' quality. In our work (Rožanec et al. [34]), we advocate measuring the forecast quality considering three dimensions: quality of demand event forecasts, quality of demand quantity forecasts, and quality of forecasts when considering inventory behavior. We proposed measuring the quality of demand event forecasts through a classification metric (AUC ROC) unaffected by the class imbalance. We propose two MASE variations to measure the quality of demand quantity forecasts. Our choice for MASE is grounded on the fact that the metric is independent of the scale of the data. Therefore, it enables comparing models regardless of the products considered and the differences in the magnitudes of their demand. To measure the quality of the forecast from an inventory perspective, we adopt the SPEC metric, which provides means to weigh the costs of stocking a product over time and the cost of not satisfying particular demand.

SC 3. Development of novel metrics to assess the quality of probability calibration of classification models without using ground truth. Expected Calibration Error is one of the most widely adopted calibration metrics. Nevertheless, it suffers from several shortcomings. Through the research conducted on a real-world use case (Rožanec et al. [35]), we propose a set of novel metrics and show that they capture the calibration dynamics better than the aforementioned metric. Furthermore, the proposed metrics enable additional calibration scenarios, such as calibrating only for an interval of interest (e.g., an interval relevant to decision-making) and multi-class probability calibration definitions.

The main conclusions of the experiments and research presented in this thesis are:

- Demand occurrence for lumpy and intermittent demands should be considered a separate forecasting problem and decoupled from forecasting demand quantity.
- For fixed forecasting horizons, demand events can be framed as a classification problem.
- When assessing the forecasts' quality for lumpy and intermittent demands and working with a fixed horizon, three metrics should be considered: (a) AUC ROC to determine how accurate the forecasts are when predicting demand occurrence, (b) MASE to determine how accurate the forecasts are when predicting demand quantity, and (c) SPEC to determine how the forecasts impact inventory.
- The newly proposed metrics for assessing the calibration quality overcome shortcomings from the widely adopted Expected Calibration Error metric. Furthermore, the proposed metrics can be extended to multi-class problems and adapted to address two possible probability calibration definitions: (a) confidence calibration and (b) class-wise calibration.

5.2 Future Work

Demand forecasting is key to effective supply chain management. This thesis has shown how machine learning can be applied to enhance demand forecasts. We have presented new approaches and metrics to forecast demand. Furthermore, we have developed novel metrics to assess the quality of probability calibration models. The contributions described in this thesis open new research directions and applications. First, we consider the approach we developed to forecast lumpy and intermittent demand in B2B could be applied for other sparse time series problems (e.g., forecasting demand in electric vehicle charging stations [49], or forecasting sparse sales [50]). Specific domain knowledge would guide the search for features that could signal an event occurrence. Furthermore, the external data could be used to further enhance the predictive power of the existing model. Second, our approach to measuring the calibration quality opens new research venues. We want to explore new calibration methods that could exploit information regarding the distance between the reference calibration distribution and the distribution of predicted values. Furthermore, we are interested in some calibration methods that can achieve better calibration over time even when no ground truth is available, given a set of premises are fulfilled (e.g., no concept drift takes place). Both contributions have a direct impact when implemented in applications. Our new approach for lumpy and intermittent demand forecasting delivers nearly a 40% improvement against the best existing approach described in the scientific literature when predicting demand occurrence in the use case we considered. On the other hand, the calibration metrics have been shown to better summarize the calibration quality than the widely adopted Expected Calibration Error while also providing informative insights when measuring the calibration quality without a ground truth and without concept drift.

References

- [1] B. Vogel-Heuser and D. Hess, "Guest editorial industry 4.0—prerequisites and visions," *IEEE Transactions on automation Science and Engineering*, vol. 13, no. 2, pp. 411–413, 2016. DOI: 10.1109/TASE.2016.2523639. [Online]. Available: <http://dx.doi.org/10.1109/TASE.2016.2523639>.
- [2] Y.-C. Chiu, F.-T. Cheng, and H.-C. Huang, "Developing a factory-wide intelligent predictive maintenance system based on industry 4.0," *Journal of the Chinese Institute of Engineers*, vol. 40, no. 7, pp. 562–571, 2017. DOI: 10.1080/02533839.2017.1362357. [Online]. Available: <https://doi.org/10.1080/02533839.2017.1362357>.
- [3] A. W. Melton, "Implications of short-term memory for a general theory of memory," *Journal of verbal Learning and verbal Behavior*, vol. 2, no. 1, pp. 1–21, 1963. DOI: 10.1016/S0022-5371(63)80063-8. [Online]. Available: [https://doi.org/10.1016/S0022-5371\(63\)80063-8](https://doi.org/10.1016/S0022-5371(63)80063-8).
- [4] H. Rindermann and A. C. Neubauer, "The influence of personality on three aspects of cognitive performance: Processing speed, intelligence and school performance," *Personality and individual differences*, vol. 30, no. 5, pp. 829–842, 2001. DOI: 10.1016/S0191-8869(00)00076-3.
- [5] F. E. Weinert, F.-W. Schrader, and A. Helmke, "Quality of instruction and achievement outcomes," *International Journal of Educational Research*, vol. 13, no. 8, pp. 895–914, 1989. DOI: 10.1016/0883-0355(89)90072-4.
- [6] L. M. for Teaching Project, "Measuring the mathematical quality of instruction," *Journal of Mathematics Teacher Education*, vol. 14, no. 1, pp. 25–47, 2011. DOI: 10.1007/s10857-010-9140-1.
- [7] J. Austin, H. Stevenson, and J. Wei-Skillern, "Social and commercial entrepreneurship: Same, different, or both?" *Entrepreneurship theory and practice*, vol. 30, no. 1, pp. 1–22, 2006. DOI: 10.1111/j.1540-6520.2006.00107. [Online]. Available: <https://doi.org/10.1111/j.1540-6520.2006.00107>.
- [8] A. A. Syntetos, J. E. Boylan, and J. Croston, "On the categorization of demand patterns," *Journal of the operational research society*, vol. 56, no. 5, pp. 495–503, 2005. DOI: 10.1057/palgrave.jors.2601841. [Online]. Available: <https://doi.org/10.1057/palgrave.jors.2601841>.
- [9] B. E. Flores and D. C. Whybark, "Multiple criteria abc analysis," *International Journal of Operations & Production Management*, 1986. DOI: 10.1108/eb054765. [Online]. Available: <https://doi.org/10.1108/eb054765>.
- [10] S. Mitra, M. S. Reddy, and K. Prince, "Inventory control using fsn analysis—a case study on a manufacturing industry," *International Journal of Innovative Science, Engineering & Technology*, vol. 2, no. 4, pp. 322–325, 2015.

- [11] B. Scholz-Reiter, J. Heger, C. Meinecke, and J. Bergmann, "Integration of demand forecasts in abc-xyz analysis: Practical investigation at an industrial company," *International Journal of Productivity and Performance Management*, 2012. DOI: 10.1108/17410401211212689. [Online]. Available: <https://doi.org/10.1108/17410401211212689>.
- [12] R. Botter and L. Fortuin, "Stocking strategy for service parts—a case study," *International Journal of Operations & Production Management*, 2000. DOI: 10.1108/01443570010321612. [Online]. Available: <https://doi.org/10.1108/01443570010321612>.
- [13] S. Nallusamy, "Performance measurement on inventory management and logistics through various forecasting techniques.," *International Journal of Performability Engineering*, vol. 17, no. 2, 2021. DOI: 10.23940/ijpe.21.02.p6.216228. [Online]. Available: <http://dx.doi.org/10.23940/ijpe.21.02.p6.216228>.
- [14] J. D. Croston, "Forecasting and stock control for intermittent demands," *Journal of the Operational Research Society*, vol. 23, no. 3, pp. 289–303, 1972. DOI: 10.1057/jors.1972.50. [Online]. Available: <https://doi.org/10.1057/jors.1972.50>.
- [15] T. Williams, "Stock control with sporadic and slow-moving demand," *Journal of the Operational Research Society*, vol. 35, no. 10, pp. 939–948, 1984. DOI: 10.1057/jors.1984.185. [Online]. Available: <https://doi.org/10.1057/jors.1984.185>.
- [16] A. H. Eaves and B. G. Kingsman, "Forecasting for the ordering and stock-holding of spare parts," *Journal of the Operational Research Society*, vol. 55, no. 4, pp. 431–437, 2004. DOI: 10.1057/palgrave.jors.2601697. [Online]. Available: <https://doi.org/10.1057/palgrave.jors.2601697>.
- [17] R. Verganti, "Order overplanning with uncertain lumpy demand: A simplified theory," *International journal of production research*, vol. 35, no. 12, pp. 3229–3248, 1997. DOI: 10.1080/002075497194057. [Online]. Available: <https://doi.org/10.1080/002075497194057>.
- [18] E. Bartezzaghi, R. Verganti, and G. Zotteri, "A simulation framework for forecasting uncertain lumpy demand," *International Journal of Production Economics*, vol. 59, no. 1-3, pp. 499–510, 1999. DOI: 10.1016/S0925-5273(98)00012-7. [Online]. Available: [https://doi.org/10.1016/S0925-5273\(98\)00012-7](https://doi.org/10.1016/S0925-5273(98)00012-7).
- [19] K. Nikolopoulos, A. A. Syntetos, J. E. Boylan, F. Petropoulos, and V. Assimakopoulos, "An aggregate–disaggregate intermittent demand approach (adida) to forecasting: An empirical proposition and analysis," *Journal of the Operational Research Society*, vol. 62, no. 3, pp. 544–554, 2011. DOI: 10.1057/jors.2010.32. [Online]. Available: <https://doi.org/10.1057/jors.2010.32>.
- [20] M. Z. Babai, A. Syntetos, and R. Teunter, "Intermittent demand forecasting: An empirical study on accuracy and the risk of obsolescence," *International Journal of Production Economics*, vol. 157, pp. 212–219, 2014. DOI: 10.1016/j.ijpe.2014.08.019. [Online]. Available: <https://doi.org/10.1016/j.ijpe.2014.08.019>.
- [21] F. Johnston, J. E. Boylan, and E. A. Shale, "An examination of the size of orders from customers, their characterisation and the implications for inventory control of slow moving items," *Journal of the Operational Research Society*, vol. 54, no. 8, pp. 833–837, 2003. DOI: 10.1057/palgrave.jors.2601586. [Online]. Available: <https://doi.org/10.1057/palgrave.jors.2601586>.

- [22] M. R. Amin-Naseri and B. R. Tabar, “Neural network approach to lumpy demand forecasting for spare parts in process industries,” in *2008 International Conference on Computer and Communication Engineering*, IEEE, 2008, pp. 1378–1382. DOI: 10.1109/ICCCE.2008.4580831. [Online]. Available: <http://dx.doi.org/10.1109/ICCCE.2008.4580831>.
- [23] S. Mukhopadhyay, A. O. Solis, and R. S. Gutierrez, “The accuracy of non-traditional versus traditional methods of forecasting lumpy demand,” *Journal of Forecasting*, vol. 31, no. 8, pp. 721–735, 2012. DOI: 10.1002/for.1242. [Online]. Available: <https://doi.org/10.1002/for.1242>.
- [24] F. Petropoulos, D. Apiletti, V. Assimakopoulos, *et al.*, “Forecasting: Theory and practice,” *International Journal of Forecasting*, 2022. DOI: 10.1016/j.ijforecast.2021.11.001. [Online]. Available: <https://doi.org/10.1016/j.ijforecast.2021.11.001>.
- [25] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, “The m4 competition: 100,000 time series and 61 forecasting methods,” *International Journal of Forecasting*, vol. 36, no. 1, pp. 54–74, 2020. DOI: 10.1016/j.ijforecast.2019.04.014. [Online]. Available: <https://doi.org/10.1016/j.ijforecast.2019.04.014>.
- [26] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, “The m5 accuracy competition: Results, findings and conclusions,” *International Journal of Forecasting*, vol. 38, no. 4, pp. 1346–1364, 2022, Special Issue: M5 competition, ISSN: 0169-2070. DOI: 10.1016/j.ijforecast.2021.11.013. [Online]. Available: <https://doi.org/10.1016/j.ijforecast.2021.11.013>.
- [27] P. Montero-Manso and R. J. Hyndman, “Principles and algorithms for forecasting groups of time series: Locality and globality,” *International Journal of Forecasting*, vol. 37, no. 4, pp. 1632–1653, 2021, ISSN: 0169-2070. DOI: 10.1016/j.ijforecast.2021.03.004. [Online]. Available: <https://doi.org/10.1016/j.ijforecast.2021.03.004>.
- [28] H. Song, M. Perello-Nieto, R. Santos-Rodriguez, M. Kull, P. Flach, *et al.*, “Classifier calibration: How to assess and improve predicted class probabilities: A survey,” *arXiv preprint arXiv:2112.10327*, 2021.
- [29] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, “On calibration of modern neural networks,” in *International conference on machine learning*, PMLR, 2017, pp. 1321–1330.
- [30] B. Zadrozny and C. Elkan, “Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers,” in *Proceedings of the Eighteenth International Conference on Machine Learning*, ser. ICML ’01, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001, pp. 609–616, ISBN: 1558607781.
- [31] N. Posocco and A. Bonnefoy, “Estimating expected calibration errors,” in *Artificial Neural Networks and Machine Learning–ICANN 2021: 30th International Conference on Artificial Neural Networks, Bratislava, Slovakia, September 14–17, 2021, Proceedings, Part IV 30*, Springer, 2021, pp. 139–150. DOI: 10.1007/978-3-030-86380-7_12. [Online]. Available: https://doi.org/10.1007/978-3-030-86380-7_12.
- [32] J. Nixon, M. W. Dusenberry, L. Zhang, G. Jerfel, and D. Tran, “Measuring calibration in deep learning,” in *CVPR workshops*, vol. 2, 2019.

- [33] M. P. Naeini, G. F. Cooper, and M. Hauskrecht, "Obtaining well calibrated probabilities using bayesian binning," in *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, ser. AAAI'15, Austin, Texas: AAAI Press, 2015, pp. 2901–2907, ISBN: 0262511290.
- [34] J. M. Rožanec, B. Fortuna, and D. Mladenović, "Reframing demand forecasting: A two-fold approach for lumpy and intermittent demand," *Sustainability*, vol. 14, no. 15, p. 9295, 2022. DOI: 10.3390/su14159295. [Online]. Available: <https://doi.org/10.3390/su14159295>.
- [35] J. M. Rožanec, L. Bizjak, E. Trajkova, *et al.*, "Active learning and novel model calibration measurements for automated visual inspection in manufacturing," *Journal of Intelligent Manufacturing*, 2023. DOI: 10.1007/s10845-023-02098-0. [Online]. Available: <https://doi.org/10.1007/s10845-023-02098-0>.
- [36] J. M. Rožanec, B. Kažič, M. Škrjanc, B. Fortuna, and D. Mladenović, "Automotive oem demand forecasting: A comparative study of forecasting algorithms and strategies," *Applied Sciences*, vol. 11, no. 15, p. 6787, 2021. DOI: 10.3390/app11156787. [Online]. Available: <https://doi.org/10.3390/app11156787>.
- [37] J. Rožanec, E. Trajkova, K. Kenda, B. Fortuna, and D. Mladenović, "Explaining bad forecasts in global time series models," *Applied Sciences*, vol. 11, no. 19, p. 9243, 2021. DOI: 10.3390/app11199243. [Online]. Available: <https://doi.org/10.3390/app11199243>.
- [38] Ç. Pınar, L. Turrini, and J. Meissner, "Intermittent demand forecasting for spare parts: A critical review," *Omega*, vol. 105, p. 102513, 2021. DOI: 10.1016/j.omega.2021.102513. [Online]. Available: <https://doi.org/10.1016/j.omega.2021.102513>.
- [39] M. Sigwarth, "Identifying drivers of forecasting model performance for highly intermittent sku demand," 2021.
- [40] R. Dijkman, I. Adan, Z. Bukhsh, I. K. Herps, and A. Van de Velde, "Data-driven reliability calculations of demand forecast data," 2021.
- [41] X. Zhuang, Y. Yu, and A. Chen, "A combined forecasting method for intermittent demand using the automotive aftermarket data," *Data Science and Management*, 2022. DOI: 10.1016/j.dsm.2022.04.001. [Online]. Available: <https://doi.org/10.1016/j.dsm.2022.04.001>.
- [42] R. Sarlo, C. Fernandes, and D. Borenstein, "Lumpy and intermittent retail demand forecasts with score-driven models," *European Journal of Operational Research*, vol. 307, no. 3, pp. 1146–1160, 2023, ISSN: 0377-2217. DOI: 10.1016/j.ejor.2022.10.006. [Online]. Available: <https://doi.org/10.1016/j.ejor.2022.10.006>.
- [43] E. Chanada, F. Riady, A. Aurellia, *et al.*, "Pengelolaan permintaan dan kapasitas produksi pada getnailed. id di kota Batam," *J-MAS: Jurnal Pengabdian Masyarakat*, vol. 1, no. 3, pp. 271–278, 2022. DOI: 10.59004/jmas.v1i3.72. [Online]. Available: <https://doi.org/10.59004/jmas.v1i3.72>.
- [44] R. R. Hoffman, S. T. Mueller, G. Klein, and J. Litman, "Metrics for explainable ai: Challenges and prospects," *arXiv preprint arXiv:1812.04608*, 2018.
- [45] J. Rožanec, "Explainable demand forecasting: A data mining goldmine," *Companion Proceedings of the Web Conference 2021 (WWW '21 Companion)*, April 19–23, 2021, Ljubljana, Slovenia, 2021. DOI: 10.1145/3442442.3453708.

- [46] J. M. Rožanec, B. Fortuna, and D. Mladenić, “Knowledge graph-based rich and confidentiality preserving explainable artificial intelligence (xai),” *Information Fusion*, vol. 81, pp. 91–102, 2022. DOI: 10.1016/j.inffus.2021.11.015. [Online]. Available: <https://doi.org/10.1016/j.inffus.2021.11.015>.
- [47] J. Rožanec, E. Trajkova, I. Novalija, *et al.*, “Enriching artificial intelligence explanations with knowledge fragments,” *Future Internet*, vol. 14, no. 5, p. 134, 2022. DOI: 10.3390/fi14050134. [Online]. Available: <https://doi.org/10.3390/fi14050134>.
- [48] J. M. Rožanec, J. Lu, J. Rupnik, *et al.*, “Actionable cognitive twins for decision making in manufacturing,” *International Journal of Production Research*, vol. 60, no. 2, pp. 452–478, 2022.
- [49] M. Majidpour, C. Qiu, P. Chu, R. Gadh, and H. R. Pota, “Fast prediction for sparse time series: Demand forecast of ev charging stations for cell phone applications,” *IEEE Transactions on Industrial Informatics*, vol. 11, no. 1, pp. 242–250, 2014. DOI: 10.1109/TII.2014.2374993. [Online]. Available: <http://dx.doi.org/10.1109/TII.2014.2374993>.
- [50] A. Jha, S. Ray, B. Seaman, and I. S. Dhillon, “Clustering to forecast sparse time-series data,” in *2015 IEEE 31st International Conference on Data Engineering*, IEEE, 2015, pp. 1388–1399. DOI: 10.1109/ICDE.2015.7113385. [Online]. Available: <http://dx.doi.org/10.1109/ICDE.2015.7113385>.

Bibliography

Publications Related to the Thesis

1.01 Original Scientific Articles

- [1] J. M. Rožanec, B. Kažič, M. Škrjanc, B. Fortuna, and D. Mladenec, “Automotive oem demand forecasting: A comparative study of forecasting algorithms and strategies,” *Applied Sciences*, vol. 11, no. 15, p. 6787, 2021. DOI: 10.3390/app11156787. [Online]. Available: <https://doi.org/10.3390/app11156787>.
- [2] J. M. Rožanec, B. Fortuna, and D. Mladenec, “Reframing demand forecasting: A two-fold approach for lumpy and intermittent demand,” *Sustainability*, vol. 14, no. 15, p. 9295, 2022. DOI: 10.3390/su14159295. [Online]. Available: <https://doi.org/10.3390/su14159295>.
- [3] J. M. Rožanec, L. Bizjak, E. Trajkova, *et al.*, “Active learning and novel model calibration measurements for automated visual inspection in manufacturing,” *Journal of Intelligent Manufacturing*, 2023. DOI: 10.1007/s10845-023-02098-0. [Online]. Available: <https://doi.org/10.1007/s10845-023-02098-0>.
- [4] J. Rožanec, E. Trajkova, K. Kenda, B. Fortuna, and D. Mladenec, “Explaining bad forecasts in global time series models,” *Applied Sciences*, vol. 11, no. 19, p. 9243, 2021. DOI: 10.3390/app11199243. [Online]. Available: <https://doi.org/10.3390/app11199243>.
- [5] J. M. Rožanec, B. Fortuna, and D. Mladenec, “Knowledge graph-based rich and confidentiality preserving explainable artificial intelligence (xai),” *Information Fusion*, vol. 81, pp. 91–102, 2022. DOI: 10.1016/j.inffus.2021.11.015. [Online]. Available: <https://doi.org/10.1016/j.inffus.2021.11.015>.
- [6] J. Rožanec, E. Trajkova, I. Novalija, *et al.*, “Enriching artificial intelligence explanations with knowledge fragments,” *Future Internet*, vol. 14, no. 5, p. 134, 2022. DOI: 10.3390/fi14050134. [Online]. Available: <https://doi.org/10.3390/fi14050134>.

1.08 Published Scientific Conference Contributions

- [1] J. Rožanec, “Explainable demand forecasting: A data mining goldmine,” *Companion Proceedings of the Web Conference 2021 (WWW ’21 Companion)*, April 19–23, 2021, Ljubljana, Slovenia, 2021. DOI: 10.1145/3442442.3453708.

Biography

JOSE MARTIN ROŽANEC was born in Buenos Aires, Argentina. He graduated with honors as a software engineer at Universidad Austral. He complemented his studies with a Post-graduate Diploma of Teaching in Higher Education at the Universidad Austral, and later a Master programme in Data Mining and Knowledge Discovery at the Universidad de Buenos Aires (thesis pending). In 2016, he was awarded the ASEF Fellowship and got the opportunity to perform research with Prof. Dr. Dawn Song at the University of California, Berkeley.

Between 2010 and 2018 he worked in several companies in software engineering and machine learning roles, while also teaching for certain periods. Among the companies he worked for were Mercado Libre, Navent, and Globant - two of them unicorns. In 2018, he joined Qlector d.o.o. and started a Ph.D. in Information and Communication Technologies at the Jožef Stefan International Postgraduate School while working full time. He performed his Ph.D. under the supervision of Prof. Dr. Dunja Mladenić and Dr. Blaž Fortuna. In 2020, he reduced his dedication at Qlector d.o.o. to become a researcher at the Jožef Stefan Institute. There he joined two successful European Horizon 2020 projects: FACTLOG and STAR. Both projects gave him the opportunity to dive deeper into research, gain deeper knowledge, and get to know multiple brilliant researchers. His current research interests include machine learning methods for demand forecasting, active learning, and explainable artificial intelligence (XAI).

