RESEARCH PAPER

Explainable product backorder prediction exploiting CNN: Introducing explainable models in businesses

Md Shajalal[1,2] · Alexander Boden[1,3] · Gunnar Stevens[2,3]

Abstract

Due to expected positive impacts on business, the application of artificial intelligence has been widely increased. The decision-making procedures of those models are often complex and not easily understandable to the company's stakeholders, i.e. the people having to follow up on recommendations or try to understand automated decisions of a system. This opaqueness and black-box nature might hinder adoption, as users struggle to make sense and trust the predictions of AI models. Recent research on eXplainable Artificial Intelligence (XAI) focused mainly on explaining the models to AI experts with the purpose of debugging and improving the performance of the models. In this article, we explore how such systems could be made explainable to the stakeholders. For doing so, we propose a new convolutional neural network (CNN)-based explainable predictive model for product backorder prediction in inventory management. Backorders are orders that customers place for products that are currently not in stock. The company now takes the risk to produce or acquire the backordered products while in the meantime, customers can cancel their orders if that takes too long, leaving the company with unsold items in their inventory. Hence, for their strategic inventory management, companies need to make decisions based on assumptions. Our argument is that these tasks can be improved by offering explanations for AI recommendations. Hence, our research investigates how such explanations could be provided, employing Shapley additive explanations to explain the overall models' priority in decision-making. Besides that, we introduce locally interpretable surrogate models that can explain any individual prediction of a model. The experimental results demonstrate effectiveness in predicting backorders in terms of standard evaluation metrics and outperform known related works with AUC 0.9489. Our approach demonstrates how current limitations of predictive technologies can be addressed in the business domain.

✉ Md Shajalal
md.shajalal@fit.fraunhofer.de

✉ Alexander Boden
alexander.boden@fit.fraunhofer.de

Gunnar Stevens
gunnar.stevens@uni-siegen.de

1   Fraunhofer-Institute for Applied Information Technology FIT, Schloss Birlinghoven, 53757 Sankt Augustin, Germany

2   University of Siegen, Siegen 57072, Germany

3   Bonn-Rhein-Sieg University of Applied Science, 53757 Bonn, Germany

Introduction

Due to their superior predictive performance, complex machine learning and deep neural network-based models have received high attention and are widely exploited in the business domain (Bawack et al., 2022; Janiesch et al., 2021; Cliff et al., 2011) along with other fields including image processing (Jiao and Zhao, 2019), health (Panesar, 2019; Bartoletti, 2019) and bioinformatics (Cao et al., 2020; Li et al., 2019). The tasks of those technologies range across different application areas including supply chain management, credit risk prediction (Moscato et al., 2021; Bussmann et al., 2021), detection of fraud credit card transaction (Carcillo et al., 2021; Randhawa et al., 2018) and marketing campaigns in retail banking (Ładyżyński et al., 2019).

Generally, artificial intelligence (AI) techniques employ a huge size of training data for making predictions. While there is a huge interest in such predictions in various business domains

(Ribeiro et al., 2016), one of the major problems of complex machine learning models is that they are very difficult to understand (Abedin et al., 2022; Adadi et al., 2018; Thiebes et al., 2021). Several Methods using induced ordered weighted averaging (IOWA) adaptive neuro-fuzzy inference system (ANFIS) can deal with multidimensional data to predict the quality of service and hence it help stakeholders in the decision-making process (Hussain et al., 2022a, b, 2021). As decisions often depend on a huge number of model parameters (Alvarez-Melis and Jaakkola, 2017), machine learning and deep learning techniques are like black-boxes or magic boxes to the general users (and often even for developers). The higher the accuracy of a complex machine learning model, the more opaque the models tend to become (Ribeiro et al., 2016). This opaqueness leads to a situation where users might question the predictions, because they are unable to understand the underlying decision making processes (i.e. the reasons for why a maybe counter-intuitive recommendation has been given) (Arya et al., 2019).

User acceptance is generally one of the main barriers for the success of technologies in companies. As AI-based recommendations can potentially have a huge impact on operational as well as strategic decisions in companies, it seems to be beneficial if users or consumers of AI models could better understand why those recommendations have been made (Meske et al., 2022). Apart from increasing trust in AI-recommendations, having factual explanations of a certain decision would also help users to learn about the field of application (for instance gaining a better understanding in the importance or non-importance of certain factors for business decisions) (Förster et al., 2020). In addition, according to the general data protection regulation (GDPR) by the European Union, EU citizens have the right to receive explanations about AI-based decisions, for instance if an AI recommendation affects credit worthiness or insurance rates (Meske et al., 2022; Došilović et al., 2018).

In this research, we propose a novel explainable predictive model for product backorder prediction. A backorder is a situation where customers can order a product even though that particular product is out of stock at the time when the order is placed (Hajek and Abedin, 2020; Ntakolia et al., 2021). Basically, its an order to a future inventory, going along with contingencies as time of delivery can vary and is not definitely known. Backorders are especially common for items that are highly popular. While for some items such as the latest flagship Apple iPhone, such events are quite common, it can be very unpredictable for other types of products. When retail companies order high amounts of products based on backorders, they risk their reputation if they are unable to keep the expected delivery dates. Another risk is that customers can cancel their orders because they don't want to wait any longer or found another retailer where the product is in stock, leaving the company with excess products in their inventory. Here, predictive models can help to

tackle these challenge by predicting the probability whether a certain product will be backordered or not, giving companies more time to plan and supporting them in their inventory management. In related works, researchers have proposed complex machine learning based methods to predict future product backorders. The predictive models include the application of support vector machine, XBoost, ensemble classifier and deep neural networks (Islam and Amin, 2020; Hajek and Abedin, 2020; Li, 2017; Shajalal et al., 2021).

However, the mere prediction of future backorders only solves part of the problem. Suppose you are responsible for a particular inventory management system at a retail company. When you are notified that the AI model decided that a particular product is going to be backordered in the near future, what will you do? *Would you increase the inventory level (i.e. obtaining more products in advance)? Would you change any policy (negotiating with suppliers about faster transit times, lead times etc)?* If you increase the inventory level, *how many products would you order, assuming that some would surely be cancelled?* For taking these decisions, you would need to understand the reasons for the prediction. Hence, our approach tries to provide insights into the factors that contribute to a certain prediction, helping users to adapt their strategies accordingly. Our paper contributes in the following ways:

- We proposed a new CNN-based model for backorder prediction. Since backorders are rare events in inventory management systems, it is a challenging task to identify them. Their rarity leads to an extremely imbalanced distribution within datasets. Often, the percentage of the backodered samples is less than 0.01% (specifically 0.007%, de Santis et al. (2017)). To address this data imbalance, we incorporated an adaptive synthetic oversampling (ADASYN) technique that generates synthetic samples for a minority class. The results, based on diverse experimental settings and comparison with existing known related works, illustrated that our method achieved better prediction performance achieving a new state-of-the-art methods performance in terms of standard evaluation metrics.
- To provide an overall insight of the predictive model's decision-making priorities, we investigate the impact of different attributes of an order in the predictive models. We introduce an XAI technique, namely SHAP (Shapely additive explanations), that can interpret and/ or explain the predictive model to identify the most important attributes of the decision making. Hence the stakeholders are enabled to better understand the model's decision-making priorities and consider that when they have to work with such technologies.
- By explaining specific predictions, our method can answer why a particular product will be backordered or not. Every order has different feature's values which are

considered to make predictions. Therefore, we trained a local interpretable surrogate model employing LIME (local interpretable model-agnostic explanations), and present explanations for an individual prediction to answer the question "*why has this specific decision been made?*" Hence, stakeholders can not only assess the models' priorities in general, but also analyse singular decisions to better understand them.

The organization of the rest of this paper is as follows: Section 2 summarises related works on predicting product backorders. We present a brief discussion about different XAI terminologies in Section 3. In Section 4, we present our method for predicting future product backorders and the explanation generation techniques. The predictive performance of our proposed CNN-based method and performance comparison with classical machine learning classifiers and known related works are presented in detail in Section 5. The decisions of complex machine learning and deep learning models are explained through different types of explanations both for models' priorities as well as specific predictions in Section 6. Finally, Section 7 concludes our proposed methods and findings of this study by discussing the prospects of introducing XAI technology in the business domain.

Related work

This section presents the discussion of related research on backorder prediction and explainable artificial intelligence in supply chain management. Existing works proposed different models to predict plausible future backorders in inventory management systems. Based on the types of techniques used, the predictive models can broadly be classified into two categories: i) Classical machine learning classifiers and ii) Deep learning-based predictive models. In the former category, the classifiers include support vector machine (Hajek and Abedin, 2020), gradient boosting (Ntakolia et al., 2021; de Santis et al., 2017), decision trees, and random forests (Islam and Amin, 2020). The deep learning-based models employed recurrent neural networks (RNN) (Li, 2017), deep auto-encoders (Saraogi et al., 2021), as well as deep neural networks (DNN) (Shajalal et al., 2021).

Islam and Amin (2020) proposed a method to predict future backorders by applying distributed random forest and gradient boosting classifiers. They introduced a ranged-based approach to cope with the numerous types of real-time data. However, they did not include some features of the samples such as features related to inventory level, previous sales, future sale forecasting, and lead time. A profit-maximizing function based approach is introduced by Hajek and Abedin (2020).

They aligned their profit maximization function with classical machine learning classifiers. The performance of their methods demonstrated how much profit can be increased by predicting future backorders. An explainable classical machine learning-based method is proposed by Ntakolia et al. (2021). Their method applied several classifiers such as random forest, XGBoost, SVM, etc. They also applied shapely additive values to present the global explanations to interpret the models. Similarly, de Santis et al. (2017) also used different classical classifiers. The performance of deep learning approaches is comparatively better than the classical classifiers. Shajalal et al. (2021) proposed a deep neural network (DNN) based backorder prediction model. Inspired by the success of deep learning classifiers, Li (2017), Saraogi et al. (2021) and Lawal and Akintola (2021) applied deep auto-encoder, a recurrent neural network-based classification models.

Backorders are not a common scenario in inventory management systems. In turn, the number of non-backordered items is much larger than the backordered ones. Hence, real-time data collected from any inventory system will be strongly imbalanced, leading to challenges in predicting future backorders on that basis. In this particular task (Li, 2017), the ratio between majority (non-backordered) and minority (backordered) samples is 100:0.007. In the case of an imbalanced dataset, the classifiers might learn the pattern with potential bias. That is why different under-sampling, oversampling, and class weight-based approaches are common to balance the dataset and bias (Hajek and Abedin, 2020). Randomly duplicating the minority samples or randomly discarding the majority samples has also been applied to balance the dataset (Chawla et al., 2002). But randomly duplicating the minority samples will increase redundant samples and hence the model might be biased. Therefore, generating synthetic minority samples based on the Euclidean distance is a popular approach to balance the dataset. This method is called SMOTE (Synthetic Minority Over-sampling Technique) (Chawla et al., 2002). The combination of SMOTE and random under-sampling has been applied by Hajek and Abedin (2020) and Shajalal et al. (n.d., 2021). Li (2017) applied different balancing techniques including SMOTE, ADASYN (Adaptive Synthetic Sampling) (He et al., 2008) and random under-sampling. Bagging (Błaszczyński and Stefanowski, 2015) is also applied for the same purpose by (de Santis et al., 2017). Table 1 summarises the existing methods for predicting product backorders.

However, to the best of our knowledge, none of the studies applied XAI to interpret their machine learning model except Ntakolia et al. (2021).

In our paper, we propose a convolutional neural network framework-based model that outperformed different classifiers including classical and deep learning-based models in backorder prediction. Ntakolia et al. (2021) interpreted only classical models mainly with global explanations. Our method integrated explainable artificial intelligence that generates global explanations for the classical and deep learning-based prediction

Table 1 The summary of existing study on product backorder prediction

Research paper	Contributions	Explainabaility
Islam and Amin (2020)	• Applied Distributed Random Forest (DRF) and Gradient Boosting Machine	• Inherent (Global feature importance)
	• Incorporating SMOTE oversampling	• No local explanation to understand particular decision
Hajek and Abedin (2020)	• A genetic algorithm-based profit maximizing prediction system	• Not Explainable
	• Applied classical ML classifiers	
Shajalal et al. (2021)	• Proposed a deep neural network-based prediction model	• Not Explainable
	• Combined random undersampling and synthetic oversampling to overcome class imbalance problem	
Li (2017)	• Applied recurrent neural network (RNN) based predictive model	• Not Explainable
	• Exploited SMOTE, ADASYN, and random undersampling for balancing the dataset	
Saraogi et al. (2021)	• Proposed deep autoencoder based model for backorder prediction	• Not Explainable
	• Used unsupervised approach rather than supervised one	
Ntakolia et al. (2021, 2021)	• Introduced classical machine learning model to predict backorder	• Interpreted the global feature importance to explain the model
	• Incorporated interpretability to understand decision making	• No local explanation for particular decisions
de Santis et al. (2017)	• Exploited classical machine learning classifier including gradient boosting and ensemble model	• Not Explainable
	• Used bagging to overcome data imbalance problem	
LAWAL and AKINTOLA (2021)	• Applied recurrent neural network (RNN) based predictive model	• Not Explainable
	• Exploited SMOTE, ADASYN, and random undersampling for balancing the dataset	

model. Though the global interpretation is useful to illustrate the general mechanisms and behavior of the model, it can not explain a particular prediction. We introduced a model applying shapely additive explanation (Lundberg and Lee, 2017) and local interpretable model-agnostic explanation (Ribeiro et al., 2016) to interpret the overall model and local specific decisions.

To clearly illustrate the research gap in the existing literature and our research focus, we present a comparative analysis in Table 2.

XAI terminology

In this paper, we employed two XAI techniques, namely shapely additive explanations (SHAP) and local interpretable model-agnostic explanations (LIME). Here, we present the background and working principle of these two techniques.

SHapely Additive exPlanation (SHAP)

Lundberg and Lee Lundberg and Lee (2017) first proposed a unified approach to explain and interpret the prediction of machine learning models. The explanations basically illustrate the contributions (positive and negative importance or influence) of different features for the predicted decision of a particular sample x. The overall feature importance of different features of the whole model can also be interpreted as global explanations. In that case, the importance score resembles the weight of features as in the linear model. The SHAP values represent the importance of the features. The explanation of every single prediction can be seen as a vector of shap values. The same representation is used to interpret the overall model. For a given instance x, the explanation using SHAP can be defined as

Table 2 Existing research gaps in explainable product backorder prediction and our steps to fulfil the research gaps

Issue	Research gaps in literature	Our contributions
Performance	Most existing methods suffer from low performance in modelling product backorder prediction due to extreme data imbalance problem. The majority of prior studies applied classical ML methods.	We proposed a novel CNN-based prediction model with the ADASYN technique that achieved new state-of-the-art performance.
Model's Interpretability	Lack of interest in applying XAI to explain the predictive model's decision-making priorities. Hence the existing models can be seen by the stakeholders as black-box.	We introduced shapely additive explanation (SHAP), one of the most successful XAI techniques to explain the models' global priorities that help stakeholders in sense-making about the working strategy of the predictive models.
Local Explaiability	No existing works explain the specific prediction to answer *"why has this specific decision been made?"* (i.e., why a certain product is going to be backordered?)	We exploit LIME and SHAP to explain specific predictions about why a particular product is assumed to be on backordered or not. These techniques can explain which features/attributes are responsible for a particular decision. Hence the stakeholders are enabled to take steps to overcome future backorder and reduce the company's loss.

$$g(z') = \phi_0 + \sum_{i=1}^{M} \phi_j z'_j, \tag{1}$$

where g is denoted as the explanation model. The vector for simplified features, known as the coalition vector is represented by z' ($z' \in \{0,1\}^M$). The 1 represents that features' values are the same as the original instances and vice-versa. The attribution of particular features j of the instance x is denoted by ϕ_j which is a real number. The higher the value of ϕ_j, the more important the feature j. The ϕ_j is computed based on Shapely values (Nowak and Radzik, 1994), a game-theoretic approach that identifies and detects the contribution of all players in a collaborative game. The collaborative game with multiple players is analogue to the prediction of the instance having multiple features. In turn, applying this game-theoretic approach we can examine the contribution of each feature to a particular decision. For a given feature vector x' and a predictive model f, the computation is done as follows:

$$\phi_i(f, x') = \sum_{z' \subseteq \{x'_1, x'_2, \dots, x'_n\} \setminus \{x'_i\}} \frac{(|z'|)! \, (M - |z'| - 1)!}{M!} \cdot [f(z' \cup x'_i) - f(z')], \tag{2}$$

The subset of the features employed by the model is denoted as z'. x' is the vector with features values to be explained and can be defined as $[f(z' \cup x'_i) - f(z')]$ and M is the number of features. The prediction by the model f is denoted by $f(z')$.

Moreover, SHAP values are computed by a standard game-theoretical approach and utilized Shapely values to have a unified interpretable model with fast computation. More mathematical and technical details for SHAP can be found in the study published by Lundberg and Lee (2017) as well as in Nowak and Radzik (1994).

Local interpretable model-agnostic explanation (LIME)

LIME mainly provides model-agnostic explanations based on local surrogate models. Ribeiro et al. (2016) first introduced this approach for training a local surrogate model instead of a global model for providing explanations for a particular prediction. LIME employed a new local dataset containing the permuted samples with corresponding predictions to train the local interpretable surrogate model. This surrogate model is then used to explain individual predictions. The model is considered as an approximation of the original complex, black-box predictive model. The computation of the surrogate model can be defined as follows:

$$\xi(x) = \arg \min_{g \in \mathbf{G}} L(f, g, \pi_{x'}) + \Omega(g) \tag{3}$$

The explanation model for a particular instance x and the explanation family are represented by g and G, respectively. The original model is denoted by f and L is the loss function.

Fig. 1 Proposed explainable backorder prediction approach
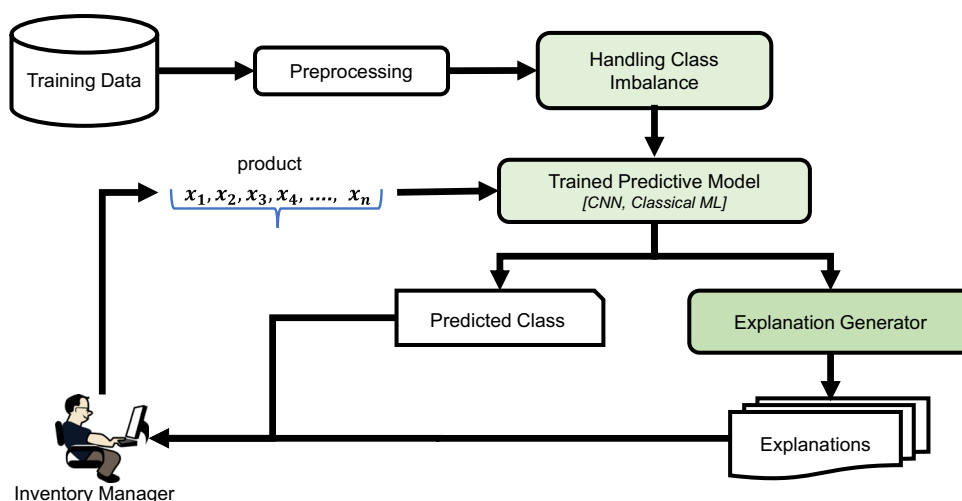
The complexity of the model can be defined by $\Omega(g)$. LIME is useful to explain specific decision predicted by the model (i.e., local prediction).

Explainable product backorder prediction

The overview of our proposed explainable product backorder prediction framework is depicted in Fig. 1. We first apply preprocessing step to handle the missing values, converting qualitative variables into quantitative ones and normalizing the values in a similar range. Next, we apply our proposed convolutional neural network-based backorder prediction model to classify the product. Finally, we introduce explainable AI techniques to explain both global, model agnostic aspects as well as individual decisions with the intent to make the inventory manager understand better why his or her backorder prediction system acts as it does.

Preprocessing and feature analysis

In our dataset, each particular sample has 21 different features/attributes including current inventory, lead time, forecasting for a different time, sales performance, different risk flags. The details of the dataset are presented in Section 5.1.1. The value of different features is varied widely among binary, quantitative, qualitative, and categorical. In this step, all the feature values are transformed into a real number. The missing values are handled by filling them in with the median of other samples' values. A normalization technique is then applied to convert each feature value into a certain range [0,1]. Here, we applied the most widely recognized MinMax normalization technique.

However, a dataset having highly correlated features is not suitable for applying classification methods. We investigate to see whether any high correlated features are available, exploiting the Pearson correlation coefficient measure for this purpose. According to the findings, we observe that there are no features with a high correlation ($\rho > .80$). Hence, the dataset should now be suitable for our purpose of backorder predictions.

Handling class imbalance with ADASYN

As we noted earlier, a product backorder scenario is a rare event that leads the dataset to be extremely imbalanced. Therefore, we employed one of the efficient synthetic oversampling methods, ADASYN (Adaptive Synthetic Oversampling) (He et al., 2008) to balance the dataset. Considering the difficulty level of learning, ADASYN generates synthetic minority class examples utilizing the weighted distribution. ADASYN focused on generating more synthetic minority class examples for those minority samples that are harder to classify. Given a training dataset, D_{train} with N number of samples where each sample is denoted as x, y, the vector x is represented by a K dimensional vector containing different attributes of an ordered product and y is the binary value that indicates the label (0 for non-backordered and 1 for backordered one).

Let m_{min} and m_{maj} be the number of examples of minority class and majority class, respectively such that $m_{min} + m_{maj} = N$, and in this backorder prediction task $m_{min} << m_{maj}$. ADASYN oversampling techniques generate synthetic minority class examples to balance the dataset according the algorithm illustrated in Algorithm 1.

It first calculates the degree of imbalance d and then, depending on the tolerated imbalance ratio, computes the number G that denotes the number of synthetic minority class examples needed to be generated. Here $\beta \in [0, 1]$ indicates the desired bleaching ratio, $\beta = 1$

Algorithm 1 ADASYN: Adaptive Synthetic Oversampling

1: **procedure** ADASYN_OVERSAMPLING(D_{train})
2:     $d = \frac{m_{min}}{m_{maj}}$ ▷ Finding the imbalance ratio
3:     **if** $d < d_{th}$ **then**
4:         $G = (m_{maj} - m_{min}) \times \beta$ ▷ Number of synthetic examples need to generate
5:         **for** each $x_i \in X_{min}$ **do**
6:             $N_{kn} = $ K_Nearest_Neighbors(x_i)
7:             $r_i = \Delta_i / K, \ i = 1, ..., m_{min}$
8:         **end for**
9:         Normalize r_i such that $\hat{r}_i = r_i / \sum_{i=1}^{m_{min}} r_i$
10:        $g_i = \hat{r}_i \times G$ ▷ Number of synthetic example per sample x_i
11:        **for** 1 to **do** g_i
12:            $x_{zi} = Random\_choice(X_{min})$
13:            $s_i = x_i + (x_{xi} - x_i) \times \lambda$
14:        **end for**
15:     **end if**
16:

indicated that the dataset will be fully balanced. For each minority example x_i, ADASYN then calculate the ratio r_i applying K-nearest neighbors with Euclidean distance, where Δ_i is the number of nearest neighbors of x_i. Using the normalized ratio $\hat{r}_i$, then it computes the number of synthetic examples for each minority examples x_i. Finally it generates the synthetic minority class examples applying the distance vector and the random number λ.

Convolutional neural network-based prediction model

Inspired by the success of the convolutional neural network (CNN)-based models in computer vision, natural language processing and other classification tasks, we proposed a 1-dimensional CNN classifier to predict product backorder in advance. The structure of our proposed CNN-based predictive model is illustrated in Fig. 2.

Our CNN-based predictive model has two convolutional hidden layers with batch normalization, max-pooling, and dropout layers. To extract unique and low-level features, the max-pooling layers are exploited. In addition, max-pooling makes the computation faster by reducing the dimension and parameters (Wu and Gu, 2015). Moreover, it reduces the variance. Then we utilized one flattened layer followed by three dense layers with dropout layers. To overcome the over-fitting problem, dropout layers are applied to randomly drop some neurons in the training process for regularization (Kingma et al., 2015; Srivastava, 2013). The parameters and activation functions in different layers of convolutional neural networks are summarized in Table 3. In the convolutional layers and all hidden dense layers, we employed the *Relu* (Ramachandran et al., 2017) activation function. Finally,

Sigmoid (Ramachandran et al., 2017) activation function is applied in the output layer.

Experiments and evaluation

Dataset collection and evaluation metrics

This section presents the details of dataset that is leveraged to conduct experiments using our proposed method. We also present a brief discussion about the evaluation metrics considered to measure and validate the performance.

Dataset

We carried out a wide range of experiments to validate the performance of our methods on a publicly available benchmark dataset called "*Can you Predict Product Backorder*[1]"The dataset has an 8 weeks inventory of historical data. The brief statistical summary of the dataset is depicted in Table 4.

The numerical figures in Table 4 illustrate that the number of backordered (positive) samples is much lower than the number of non-backordered (negative) samples. Hence, the ratio (1:137) indicates that this dataset is an extremely imbalanced one. For a better understanding of why this is a challenging problem, we illustrated the distribution of backordered (positive) and non-backordered (negative) samples using a doughnut chart in Fig. 3. There are 22 features for each sample and the attributes/features include current inventory, transit time, quantity, forecasting, and different

---

Fig. 2 Structure of our proposed convolutional neural network-based backorder prediction model
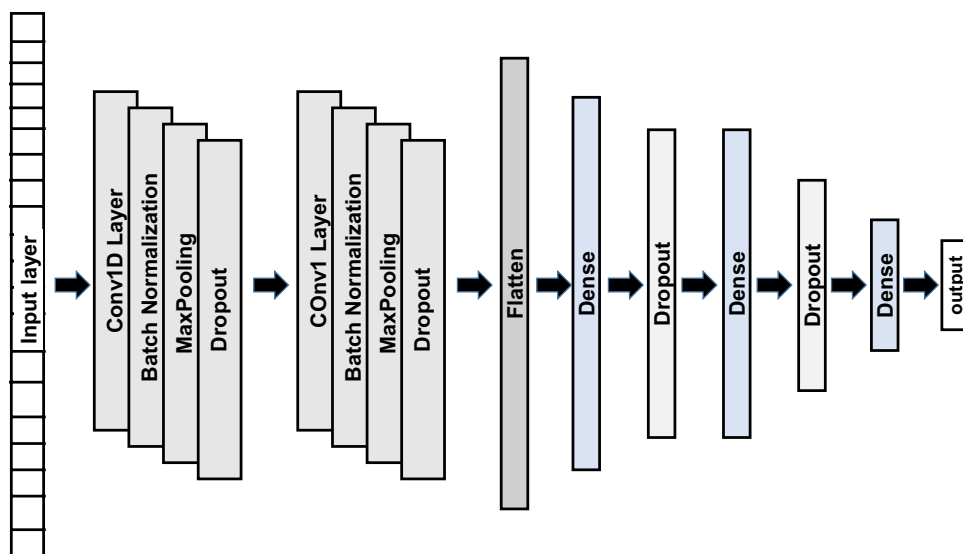
Table 3 The summary of different layers with parameters and activation functions

SL	Layer	Input/Output	Activation
1.	Conv1D	(20,32)	*Relu*
2.	Batch Normalization	(20,32)	–
3.	Max Pooling	(10,32), stride=2	–
4.	Dropout	(10,32)	–
5.	Cov1D	(9,64)	*Relu*
6.	Batch Normalization	(9,64)	–
7.	Max Pooling	(4,64), strid=2	–
8.	Dropout	(4,64)	–
9.	Flatten	–	–
10.	Dense	64	*Relu*
11.	Dropout	64	–
12.	Dense	32	*Relu*
13.	Dropout	32	–
14.	Dense	1	*Sigmoid*

risk flag. The list of features with a brief description is depicted in Table 5.

Evaluation metrics

Generally, the performance of any classification method is measured based on the common evaluation metrics including *accuracy*, *precision*, *recall* and f_1-*score*. The confusion metrix is used to compute those metrics. However, the backorder prediction dataset is extremely class imbalanced, and the above mentioned evaluation metrics are not enough to validate the performance of

Table 4 Brief statistical summary of the dataset

No of samples	No of positive samples	No negative samples	Imbalance ratio
1,929,936	13,981	1,915,954	1:137

Fig. 3 Distribution of backordered and non-backordered samples
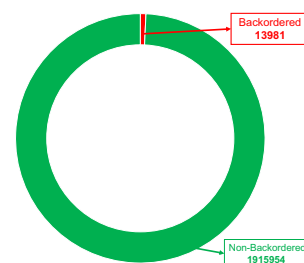
any classifier on a imbalanced dataset. Therefore, we employed *accuracy*, *AUC* (**A**rea **U**nder the **C**urve) and *ROC* (**R**eceiver **O**perating **C**haracteristics) curves to measure and visualize the performance of our proposed backorder prediction method. The accuracy score is calculated by using the measures from confusion metrics as follows:

$$Acc = \frac{tp + tn}{tp + fp + fn + tn}, \qquad (4)$$

where *tp*, *fp*, *fn* and *tn* denote the number of classified samples as true positive, false positive, false negative and true negative, respectively.

AUC is one of the most efficient metrics to measure the performance of any classification model on imbalanced data. The *AUC* is calculated as follows:

Table 5 Description of different features/attributes of a particular order†

Feature f_i	Explanation
Current Inventory f_1	Current inventory level of component
Lead Time f_2	Registered transit time
Transit Quantity f_3	Product amount in transit
Sale Forecasts f_4, f_5, f_6	Forecasting stock amount for upcoming 1, 3, 6, 9 months
Sales f_7, f_8, f_9, f_{10}	Amount of sold product in last 1, 3, 6, 9 months
Reco. Stock Amount f_{11}	Recommended amount (amount) in stock
Overdue f_{12}	Overdue parts from source
Performance f_{13}, f_{14}	Performance of the product in last 6 months and 12 months
Stock Overdue f_{15}	Overdue Stock amount for orders
Risks $f_{16}, f_{17}, f_{18}, f_{19}, f_{20}, f_{21}$	Different general risk flags
Label Y	Product went on backorder or not

$$AUC = \frac{1 + P - F}{2}, \tag{5}$$

where P is the precision of the classifier and F is the false positive rate. The details of the these metrics can be found elsewhere in the published study by Chawla et.al (2002) and de Santis et al. (2017).

Prediction performance

We conducted a wide range of experiments with multiple settings to illustrate the performance of our backorder prediction methods. Since our major goal in this study is to introduce the explainability of in backorder prediction, we first applied classical machine learning and a deep neural network-based classification approach. Then, we exploit XAI technologies (SHAP and LIME) to explain the model's priorities and individual prediction. Classifiers from classical machine learning including decision tree, support vector machine, gradient boosting, etc., were applied. All experimental settings can be broadly classified into three different types based on the chosen dataset balancing strategy, classical ML, and deep learning. In all experimental setups, we applied two different dataset balancing techniques ADASYIN and SMOTE (Chawla et.al, 2002). Based on the predictive models, we report the experimental results in two categories, classical and deep classifiers.

The prediction performance of all experimental setups using classical machine learning is presented in Table 6. The results conclude that the ASASYN balancing technique is more efficient and achieved higher accuracy as well as AUC than SMOTE in most of the experimental setups. In turn, we can conclude that for the backorder prediction task, our introduced ADASYN balancing technique would be a better choice to implement any real-time backorder prediction system. Among all five different classical machine learning models, the gradient boosting (XGBoost) classifier achieved higher accuracy. On the other hand, in terms of

the most effective and important evaluation metric, AUC, support vector machine performs better than other models. In addition, other classification models including decision tree, SVM, and KNN also achieved effective performance in backorder prediction except for Gaussian Naive Bayes.

The experimental setup for deep learning techniques can be classified based on the parameters. The experiments are conducted by training CNN-based models with different settings. Two types of CNN models are applied. One has max-pooling layers and the other does not. The models were trained using two different epoch sizes which are 50 and 100. The performance of all experimental settings is presented in Table 7.

From the results, we can see that the convolutional neural network-based model with max-pooling layer (MxCNN_100 and MxCNN_50) performed better among other experimental settings. It can also be concluded that our introduced ADASYN data balancing technique achieved efficient performance in both evaluation metrics. We added dropout layers that randomly drop some neurons in the training process for regularisation to overcome the

Table 6 Performance of classical machine learning models in terms of *accuracy* and *AUC*

| Predictive model	Balancing tech.	Accuracy	AUC
Decision Tree	ADASYN	0.9366	0.8134
	SMOTE	0.9338	0.8111
Support Vector Machine	ADASYN	0.8511	**0.8711**
	SMOTE	0.8455	0.8696
Gradient Boosting	ADASYN	**0.9548**	0.8298
	SMOTE	0.9538	0.8288
Gaussian Naive Bayes	ADASYN	0.7947	0.8153
	SMOTE	0.7836	0.8142
K-nearest Neighbor	ADASYN	0.8970	0.8498
	SMOTE	0.8977	0.8507

The best result is in **bold**

Table 7 Performance of CNN-based models in terms of *accuracy* and *AUC*

Predictive model	Balancing tech.	Accuracy	AUC
CNN_50	ADASYN	0.8756	0.9443
	SMOTE	0.8936	0.9425
CNN_100	ADASYN	0.8868	0.9460
	SMOTE	0.8938	0.9432
MxCNN_50	ADASYN	**0.8947**	0.9475
	SMOTE	0.8938	0.9432
MxCNN_100	ADASYN	0.8903	**0.9489**
	SMOTE	0.8877	0.9462

The best result is in **bold**

over-fitting problem. To illustrate the necessity of dropout layers in CNN-based models, we carried out experiments with and without dropout layers. The performance based on the evaluation metrics concludes that dropout layers overcome the over-fitting problem. The MxCNN model without dropout layers achieved accuracy and AUC in the training data of 0.9081 and 0.9651, respectively. On the other hand, for testing data, the performance is lower in terms of both metrics. Without dropout layers, the performance of the model on test data based on accuracy and AUC are 0.8792 and 0.9411, respectively. Though the performance difference (2.89% in accuracy and 2.4% in AUC) between the training and testing data is not that big but still it has over-fitting. The method with dropout layers got the training accuracy and AUC of 0.8843 and 0.9499, respectively. For test data, the performance is quite consistent with accuracy and AUC of 0.8903 and 0.9489, respectively. Thus, we can say that the inclusion of dropout layers overcomes the over-fitting problem and eventually increases the performance.

As compared to the performance of classical machine learning models reported in Table 6, the prediction power of our proposed CNN-based approach is way higher than the performance of ML methods. Although classical machine learning classifiers achieved higher *accuracy*, our CNN based model achieved a huge improvement in predicting future backorder prediction in terms of *AUC*, which is a more important metric to judge the performance of a classifier on data imbalance, because higher accuracy alone might not guarantee the predictive power of a classifier in case of extreme data imbalance. The performance of our method is also depicted by the Receiver Operating Characteristic curve (ROC curve) in Fig. 4. The curve illustrated the performance of our predictive model as compared to a random classifier. The area within the green curve shows the higher AUC achieved in predicting product backorders.
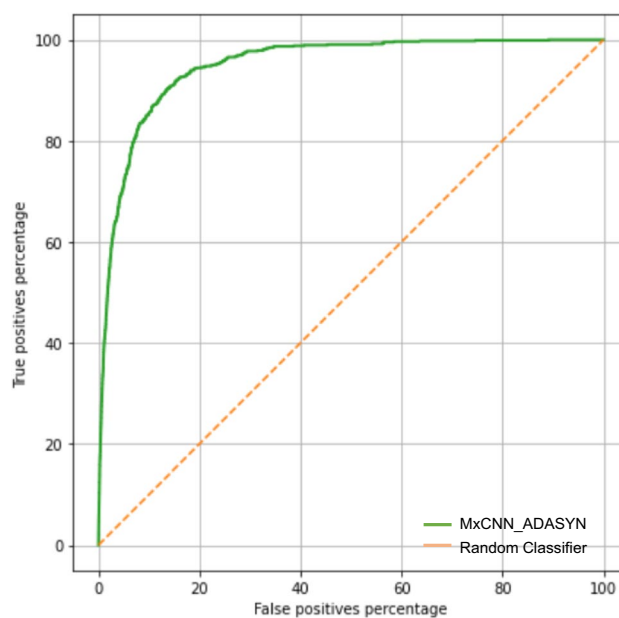
Fig. 4 Performance of convolutional neural network based predictive model in terms of Receiver Operating Characteristic curve (ROC curve). The X-axis and Y-axis indicate the false positive and true positive percentage, respectively

Performance comparison with state-of-the-art methods

The performance comparison of our proposed backorder prediction model with existing state-of-the-art methods is presented in Table 8. We directly reported the performance in terms of accuracy and AUC from existing published papers. Some existing works reported the performance only in terms of AUC but did not use accuracy and some others did the opposite. The blank cells (i.e., "-") in the table indicate that the performance based on particular evaluation metric is not reported in the published paper. According to the performance of different state-of-the-art methods reported in the table, we can see that our CNN-based predictive model outperformed the known related works in terms of both evaluation metrics except for one method by Shajalal et al. (2021). In turn, our methods significantly outperformed all methods based on accuracy. Shajalal et al. (2021) applied a deep neural network with SMOTE oversampling technique. They applied four different variants of their methods utilising oversampling and under sampling techniques. Compared to the performance of those methods, our model got the best performance except one. Though one of their methods achieved a higher performance in terms of AUC, the performance difference with our method is subtle. In addition, their method lacks global interpretability and local explainability.

Table 8 Performance comparison of our method with known related work on the same dataset in terms of *accuracy* and *AUC*

Method	Technique	Accuracy	AUC
Our Method	ADASYN + CNN	**0.8947**	**0.9489**
Ntakolia et al. (2021)	NN (MLP)	0.8568	0.9200
Shajalal et al. (2021)	SMOTE_DNN	–	0.9586
	Weighted_DNN	–	0.9350
	Ran_Over_DNN	–	0.9475
Islam and Amin (2020)	DRF	0.8436	0.7870
	GMB	0.7919	0.7950
Hajek and Abedin (2020)	RF	–	0.9157

The performance of our method is in **bold**

Islam and Amin (2020) applied a distributed random forest (DRF) and gradient boosting machine (GBM) classifier to model product backorder. The performance of their models is struggling compared to the CNN-based model in terms of both evaluation metrics. Another noticeable concern in the performance of their method is substantial over-fitting. Numerically, their training accuracy of 0.9835 is way higher than the testing accuracy of 0.8436. Another work by Hajek and Abedin (2020) applied classical machine learning classifiers to model product backorder prediction. From their applied ML classifiers, random forest (RF) achieved the best AUC, which is still lower than our method. Note that they did not report the accuracy in the paper. Similar to Shajalal et al. (2021); Ntakolia et al. (2021) proposed a multi-layer perceptron (MLP) based neural network (NN) for modelling product backorder. But their performance is still much lower than ours in terms of both evaluation metrics. We think adding ADASYN oversampling technique overcome the data imbalance problem better. With this, our convolutional neural network-based predictive model capture the product backorder more efficiently as compared to other state-of-the-art methods. Having this comparative analysis, we can conclude that our method has got a new state-of-the-art performance in predicting product backorder in the inventory management system.

Explaining backorder prediction model

This section presents the explainability of our introduced XAI techniques to interpret and/or explain the overall model and particular decisions of the proposed backorder prediction model. We first present the global model agnostic explanations generated to interpret the overall model's prediction priorities. Then the local explanations for a certain prediction are presented to provide the overall insight to understand a certain product will be going to be backordered or not.

Explaining overall model's priority

To interpret and explain the overall model, we exploit Shapely Additive values (Shap Values) that highlight the overall features' contributions in predicting the model's decisions. The feature contributions for the best performing model are depicted in Fig. 5 and 6.

We can see both figures indicate the top ten most important features that the prediction model has given higher importance, such as current inventory, transit quantity, lead time, performance in the last 12 and 6 months, and sales. We can conclude that these are the top 10 most important features on which the model depends more to predict whether a product is going to be backordered or not.

Explaining individual predictions

The features described in the previous figures (Fig. 5 and 6) have overall high importance in the predictive models. But it is expected that every sample (order) is different and unique in terms of features' values. Therefore, the importance and contributions of different features also will be different for particular order. To identify the most contributing features of each order, we employed local interpretable model agnostic explanation (LIME) to explain individual predictions. Using LIME, we trained a surrogate model with a portion of training data that mimics the performance and decision-making priorities of the proposed backorder prediction model. The explanation using LIME is depicted in Figs. 7 and 8. These are the explanations for two individual backordered samples.

The labels of these two products are 1 (backordered) and the model also predicted the same. The features in the right side marked by Yellow color pushed the predictive model to classify as backordered, while Blue colored features in the left side did the opposite. From Fig. 7, we can see that the probability for being classified as backordered and non-backordered is 66% and 34%, respectively. The figure also indicates that the most important features (features in the right side) that lead to the prediction as backordered are local_bo_qty, current_inventory, and sales in the last 1 and 3 months and a risk flag. On the other hand, features (features in the left side) like lead time, in_transit_quantity, performance in the last 6 and 12 months, etc. try to push the model to predict a product as non-backordered. However, for another backordered sample, we can see that the list of contributed features for the backorder decision is different than the previous one. In Fig. 8, features such as lead time contributed the most to pushing the model to decide as a backordered one, which was the opposite for the previous sample.

To explain the local individual predictions more transparently, we applied shap values to plot the explanation as a force plot. Figures 9 and 10 illustrate the explanations for two different backordered samples. In both figures, the predictions from

Fig. 5 Global interpretation of the features' contributions of backorder prediction model as summary plot
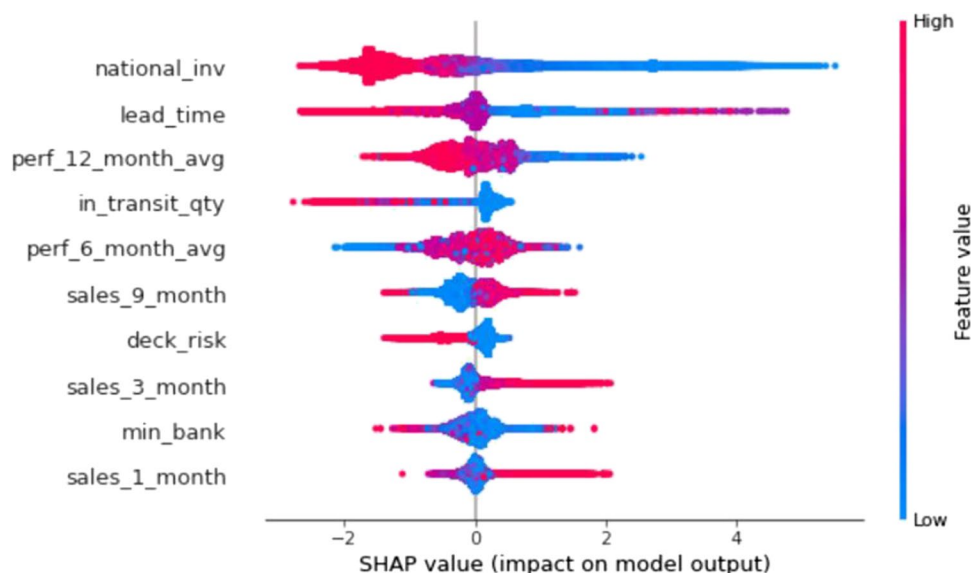
Fig. 6 Global interpretation of the features' contributions of backorder prediction model as bar chart
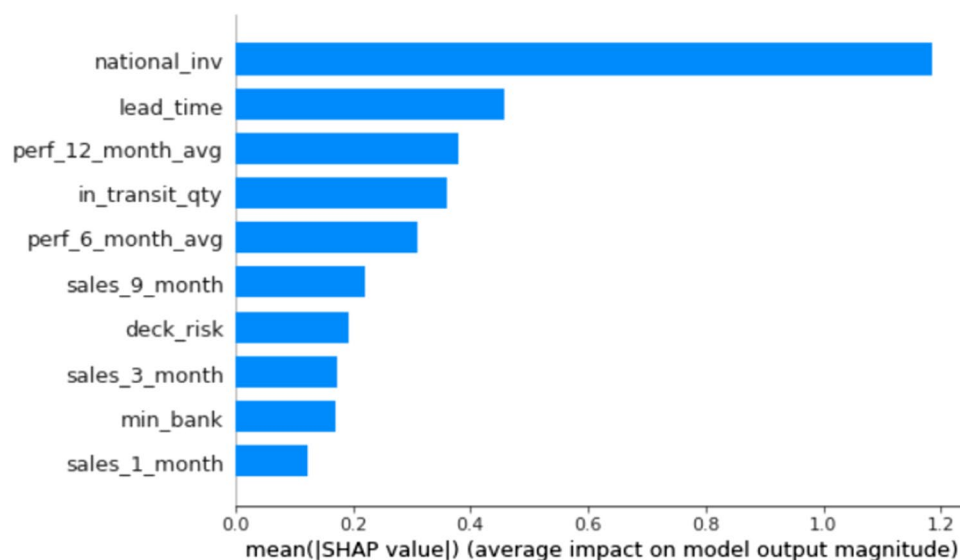
the models, referred to as base values are **0.67** and **0.75**, respectively for both samples. The closer the value is to 1, the more the prediction leans toward backordered, while the closer to 0, the decision will then predicted as non-backordered. The red marked features contributed to increase the base value that help to decide the sample as a backordered one, and blue marked features did the opposite. The features having more impact on the base value remain closer to the boundary. For example, the two most-contributing features that push the model to decide the samples as backordered are current inventory and per_6_months for the first force plot (Fig. 9). For another example, the features with the most impact are current inventory and the forecasting for 9 months (Fig. 10). The explanations for the same two samples' decisions are also presented using the waterfall plot in Figs. 11 and 12. The red marked features contributed

to predicting the sample backordered and the blue colored try to push the classifier to predict the sample as non-backordered. Here the number and the span indicate the level of contributions of the features towards the decision.

With the help of our approach, stakeholders without in-depth knowledge of how backorder prediction systems work can have a better understanding both in terms of how the models generally factor in different types of data for making their decisions, as well as be enabled to analyse concrete decisions (that might seem counter intuititve or risky) in more depth than is possible with existing approaches. By applying such visualisations in practice, stakeholders would thus be enabled to enact suggestions from AI based systems more competently, and adapt their business strategies and decisions accordingly. This has the potential to both improve the
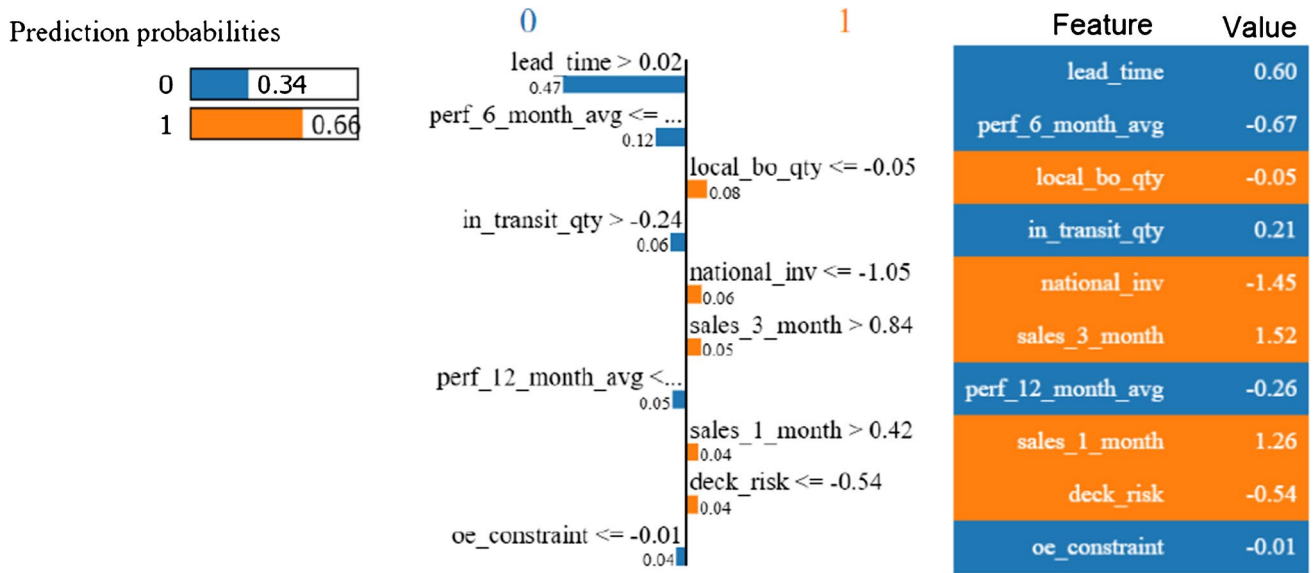
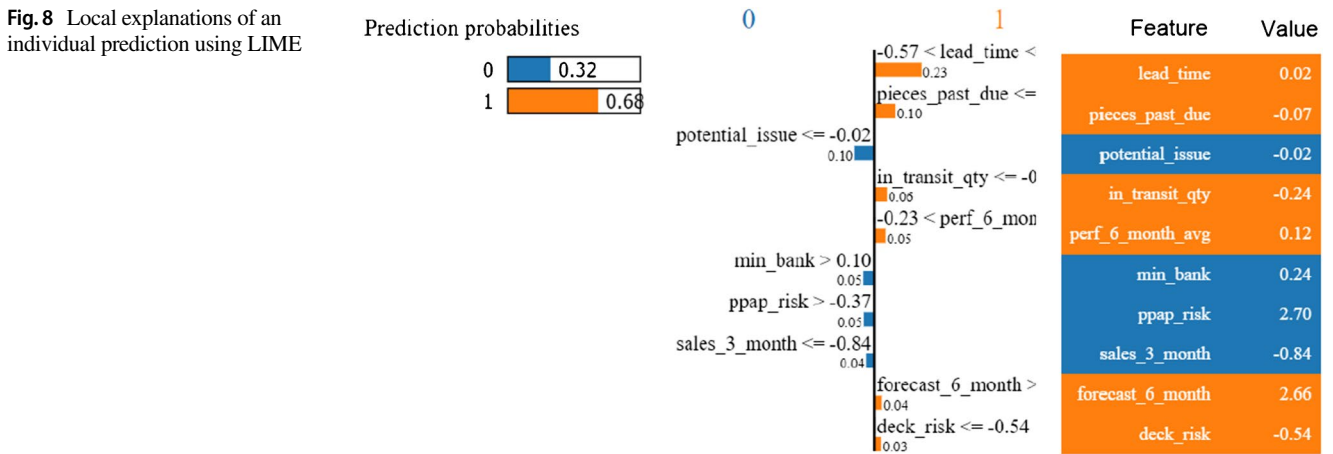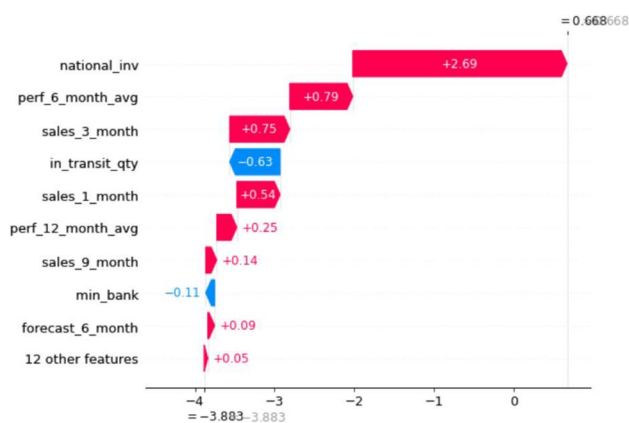**Fig. 7** Local explanations of an individual prediction using LIME

Fig. 8 Local explanations of an individual prediction using LIME

Fig. 9 Local explanations of an individual prediction using force plot
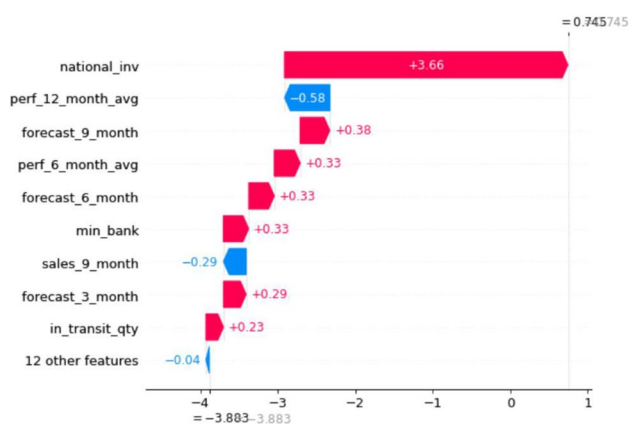

Fig. 10 Local explanations of an individual prediction using force plot

Fig. 11 Local explanations of an individual prediction using waterfall plot

Fig. 12 Local explanations of an individual prediction using waterfall plot

usefulness, as well as also the willingness to adopt such systems in practice. While our introduced explainabilty has still to be evaluated with users, its is not trivial to implement such systems in practice. In this regard, our paper contributes a demonstration of the applicability, and shows how such techniques can be implemented in a way that provides value to other stakeholders than developers of machine learning systems, to whose such applications are currently targeted.

Conclusion & future directions

This paper proposed a novel CNN-based model for product backorder prediction in an inventory management system and introduced global and local explainability that can explain the overall model decision-making priorities and answer the "*why*" question regarding any specific prediction. First, we proposed a novel convolutional neural network-based prediction model incorporating ADASYN oversampling technique to address data imbalance

problem. The performance carrying out diverse experiment setups concluded that our proposed CNN-based backorder prediction model achieved a new state-of-the art result in product backorder prediction. In addition, the performance comparison with some known related methods demonstrated that our methods outperformed others in terms of multiple evaluation metrics. Secondly, our model is not only able to predict the backordered item but also can explain the reasons why the model predicts that a product is going to be backordered. For doing so, we utilised existing successful XAI techniques, SHAP and LIME, to explain the overall predictive model and individual decisions. Using global explanations, the stakeholder, and inventory managers can have an idea and understanding of how the overall model is making the decision. On the other hand, they can explicitly know and analyse why a certain product has a high chance to be backordered in the future, leveraging the explanations for their business decisions. Hence, they can identify which attributes have the most impact on a particular decision, and then react by adapting controllable attributes (i.e. current inventory, lead time, etc.). Therefore, even when our approach still needs to be evaluated in practice, we believe these explanations can help the stakeholders to make their decision and minimize future losses. Most importantly, these explanations can increase the trust, transparency, and accountability of the AI-based predictive models in business problems, thus helping to overcome limitations of existing approaches that are more like black boxes for the users. While our study demonstrated the applicability of XAI techniques in the business domain on the concrete example of backorder predictions, there are multiple application areas such as customer churn prediction, customer behavior prediction, credit-worthiness assessment, fraud detection etc. where our explainable predictive model can be applied.

In the future, we plan to develop a collaborative interface to represent the explanations so that people can understand the decision-making more efficiently. We are also planning to introduce counterfactual explanations to provide a clear understanding about what are the possible actions she needs to take into account in the future.

References

Abedin, B., Klier, M., Meske, C., & Rabhi, F. (2022) Introduction to the minitrack on explainable artificial intelligence (XAI). *Proceedings of the 55th Hawaii International Conference on System Sciences,* 1–2. http://hdl.handle.net/10125/70765

Adadi, A., & Berrada, M. (2018). Peeking inside the black-box: A survey on explainable artificial intelligence (xai). *IEEE Access, 6,* 52138–52160. https://doi.org/10.1109/ACCESS.2018.2870052

Alvarez-Melis, D., & Jaakkola, T.S. (2017) A causal framework for explaining the predictions of black-box sequence-to-sequence models. *Arxiv.* https://doi.org/10.48550/arXiv.1707.01943

Arya, V., Bellamy, R. K., Chen, P.-Y., Dhurandhar, A., Hind, M., Hoffman, S. C., Houde, S., Liao, Q. V., Luss, R., Mojsilović, A., Mourad, S., Pedemonte, P., Raghavendra, R., Richards, J., Sattigeri, P., Shanmugam, K., Singh, M., Varshney, K. R., Wei, D., & Zhang, Y. (2019). One explanation does not fit all: A toolkit and taxonomy of ai explainability techniques. *Arxiv.* https://doi.org/10.48550/arXiv.1909.03012

Bartoletti, I. (2019). AI in healthcare: Ethical and privacy challenges. In: D. Riaño, S. Wilk, A. ten Teije (Eds.), *Artificial Intelligence in Medicine* (vol. 11526, pp. 7–10). AIME 2019. Lecture Notes in Computer Science. Springer. https://doi.org/10.1007/978-3-030-21642-9_2

Bawack, R. E., Wamba, S. F., Carillo, K. D. A., & Akter, S. (2022). Artificial intelligence in e-commerce: A bibliometric study and literature review. *Electronic Markets, 32*(1), 1–42. https://doi.org/10.1007/s12525-022-00537-z

Błaszczyński, J., & Stefanowski, J. (2015). Neighbourhood sampling in bagging for imbalanced data. *Neurocomputing, 150,* 529–542. https://doi.org/10.1016/j.neucom.2014.07.064

Bussmann, N., Giudici, P., Marinelli, D., & Papenbrock, J. (2021). Explainable machine learning in credit risk management. *Computational Economics, 57*(1), 203–216. https://doi.org/10.1016/j.ins.2019.05.042

Cao, Y., Geddes, T. A., Yang, J. Y. H., & Yang, P. (2020). Ensemble deep learning in bioinformatics. *Nature Machine Intelligence, 2*(9), 500–508. https://doi.org/10.1038/s42256-020-0217-y

Carcillo, F., Le Borgne, Y.-A., Caelen, O., Kessaci, Y., Oblé, F., & Bontempi, G. (2021). Combining unsupervised and supervised learning in credit card fraud detection. *Information sciences, 557,* 317–331. https://doi.org/10.1016/j.ins.2019.05.042

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research, 16,* 321–357. https://doi.org/10.5555/1622407.1622416

Cliff, D., Brown, D., & Treleaven, P. (2011). *Technology trends in the financial markets: A 2020 vision.* UK Government Office for Science. http://www.bis.gov.uk/assets/bispartners/foresight/docs/computer-trading/11-1222-dr3-technology-trends-in-financial-markets.pdf

de Santis, R. B., de Aguiar, E. P., & Goliatt, L. (2017). Predicting material backorders in inventory management using machine learning. *2017 IEEE Latin American Conference on Computational Intelligence (LA-CCI),* pp. 1–6. https://doi.org/10.1109/LA-CCI.2017.8285684

Došilović, F. K., Brčić, M., & Hlupić, N. (2018). Explainable artificial intelligence: A survey. *41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)* pp. 0210–0215. https://doi.org/10.23919/MIPRO.2018.8400040

Förster, M., Klier, M., Kluge, K., & Sigler, I. (2020). Fostering human agency: A process for the design of user-centric XAI systems. *41st International Conference on Information Systems (ICIS).* https://aisel.aisnet.org/icis2020/hci_artintel/hci_artintel/12

Hajek, P., & Abedin, M. Z. (2020). A profit function-maximizing inventory backorder prediction system using big data analytics. *IEEE Access, 8,* 58982–58994. https://doi.org/10.1109/ACCESS.2020.2983118

He, H., Bai, Y., Garcia, E. A., & Li, S. (2008). Adasyn: Adaptive synthetic sampling approach for imbalanced learning. *2008 IEEE International Joint Conference on Neural Networks. IEEE World Congress On Computational Intelligence,* pp. 1322–1328. https://doi.org/10.1016/j.ins.2019.05.042

Hussain, W., Merigó, J. M., & Raza, M. R. (2021). Predictive intelligence using anfisinduced owawa for complex stock market prediction. *International Journal of Intelligent Systems.* https://doi.org/10.1002/int.22732

Hussain, W., Gao, H., Raza, M. R., Rabhi, F. A., & Merigo, J. M. (2022a). Assessing cloud QoS predictions using OWA in neural network methods. *Neural Computing and Applications, 34,* 1–18. https://doi.org/10.1007/s00521-022-07297-z

Hussain, W., Merigó, J. M., Raza, M. R., & Gao, H. (2022b). A new QoS prediction model using hybrid IOWA-ANFIS with fuzzy C-means, subtractive clustering and grid partitioning. *Information Sciences, 584,* 280–300. https://doi.org/10.1016/j.ins.2021.10.054

Islam, S., & Amin, S. H. (2020). Prediction of probable backorder scenarios in the supply chain using distributed random forest and gradient boosting machine learning techniques. *Journal of Big Data, 7*(1), 1–22. https://doi.org/10.1186/s40537-020-00345-2

Janiesch, C., Zschech, P., & Heinrich, K. (2021). Machine learning and deep learning. *Electronic Markets, 31*(3), 685–695. https://doi.org/10.1007/s12525-021-00475-2

Jiao, L., & Zhao, J. (2019). A survey on the new generation of deep learning in image processing. *IEEE Access, 7,* 172231–172263. https://doi.org/10.1109/ACCESS.2019.2956508

Kingma, D. P., Salimans, T., & Welling, M. (2015). Variational dropout and the local reparameterization trick. *Advances in neural information processing systems,* vol 28. https://doi.org/10.48550/arXiv.1506.02557

Ładyyżyński, P., Żbikowski, K., & Gawrysiak, P. (2019). Direct marketing campaigns in retail banking with the use of deep learning and random forests. *Expert Systems with Applications, 134,* 28–35. https://doi.org/10.1016/j.eswa.2019.05.020

Lawal, S., & Akintola, K. (2021). A product backorder predictive model using recurrent neural network. *IRE Journals, 4*(8).

Li, Y. (2017). *Backorder prediction using machine learning for danish craft beer breweries.* [PhD disseration, Aalborg University].

Li, Y., Huang, C., Ding, L., Li, Z., Pan, Y., & Gao, X. (2019). Deep learning in bioinformatics: Introduction, application, and perspective in the big data era. *Methods, 166,* 4–21. https://doi.org/10.1016/j.ymeth.2019.04.008

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in neural information processing systems,* vol 30. https://doi.org/10.48550/arXiv.1705.07874

Meske, C., Bunde, E., Schneider, J., & Gersch, M. (2022). Explainable artificial intelligence: Objectives, stakeholders, and future research opportunities. *Information Systems Management, 39*(1), 53–63. https://doi.org/10.1080/10580530.2020.1849465

Moscato, V., Picariello, A., & Sperlí, G. (2021). A benchmark of machine learning approaches for credit score prediction. *Expert Systems with Applications, 165,* 113986. https://doi.org/10.1016/j.eswa.2020.113986

Nowak, A. S., & Radzik, T. (1994). The shapley value for n-person games in generalized characteristic function form. *Games and Economic Behavior, 6*(1), 150–161. https://doi.org/10.1006/game.1994.1008

Ntakolia, C., Kokkotiis, C., Moustakidis, S., & Papageorgiou, E. (2021). An explainable machine learning pipeline for backorder prediction in inventory management systems. *25th Pan-Hellenic Conference on Informatics,* pp. 229–234. https://doi.org/10.1145/3503823.3503866

Ntakolia, C., Kokkotis, C., Karlsson, P., & Moustakidis, S. (2021). An explainable machine learning model for material backorder prediction in inventory management. *Sensors, 21*(23), 7926. https://doi.org/10.3390/s21237926

Panesar, A. (2019). *Machine learning and AI for healthcare* (pp. 1–407). Springer. https://doi.org/10.1007/978-1-4842-3799-1

Ramachandran, P., Zoph, B., & Le, Q. V. (2017). Searching for activation functions. *Arxiv*.https://doi.org/10.48550/arXiv.1710.05941

Randhawa, K., Loo, C. K., Seera, M., Lim, C. P., & Nandi, A. K. (2018). Credit card fraud detection using adaboost and majority voting. *IEEE Access, 6*, 14277–14284. https://doi.org/10.1109/ACCESS.2018.2806420

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). Why should I trust you? Explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1135–1144. https://doi.org/10.1145/2939672.2939778

Saraogi, G., Gupta, D., Sharma, L., & Rana, A. (2021). An un-supervised approach for backorder prediction using deep autoencoder. *Recent Advances in Computer Science and Communications Formerly: Recent Patents on Computer Science, 14*(2), 500–511. http://doi.org/10.2174/2213275912666190819112609

Shajalal, M., Abedin, M. Z., & Uddin, M. M. (n.d.). *Handling class imbalance data in business domain. The essentials of machine learning in finance and accounting* (pp. 199-210). Routledge.

Shajalal, M., Hajek, P., & Abedin, M. Z. (2021). Product backorder prediction using deep neural network on imbalanced data. *International Journal of Production Research*, 1–18. https://doi.org/10.1080/00207543.2021.1901153

Thiebes, S., Lins, S., & Sunyaev, A. (2021). Trustworthy artificial intelligence. *Electronic Markets, 31*(2), 447–464. https://doi.org/10.1007/s12525-020-00441-4

Srivastava, N. (2013). *Improving neural networks with dropout*. [Thesis, University of Toronto, 182(566), 7].

Wu, H., & Gu, X. (2015). Max-pooling dropout for regularization of convolutional neural networks. *International Conference on Neural Information Processing*, pp 46–54. https://doi.org/10.48550/arXiv.1512.01400