Tilburg University

VAT DETECTION ON INCOMING INVOICES USING ERP DATA: A MACHINE LEARNING APPROACH

ALBERT ALETRINO

thesis submitted in partial fulfillment
of the requirements for the degree of
master of science in data science & society
at the school of humanities and digital sciences
of tilburg university

word count: 8,785

STUDENT NUMBER

2082876

COMMITTEE

Boris Čule
Michal Klincewicz

LOCATION

Tilburg University
School of Humanities and Digital Sciences
Department of Cognitive Science &
Artificial Intelligence
Tilburg, The Netherlands

DATE

May 20, 2022

ACKNOWLEDGMENTS

I wish to thank my colleagues and supervisor at KPMG, René Duijkers, for their expertise and support. Additionally, I would like to thank my supervisor from the Tilburg University, Boris Čule, for his guidance throughout the thesis. Lastly, I thank my parents for their unconditional support throughout my academic career.

VAT DETECTION ON INCOMING INVOICES USING ERP DATA: A MACHINE LEARNING APPROACH

ALBERT ALETRINO

Abstract

Companies often have the feeling they do not reclaim all possible VAT. Therefore, they seek help from external consultants and tax advisors to optimise their VAT reclamation process. As these third parties have to go through all VAT-free deemed invoices manually to look for reclaimable VAT, they seek methods to automatically classify invoices. This research aims to investigate whether various machine learning methods are able to identify VAT on incoming invoices which were previously deemed VAT-free. Following the findings of Freitas (2004) and Mori and Uchihira (2019), we propose to use techniques along the lines of the accuracy interpretability trade-off stating that models become more accurate, the more complex they get. Therefore, we have tested a decision tree, a random forest, a gradient boosting, and explainable boosting classifier using invoices and metadata from the ERP system from a multinational active in the chemical industry. This research proposes two novelties. The first one is the identification of VAT instead of the identification of line items, while the second one is classifying invoices based on ERP data instead of attempting to extract features from the invoice itself. On the basis of several threshold and ranking metrics, we found the random forest classifier to perform best.

DATA SOURCE/CODE/ETHICS STATEMENT

Work on this thesis did not involve collecting data from human participants or animals. The original owner of the data and code used in this thesis retains ownership of the data and code during and after the completion of this thesis. The author of this thesis acknowledges that they do not have any legal claim to this data or code. The code used in this thesis is available on GitHub when given access by the administrator: https://github.com/albertaletrino/thesis-Aletrino.

1 INTRODUCTION

The processing of invoices can be a very tiresome task for companies, especially on the receiving end of them. AP (Accounts Payable) clerks have to manually read the invoice and input the information into the ERP (Enterprise Resource Planning) system (Van Loo et al., 2015). This is especially difficult regarding the tax code determination as AP clerks will have to possess some basic knowledge of VAT (Value Added Tax) regulations. This makes VAT recovery a very cumbersome task. Tax consultants, therefore, offer their services to check for VAT recovery potential (Horsthuis et al., 2020). Although they do possess the required knowledge, the process is still executed manually. Therefore, the question is raised whether this undertaking can be automated.

Contemporary literature emphasises an upcoming awareness of VAT compliance (Lahann et al., 2019) and VAT reclamation optimisation (Horsthuis et al., 2020). This VAT reclamation works as follows: companies are allowed to reclaim their VAT over their business expenses (Belastingdienst, 2020). Incoming invoices are categorised into two classes: VAT-reclaimable invoices and VAT-irreclaimable invoices. The latter constitutes primarily of invoices which are deemed VAT-free. However, oftentimes, companies have the feeling invoices belonging to the first class, accidentally fall into the second class due to a non-functioning OCR (Optical Character Recognition) system or because of manual failure of AP clerks. Therefore, they hire the expertise of external tax consultants to check whether the invoices which were deemed VAT-free actually do have VAT stated on them as this offers them a one-off cash opportunity. Since the majority of invoices have been processed correctly, most invoices sent to tax advisors are indeed VAT-free. However, this causes tax advisors to go through correctly assessed invoices unnecessarily. Therefore, tax advisors would benefit from a system which could process VAT-free deemed invoices and predict whether the client has assessed the invoice correctly. Using data from the ERP system, one is able to identify VAT on incoming invoices employing several machine learning methods. This constitutes a binary classification problem which categorises invoices into two classes: VAT-free and VAT-carrying invoices. Hence, the following research question is constructed:

RQ *"To what extent are machine learning techniques able to identify whether VAT is actually present on incoming invoices that have been deemed VAT-free?"*

Only incoming invoices are assessed as only the VAT on the accounts payable side of operations is able to be reclaimed. Following the views of Freitas (2004) and Mori and Uchihira (2019), a decision tree, random forest, gradient boosting, and explainable boosting classifier have been chosen based on the accuracy interpretability trade-off. Additionally, these will be evaluated based on several threshold and ranking metrics. Therefore, the following sub research questions have been drawn up:

sub-RQ1 *"What is the performance of a decision tree classifier to identify whether VAT is stated on incoming invoices in terms of threshold and ranking metrics?"*

sub-RQ2 *"What is the performance of a random forest classifier to identify whether VAT is stated on incoming invoices in terms of threshold and ranking metrics?"*

sub-RQ3 *"What is the performance of a gradient boosting classifier to identify whether VAT is stated on incoming invoices in terms of threshold and ranking metrics?"*

sub-RQ4 *"What is the performance of a explainable boosting classifier to identify whether VAT is stated on incoming invoices in terms of threshold and ranking metrics?"*

Note that the adopted sub-research questions are solely focused on the comparison of performance between models. However, the required steps for a data science research strategy are set forth in the experimental setup section.

This research aims to investigate whether machine learning methods are able to identify VAT on incoming invoices previously deemed VAT-free in addition to finding out which method performs best. As we only asses those invoices which, at first glance, have been deemed VAT-free, the data constitutes of a huge class imbalance. To identify whether VAT is states on these invoices, we turn to ERP data which constitutes of information of the line items per good or service delivered, which was manually inputted by AP clerks. Classifiers were assessed relative to a no-skill classifier on the basis of several threshold and ranking metrics. Threshold metrics constituted of precision, recall, and F2-scores, while ranking metrics included of ROC (Receiver Operator Characteristic) curves, PR (Precision Recall) curves as well as AUC (Area Under the Curve) scores. Against the expectations, the random forest classifier was deemed to perform best relative to the other classifiers.

The remainder of this paper is structured as follows: section 2 will describe prior research on the optimisation of the VAT reclamation and the current status of invoice classification methods. Section 3 and 4 will describe the employed methods as well as give an explanation of the general flow of the research and its evaluation methods. Thereafter, the general findings of this research will be presented in section 5. Next, section 6 will set forth a discussion where the findings of this research will be critically examined. In here, practical implications, limitations, and recommendations for future research will be given as well. Lastly, section 7 will close with a brief conclusion.

2 RELATED WORK

This section will give a brief overview of the relevant work in current literature. First, a short introduction into the workings of the VAT and VAT analytics in general will be given, after which we will touch briefly on the function of an ERP system. Lastly, the progress on invoice classification will be discussed in more detail.

2.1 *Value Added Tax*

VAT is an indirect tax designed to tax the consumption of goods and services. However, the tax is not directly levied at consumers, but levied at every player along the supply chain (Goossenearts et al., 2009). Tax authorities make a distinction between a taxable person and a person who is tax due. A taxable person can be any player along the supply chain, e.g. a supplier company, while a person who is tax due is always the consumer. This implies that every transaction between two suppliers is VAT due, while a consumer does not have to be at the other end of the transaction. Subsequently, suppliers are allowed to reclaim their paid VAT over business expenses in most cases (Belastingdienst, 2022). However, Pijnenburg et al. (2017) claim it is detrimental for tax administrations to become more efficient as the workload is expanding, while staff is reduced and budget is cut. Davenport and Harris (2007) propose a solution in which companies should invest and double down on analytics in order to generate a competitive advantage. Although the competitive advantage by gaining insights into your VAT reclamation does not seem clear cut, gaining an better understanding of your internal processes is always valuable practice in addition to earning a one-off cash opportunity. In fact, if you reclaim too much, this is fraudulent, if you reclaim too little, you miss out on income. VAT declaration fraud annually leads to billions of losses for EU member states, with the VAT gap being approximately 9.6% in 2020 (European Commission, 2020). Keen (2007) explains that this is partly due to the way this indirect tax is set up. By creating a system in which VAT can be reclaimed from the tax authorities, governments give way to criminals who can abuse this.

2.2 *VAT reclamation*

Until now, there has not been done a lot of academic research into the optimisation of VAT reclamation. Van den Biggelaar et al. (2008) conclude that companies lack the willingness to exploit their ERP systems, while simultaneously have a large risk exposure regarding VAT. They state a combination of expert tax knowledge and ERP technical knowledge is needed to remedy those risks. However, this research argues the opposite. When VAT can be recognised on invoices without human intervention, only expert tax knowledge is needed to be able to determine whether the VAT stated is actually reclaimable. In addition, tax authorities are doubling down on data analytics as well as making them increasingly better in performing VAT audits (Van Loo et al., 2015). It is therefore even more important companies have their own VAT processes aligned with regulations. Lahann et al. (2019) confirm this issue and propose various machine learning methods to increase VAT compliance. Using ERP data, the authors trained a classifier to predict tax rates based on related voucher information of journal reports. This research builds on the work of Lahann et al. (2019) by employing similar data to determine whether VAT is present on invoices.

2.3 *ERP management*

This research aims to classify invoices based on ERP data. Therefore, a small light is shed on the definition and workings of ERP systems. Enterprise Resource Planning systems generally have the function to automate and integrate a majority of business functions for companies through the use of various software modules (Davenport, 1998). Examples of automation and integration are the sharing, accessing, and practicing of data and information. The author identifies the defining feature of an ERP system as the integration of different organisational functions. This means that data only has to be entered once, making it available for the complete organisation to observe and use.

2.4 *Invoice classification*

Lastly, the current state of invoice classification will be discussed. Bartoli et al. (2010) classify invoices based on the nature of information they contain. The authors do this based on low-level features they extracted from the invoices themselves. They find their performance to rely solely on the employed classifier and selection of features used. Sorio et al. (2010) employ a similar approach, only this time they feed the classifier instances one at a time, providing an online learning approach. They find a support vector machine classifier to perform just as well as other classifiers in a closed world setting, i.e. a batch learning approach. The current state-of-the-art technique for the identification of invoices is OCR (Optical Character Recognition) technology. Tarawneh et al. (2019) propose a method employing OCR technology in which they classify invoices into three categories: handwritten, machine-printed, or regular receipts. To do so, they employ a random forest, k-nearest neighbours, and naïve Bayes approach and find the k-nearest neighbours classifier to perform best based on accuracy. Larsson and Segeras (2016) employed a similar OCR approach to recognise and classify line items on invoices.

On the other end, their have been efforts to optimise the invoice management process by implementing a standard electronic invoice format. In Italy, this has been mandatory for Italian companies since January 2019. Bardelli et al. (2020) have implemented a multiclass classification approach to classify invoices into account and VAT codes. They found random forests to outperform boosting techniques on the subject matter. Khan et al. (2020) employed neural nets to predict posting parameter on incoming invoices, while Hamza et al. (2009) used a special type of neural network, an incremental growing neural gas, to classify documents into eight distinct classes. Additionally, Freitas (2004) and Mori and Uchihira (2019) describe a trade-off between the accuracy and interpretability of machine learning models. When deciding which algorithm suits your problem best, one should take this trade-off into account. Therefore, we will elaborated on this further in the methods section. These works show the progress academic research has made regarding the classification of incoming invoices. This research attempts two novelties. The first one is the prediction of VAT on invoices, instead of regular line item classification. Although the latter is much more important in determining whether VAT is actually deductible, technology has not yet been proven capable of replacing an expert tax advisor. Therefore, this research takes a step back and only attempts to recognise VAT such that a tax expert can determine its deductibility. The second novelty will be making a classification based on features found in ERP data. Current research has mainly focused on features extracted from the invoices itself, while a rich database is usually already present in the companies themselves.

3 METHODS

In the following section, the choice for the employed algorithms will be laid out. Similarly, the choice to leave out specific algorithms will be substantiated. Thereafter, the inner workings and mathematical foundations of the chosen algorithms will be described.

3.1 *Classification*

This research aims to predict whether incoming invoices have VAT stated on them using various machine learning techniques, making it a classical binary classification problem: invoices either fall into a class with VAT or into a class without VAT. Unlike linear regression, a classification problem is characterised by the fact that the response variable is qualitative, rather than quantitative (James et al., 2013, p. 129). This implies that the machine learning methods should be chosen such that they support a binary or multiclass outcome. Additionally, as the findings of this research will be used by tax advisors to further optimise their processes, there is a need for the model to be sufficiently interpretable, i.e. comprehensible. Mercado et al. (2016) argue that comprehensibility stems from the degree to which a model is transparent. Therefore, we follow the definition of transparency of Chen et al. (2014) stating that agent transparency is an interface's quality to make an intelligent agent's intent, performance, future plans, and reasoning process comprehensible for an operator. In general, there has been a development amongst artificial intelligence practitioners to make their algorithms more comprehensible (Miller, 2019). In the XAI (explainable artificial intelligence) domain, it is hypothesised that the greater the level of transparency in the algorithm, the more likely one is to trust the outcome (Mercado et al., 2016; Miller, 2019). With respect to this research, tax advisors are more likely to trust and, therefore, use a machine learning technique in their daily practice when it is founded upon transparent and legally-binding rules. This is partly due to their fiduciary responsibility in their agent-principal relationship (Jensen, 1983). Therefore, this research chooses to opt for machine learning algorithms based on the accuracy versus comprehensibility trade-off, meaning that it is attempted to optimise the accuracy of several machine learning model while keeping the nature of the algorithm as understandable as possible (Freitas, 2004; Mori & Uchihira, 2019). The accuracy comprehensibility trade-off works as follows: on the x-axis, we have the comprehensibility, while the y-axis represents the accuracy. In the bottom right corner, we have the easiest to understand algorithms with poor performances, while the top left corner contains difficult to understand black-box models with very high performances. An imaginary line can be drawn between these two corners and we will choose algorithms which are situated along this line.

First, a simple decision tree will be used to classify the invoices. Although its workings are easy to understand, the model will probably perform poorly. Secondly, a gradient boosting classifier will be used to tackle our problem. This model already comes closer to a black-box model, which inner workings are more difficult to understand but probably will perform better. Thirdly, a random forest classifier, which sits in between the two previously mentioned classifiers will be employed as well. Additionally, a explainable boosting classifier will be added, which is a state-of-the-art classifier which resides in the upper right corner of the accuracy comprehensibility axis, meaning its both accurate as well as interpretable.

3.2 *Alternatives*

One other important factor to keep in mind are characteristics of each classifier. These characteristics determine whether a certain classifier would be fit for the prediction in this specific research. Classifiers which were considered but left out were a support vector machine, a naïve Bayes classifier, and a k-nearest neighbours approach. In the following, we will briefly touch upon the reasons for disregarding these classifier.

First, a support vector machine struggles with its performance when handling big datasets (Cervantes et al., 2020). As this research aims to aid large multinationals with reducing its invoice load, one should take into account that such companies sometimes process thousands of invoices on a daily basis. Therefore, when considering future use, support vector machines do not seem fit. Secondly, a naïve Bayes classifier assumes all features to be independent, which rarely happens in real life. Additionally, such a classifier struggles with categorical data unseen in the training data, yet present in the test data (Zhang, 2004). This is not uncommon for invoices received from all different kinds of suppliers, selling all different kinds of products and services. Lastly, a k-nearest neighbours approach is computationally very expensive and decreases in speed with the size of the data set. In addition, its performance decreases when dealing with imbalanced data. In the following section, this will be explained in more detail. Additionally, it is difficult to determine the optimal of value k and misspecifying this will lead to either under- or overfitting (Jadhav & Channe, 2016).

3.3 *Mathematical foundations*

In the following section, the workings as well as mathematical foundations of the employed classifiers will be given. The algorithms are ordered ascending in complexity.

3.3.1 *Decision tree*

Decision tree learning is a machine learning method in which a decision tree is made up from a set of training instances where a split is induced based upon a certain decision criterion (Su & Zhang, 2006). Its simplified workings are displayed in figure 1. A decision tree's splitting criterion can either be a given feature of the training set or based upon an actual criterion such as gini impurity or entropy (Shalev-Shwartz & Ben-David, 2014, p. 250). The splitting is continued until all instances at the end of a node have the same target label. This process is called recursive partitioning. It is claimed to be one of the most widespread and successful learning methods due to four main characteristics: its simple and understandable nature, the fact that no parameters are needed, and the ability to handle data of various types. Decision tree learning is described as a greedy, top-down, and recursive learning process as the complete training data is used to train upon an empty tree (Su & Zhang, 2006).

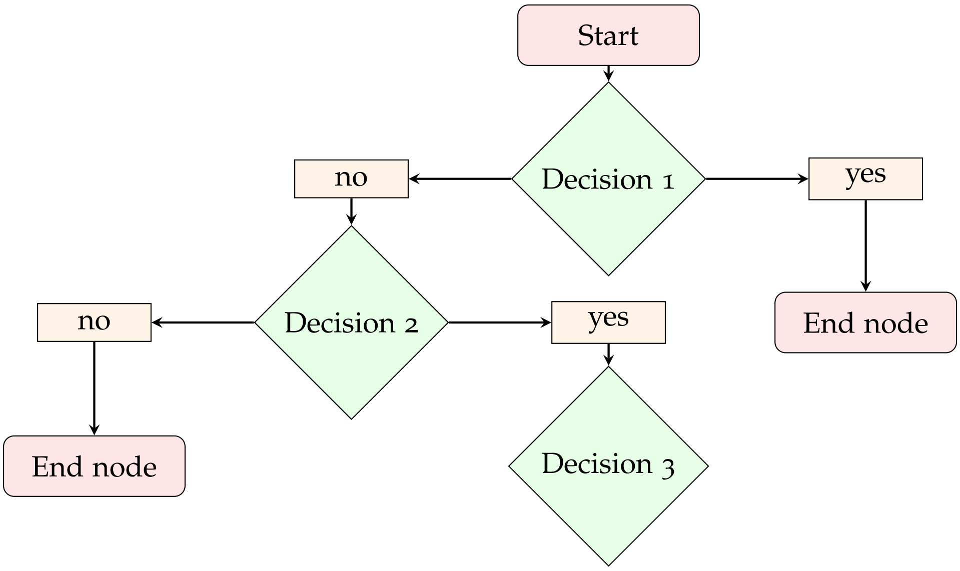

Figure 1: Simplified visualisation of the workings of a decision tree classifier

3.3.2 *Random forest*

A random forest classifier stems from the decision tree learning method. In decision tree learning, trees can have the tendency to overfit to the training data because they keep splitting until all instances from a subset at a node belong to the same target class (Shalev-Shwartz & Ben-David, 2014, p. 250). A random forest classifier solves this by averaging multiple deep decision trees, all trained on a different subset of the training data (Hastie et al., 2009, p. 587). On the other hand, this also implies that the comprehensibility of the combined trees decreases in addition to an increase in bias. However, this is often accepted as the performance increases drastically.

3.3.3 *Gradient booster*

A booster is simply a general method to boost the accuracy of any given machine learning method (Schapire, 1999). In fact, boosting has been one of the most powerful additions to the domain of machine learning in the last twenty years (Hastie et al., 2009, p. 356). A regular boosting method entails combining the outputs of several weak classifiers to come together as one powerful committee. This is displayed in figure 2. Weak classifiers are defined as classifiers whose error rate is only slightly better than random guessing. When sequentially applying weak classifiers to modified versions of the data, one essentially boosts the data until it has reached the desired accuracy, thereby producing a sequence of weak classifiers G_m *(x), m = 1,2,. . . ,M* (Hastie et al., 2009, p. 356). Next, the predictions of all the weak classifiers are combined through a weighted majority vote:

$$G(x) = sign(\sum_{m=1}^{M} \alpha_m G_m(x)) \tag{1}$$

Here, the weak classifiers are displayed as a series of rounds *m = 1,. . . ,M*. *G(m)* represents a hypothesis for the weight each classifier should receive, with weak learners receiving a higher weight such that the algorithm focuses more on the difficult examples. α_m are parameters chosen by the boosting algorithm and measures the importance of *G*. A gradient booster differs from a regular boosting method by adding an optimisation of an arbitrary loss function.

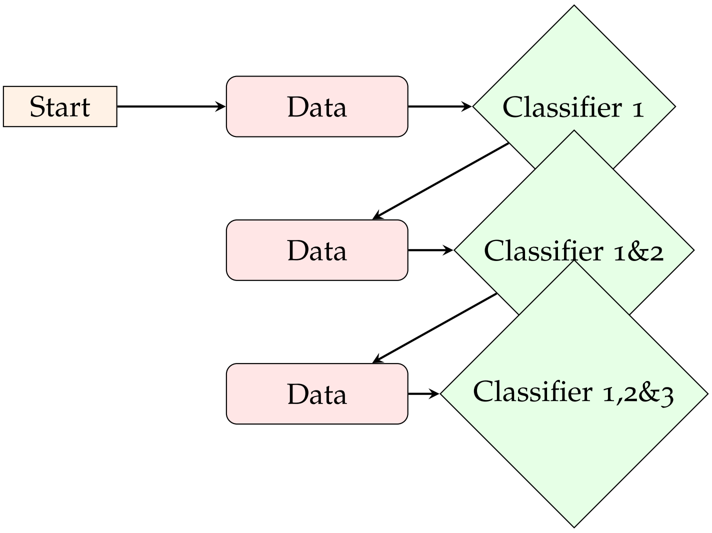

Figure 2: Simplified visualisation of the workings of a gradient boosting classifier

3.3.4 *Explainable booster*

An explainable boosting method builds on the basics of the AdaBoost method of Schapire (1999), yet is able maintain the perfect trade-off being intelligibility and comprehensibility (Nori et al., 2019). It is a state-of-the-art glassbox model which looks like the following generalised additive model:

$$g(E[y]) = \beta_0 + \sum f_j(x_j) \tag{2}$$

Where β_0 depicts a standard coefficient. f_j is a regular feature function, which is, in this case, bagging or gradient boosting. g represents the link function which combines a regular boosting function to a regression or classification. Additionally, an explainable boosting classifier is able to increase accuracy while maintaining comprehensibility using the following equation:

$$g(E[y]) = \beta_0 + \sum f_j(x_j) + \sum f_{ij}(x_i, x_j) \tag{3}$$

Lastly, the reason why an explainable boosting classifier is so easy to understand is because one is able to plot the contribution of each feature to the final prediction by plotting f_j. This is due to the fact that an explainable boosting classifier is an additive model and therefore, each feature, adds to the performance of the model (Nori et al., 2019).

4 EXPERIMENTAL SETUP

The following section gives a detailed description of the multifaceted approach to this classification problem. First, the data set and complementary exploratory data analysis will be discussed, before the processing of the data will be described. Afterwards, a section on the evaluation metrics and criteria will be given. The complete processing and analysis of the data has been conducted using Python 3.5 in a JupyterLab environment.

4.1 *Data description*

The data represents a portion of the incoming invoices of a multinational active in the chemical industry which were deemed VAT-free. 4,595 invoices have been assessed of which 153 contain VAT and 4,418 do not contain VAT. These invoices are entered per line item in the ERP system meaning that the initial 4,595 invoices lead to a total of 55,425 line items of which 1,177 contain VAT and 54,248 do not contain VAT, which constitutes a huge class imbalance. The data is complemented by ERP data with metadata regarding the line items which was gathered by AP clerks of the firm. As the objective of this research is to identify all invoices which contain VAT, line items containing no VAT that are situated on an invoice which does contain VAT were- given a positive label as well. To tackle the class imbalance, stratified sampling is employed to set aside a test set consisting of 20% of the complete sample which resembles the complete data set in terms of class distribution.

The ERP data constitutes of 68 mostly categorical features, only the document values constitute a numerical value, and is presented in table 6, Appendix A. Invoices are sent from 23 different countries and received in 44 different countries. Goods and services posted on the line items are subdivided into 256 different product groups. There are 54,360 credit invoices of which 1,111 contain VAT, while there are only 1,065 debit invoices of which 66 contain VAT. Additionally, SIC codes are added to the suppliers to group companies per industry they are active in. SIC codes were retrieved from the Eikon Datastream database and constitute 4 digits where each digit is a subcategory from the digit to its left. In our sample, there are 446 unique SIC codes which originate from 9 different industries and 83 different subindustries.

4.2 *Process*

In figure 3, the workflow of this research is displayed. This can be subdivided into six components: preprocessing, feature engineering, feature selection, over- and undersampling, hyperparameter tuning, evaluation, and error analyses.

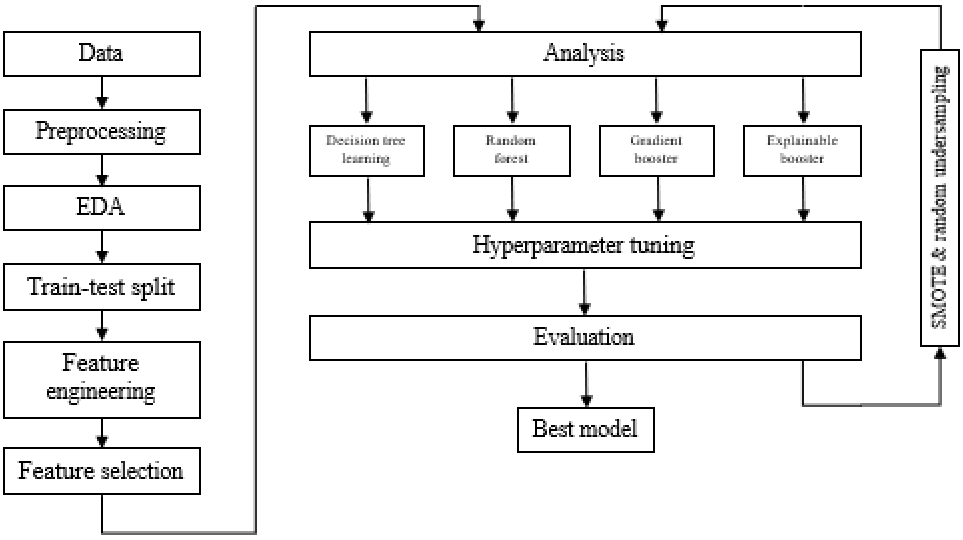

Figure 3: General overview of the workflow of the research

First, the preprocessing of the data and feature engineering is conducted. Incomplete variables are imputed with the mode of that specific variable and all categorical features are binarised in order to be able to work with them. As mentioned previously, additional features are created to categorise companies into their first, second, and third digit of their SIC code. Additionally, a 'crosscountryborder' feature is added which states whether a good or service has been delivered in a country other than the country of supply. Next, a Pearson's Chiˆ2 test is used for feature selection to determine which features should be entered into the model. This test is chosen as it tests for independence between categorical variables (Kuhn & Johnson, 2019). Ultimately, one can remove those features from the data set that prove to be independent of the target variable. The scikit-learn library SelectKBest has been used to execute this test and the benchmark is set at 10, such that all variables falling below this threshold will be removed. The model is then left with 9 features which are presented and explained in table 7, Appendix B.

Secondly, to remedy the class imbalance, SMOTE (Synthetic Minority Oversampling Technique) is used to oversample the minority class (Chawla, 2002). This is done in an effort to make the algorithm learn from as much instances as possible without adding too much 'fake' information to the model. The technique works by selecting a few instances from the minority class and generate its k-nearest neighbours inside the minority class. A line is drawn between the original instance and its k-nearest neighbour. Subsequently, a new synthetic instance is created along that line and added to the feature space (He & Ma, 2013, p.47). Additionally, Chawla (2002) describes that a combination of SMOTE oversampling and random under-sampling will generate the best performance. After testing various configurations, the best results were generated when the minority class was increase up to 10% of the complete sample and the majority class was trimmed down to 30% of its initial size. This is done keeping in mind that not too much synthetic data should be added by oversampling and not too much information should be lost by undersampling.

Next, the aforementioned algorithms are run using scikit-learn packages for the decision tree, random forest, and gradient booster. An InterpretML package is used to run the explainable booster. Additionally, a K-fold cross validation approach is employed on the training data. This statistical method involves randomly leaving a fold out of the training data which can then be used as validation (James et al., 2013, p. 183). The choice for k is set rather arbitrarily. Therefore, the choice for k is set to 10, which is common practice in the domain of machine learning as it provides a good trade-off between computational efficiency and bias (Kuhn & Johnson, 2013, p. 70).

Lastly, grid searches are used for hyperparameter tuning to find the optimal configuration of each model in combination with a 5-fold cross validation. However, as the hyperparameters and corresponding values can lead to a very high number of possibilities, only a few hyperparameters per algorithm will be chosen to be optimised. In fact, hyperparameter tuning is often warranted and not recommended due to this very reason (Bergsta et al., 2011). However, Montavoni et al. (2018) prove that the more complex the data gets, the more an algorithm will benefit from hyperparameter tuning. The hyperparameters that were chosen to be optimised are presented in table 1. This eventually came down to the configurations displayed in table 8 in Appendix C.

Table 1: Optimised hyperparameters per algorithm and number of fits needed

DTC	RFC	GBC	EBC
criterion min_samples_split min_samples_leaf	max features min_samples_split min_samples_leaf n_estimators	learning_rate min_samples_split min_samples_leaf n_estimators	learning_rate interactions min_samples_leaf
150	450	720	135

DTC: Decision Tree Classifier, RFC: Random Forest Classifier, GBC: Gradient Boosting Classifier, EBC: Explainable Boosting Classifier

4.3 Evaluation

In order to gain valuable insights from the analyses, it is of the utmost importance to take a look at how one should evaluate the models (Sun et al., 2009). Additionally, to assess the performance of our classifiers, we will compare them to a no-skill classifier which consists of a dummy classifier employing a stratified strategy.

Accuracy is traditionally the leading measure to evaluate classification problems. However, due to the huge class imbalance, accuracy does not contain that much of an added value in this case. For example, even a simple majority baseline would reach an accuracy of 99% in our data set. Therefore, one must look for other evaluation measures. Ferri et al. (2009) propose a distinction of evaluation measures: threshold metrics, which give the evaluator a qualitative understanding of error and ranking metrics, which give an impression of the algorithm's ability to separate classes. Thresholds metrics constitute, amongst others, the accuracy, precision, recall and Fβ-score, while the ranking metrics contain the ROC (Receiver Operating Characteristic) curve, the PR (Precision Recall) curve, and the AUC (Area Under the Curve). The authors propose a third evaluation metric as well, namely probabilistic metrics. However, as these are not in line with the models of this research, these metrics are disregarded.

Threshold metrics quantify the prediction errors of the classification algorithms. As discussed prior, the accuracy is the ratio of correctly predicted instances versus all predictions made displayed in equation 4.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{4}$$

where TP denotes the true positives, TN the true negatives, FP the false positives, and FN the false negatives. Positive and negative relate to the positive minority and negative majority class. The precision score displays the ratio of correctly identified instances of the relevant class versus the instances that were identified as being relevant, while recall shows the ratio of correctly identified instances of the relevant class versus all the relevant (positive) instances that are actually in the sample. Both metrics are displayed by equations 5 and 6.

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

The Fβ-score represents the mean between these two metrics and is displayed by equation 7. Keeping the objective of this research in mind, it is more costly to identify an invoice containing VAT as VAT-free than the other way around. Therefore, recall is given additional weight in the Fβ-score, such that we will assess our findings based on an F2-score instead of an F1-score, which is the harmonic mean between precision and recall.

$$F\beta - score = \frac{(1+\beta^2) * (Precision * Recall)}{\beta^2 * Precision * Recall} \tag{7}$$

Ranking metrics, on the other hand, do not quantify the prediction error but evaluate classifiers based on their ability to separate classes (Ferri et al., 2009). In addition, ranking metrics, and ROC analysis specifically, have the property to deal with class imbalance as they are not affected by models that are biased towards the minority class at the expense of the majority class (He & Ma, 2013, p. 27). An ROC curve is actually nothing more than a plot between the false positive rate on the x-axis and the true positive rate on the y-axis, with a diagonal going from the bottom left to the top right representing a no-skill classifier. Every point on the plot left of this diagonal is an increase in performance, with the top left corner of the plot representing a perfect-skill classifier. Similar to the ROC curve, one can plot the PR-curve, with the recall on the x-axis and precision on the y-axis. This time, the no-skill classifier is denoted by a horizontal line and every point above this line is seen as an increase in performance with the top-right corner denoting a perfect-skill classifier. One can calculate the AUC's of these plots to express the performance of a specific classifier in a number.

5 RESULTS

The following section is dedicated to the description and discussion of the results of this research. The aim of this research is to test whether specific classifiers are able to identify VAT on incoming invoices. In order to do so, four sub-questions have been drawn up to answer the main research question. This section aims to answer these four sub-questions by first assessing the classifiers on the basis of threshold metrics, after which the same will be done for the ranking metrics. Additionally, each algorithm is assessed relative to a no-skill classifier and relative to one another.

5.1 *Threshold metrics*

Table 2 shows the results of the threshold metrics of the four different classifiers. Column 2 shows the performance of the no-skill classifier in terms of accuracy, precision, recall, and F2-score. Recall that the no-skill classifier was a dummy classifier employing a stratified strategy meaning the classifier randomly samples one-hot encoded vectors from a multinomial distribution parametrised by the empirical class probabilities (Pedregosa et al., 2011). Additionally, the F2-score was chosen because this puts more weight on recall as false negative are more costly, i.e. predicting a VAT-carrying invoice as VAT-free should be avoided at all costs. The results show a very high accuracy of 96.0% which is common for a classification problem with such a huge class imbalance. Therefore, we have to take a look at the precision, recall, and F2-scores. However, these are 1.9%. 1.7%, and 1.7%, respectively, showing an extreme lack in skill of the dummy classifier to correctly identify invoices as VAT-free or VAT-carrying.

Columns 3 to 6 in table 2 present the performance of the examined classifiers. The decision tree, which is shown in column 3, has an accuracy of 90.6% meaning that around 9 out of 10 invoices are classified correctly. This is not a particularly good score when keeping the class imbalance in mind. However, when taking a deeper look into its performance by assessing the ratio of correctly specified positive instances versus all relevant instances, a recall of 78.9% is observed, which is the second highest recall of all the employed algorithms. Combined with a precision of 15.8%, this results in an F2-score of 43.8%.

According to the accuracy comprehensibility trade-off, one would expect a classifier's performance to increase in its complexity (Freitas, 2004; Mori & Uchihira, 2019). However, this is not really applicable to the algorithms used in this research. The accuracy of the decision tree, random forest, and gradient booster – shown in columns 3-5 - are all somewhat equal around 90-91%. The explainable booster outperforms these classifiers in terms of accuracy as its accuracy is 95.6%, shown in column 6. All classifiers show to have high recall, which is good as this proves the classifiers' ability to correctly identify the positive instances. This is especially true for the decision tree and random forest, which report recall scores of 78.9% (column 3) and 82.6% (column 4), respectively. However, this is at the expense of precision – which denotes a classifier's ability to identify positive instances among the retrieved instances - as all classifiers report a much lower precision. These are actually pretty solid findings as one should interpret them as follows: the decision tree is able to identify 78.9% of positive instances – denoted by its recall in column 3 – in addition to have 15.8% of retrieved instances being positive. Assuming that the VAT is normally distributed amongst the invoices, this means that around 80% of all VAT is found while simultaneously having only to check 1 in 6[1] invoices to find a VAT-carrying invoice.

When assessing the models based on their F2-scores, we observe the gradient booster in column 5 to have performed worst. In line with Freitas (2014) and Mori and Uchihira (2019), it is expected that more complex models generate higher F2-scores. However, this is not the what we observe in our case as the decision tree and random forest have similar F2-scores, 43.8% and 44.3% shown in column 3 and 4, respectively, while the random forest is more complex than the decision tree. In fact, the most complex model - the explainable booster - performs only slightly better than both of them with an F2-score of 34.8%. On top of that, the gradient booster – third in terms of complexity – performs worst of all with an F2-score of 34.8%. All in all, when assessing recall and precision, we can conclude that the random forest is in fact the best classifier amongst these four algorithms. However, when only assessing F2-scores, the explainable booster performs best.

Table 2: Performance per classifier including no-skill classifier

	No-skill	DTC	RFC	GBC	EBC
Accuracy	0.960	0.906	0.901	0.910	0.956
Precision	0.019	0.158	0.156	0.137	0.259
Recall	0.017	0.789	0.826	0.586	0.579
F2	0.017	0.438	0.443	0.348	0.462

DTC: Decision Tree Classifier, RFC: Random Forest Classifier, GBC: Gradient Boosting Classifier, EBC: Explainable Boosting Classifier

[1] 100%/15.8%

Table 3: AUC per classifier for ROC and PR curves

	No-skill	DTC	RFC	GBC	EBC
ROC	0.500	0.697	0.864	0.625	0.666
PR	0.000	0.281	0.195	0.230	0.192

DTC: Decision Tree Classifier, RFC: Random Forest Classifier, GBC: Gradient Boosting Classifier, EBC: Explainable Boosting Classifier

5.2 *Ranking metrics*

Next, the performance of the classifiers will be assessed on the basis of four ranking metrics: the ROC curve, the AUC-ROC, the PR curve, and the AUC-PR. Figure 4 and 5 present the ROC and PR curves of the decision tree classifier, respectively. The dashed blue line in both figures represents the performance of the no-skill classifier. Remember, the closer the curve gets to the top left corner, the better the algorithm classifies. The no-skill classifier in figure 4 shows the true positive rate to be equal to the false positive rate, meaning that for every correctly classified instance there is an incorrectly specified instance. The ROC curve of the decision tree shows to outperform the no-skill classifier as its completely left of the blue diagonal. One is able to quantify this curve by calculating the area under the curve, the AUC-ROC score. Table 3 displays all AUC-ROC scores for all classifiers. Column 3 shows that the AUC-ROC of the decision tree is higher than the AUC-ROC of the no-skill classifier, with scores of 0.697 and 0.500, respectively. Figurre 5 presents the PR curve of the decision tree. In this case, the top right corner represents a high-skill classifier. The PR curve of the decision tree runs similar to a diagonal displaying no high-skill performance, resulting in an AUC of 0.281. These ranking metrics confirm the decision tree to be a poor classifier when it comes to identifying VAT on incoming invoices.

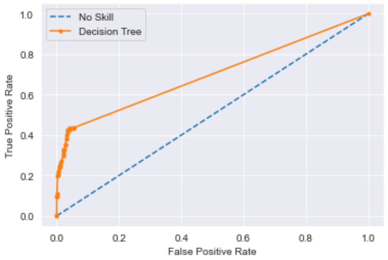

Figure 4: ROC curve of the decision tree

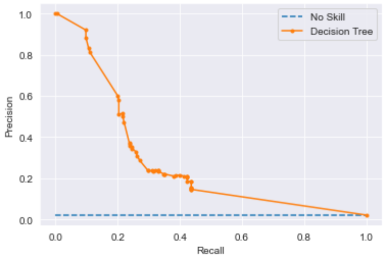

Figure 5: PR curve of the decision tree

In figure 6 and 7, the ROC and PR curves of the random forest classifier are presented. The random forest classifier outperforms the no-skill classifier clearly in addition to outperforming the decision tree. This results in an AUC-ROC of 0.864 - displayed in column 4, table 3 - which is much higher than the AUC of the no-skill classifier. The PR curve shows a steep decline initially before steadily declining further, resulting in an AUC-PR of 0.195.

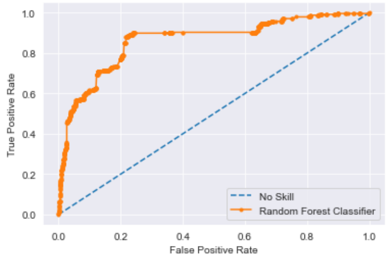

Figure 6: ROC curve of the random forest

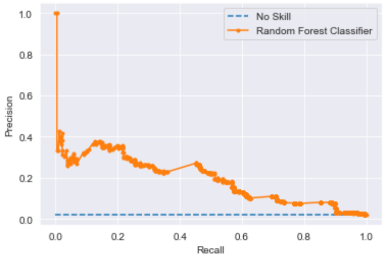

Figure 7: PR curve of the random forest

Figure 8 presents the ROC curve of the gradient boosting classifier which touches the diagonal of the no-skill classifier, indicating the poor performance of the classifier as it means that for every correctly identified instance there is approximately one incorrectly identified instance. This is confirmed by an AUC-ROC of 0.625, shown in column 5 in table 3. The PR curve of the gradient booster, displayed in figure 9, shows its poor performance as well.

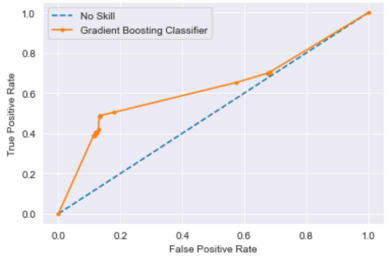

Figure 8: ROC curve of the gradient booster

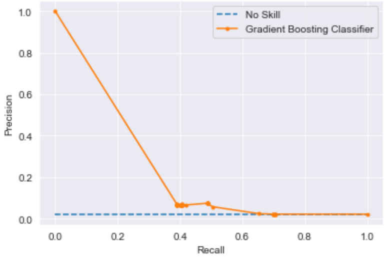

Figure 9: PR curve of the gradient booster

The explainable booster, on the other hand, confirms the steady performance which was already visible when assessing the threshold metrics. Its ROC curve, shown in figure 10, is always left of the no-skill diagonal, implying it always outperforms the dummy classifier. Column 6 in table 3 shows an AUC-ROC of 0.666 which is lower than the random forest classifier but higher than the decision tree and gradient booster. Additionally, its

PR curve, shown in figure 11 never hits the blue horizontal line. Combined with an AUC of the PR curve of 0.192, this confirms the skillful performance of the classifier. Nonetheless, taking the threshold and ranking metrics together, we must conclude the random forest classifier to perform best on the data.

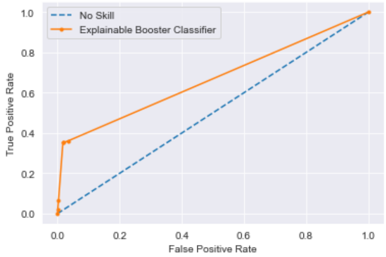

Figure 10: ROC curve of the explainable booster

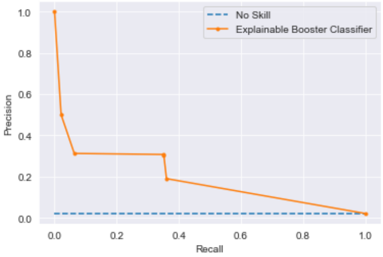

Figure 11: PR curve of the explainable booster

5.3 *Error analysis*

The following section is used to describe the post-hoc error analysis which is performed to shed a different light on the findings of this research. First, the confusion matrices per classifier will be discussed. Second, the scalability of the models will be described. Third, a post-hoc feature importance analysis is performed and will be explained. And lastly, the performance of the classifiers will be elaborated on per subclass.

5.3.1 *Confusion matrices*

Confusion matrices per classifier are displayed in figures 12, 13, 14, and 15. The plots show how many predictions were correct on the majority and minority class as well as the false positives and false negatives. All classifiers show the minority class to be predicted correct most often which was expected as there is a huge class imbalance. The decision tree shows to predict very few false negatives (0.98%) relative to false positives (4.96%). This is similar for all classifiers. However, the gradient booster shows to perform the poorest as its true negative rate, meaning its ability to identify the majority class, is lowest with 86.74%, in addition to a very high false positive rate (11.13%). However, it must also be noted that false positives are less costly than false negatives.

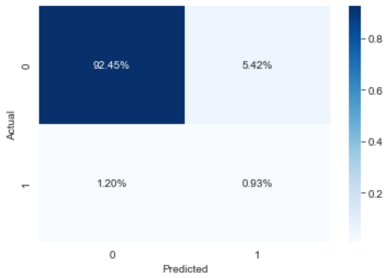

Figure 12: Confusion matrix of the decision tree

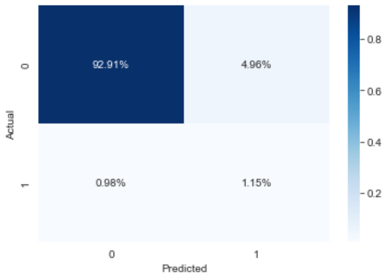

Figure 13: Confusion matrix of the random forest

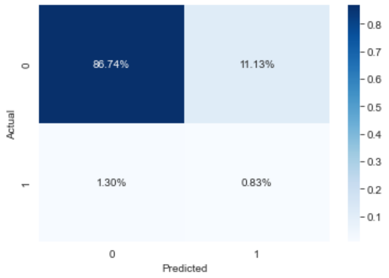

Figure 14: Confusion matrix of the gradient booster

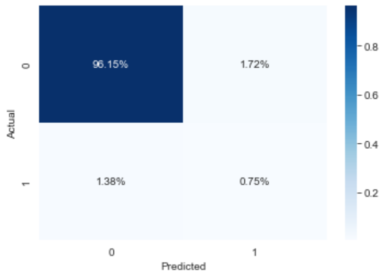

Figure 15: Confusion matrix of the explainable booster

5.3.2 *Scalability*

Figures 16, 17, 18, and 19 depict the scalability of the models, i.e. the time each classifier needs to run when increasing the number of samples. When assessing the y-axes which display the fit times of each classifier, one is able to observe the decision tree – displayed in figure 16 - to report the smallest ones. This implies the classifier to be the least computationally expensive. The explainable booster – displayed in figure 19 -, on the other hand, displays quite the opposite with fit times that are around a hundredfold larger. The random forest and gradient booster – displayed in figures 17 and 18 – are in between these two extremes with fit times around 10 times larger than the decision tree. Nonetheless, all classifiers present a somewhat linear relation between number of samples and fit times, implying the classifiers need to take proportionate time extra when additional samples are fed into the model. Especially the random forest and gradient booster show a similar course. The decision tree and explainable booster, however, give an indication that the increase in fit time attenuates with the number of samples.

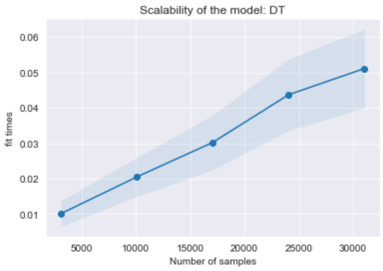

Figure 16: Scalability of the decision tree

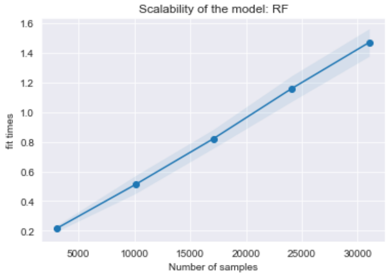

Figure 17: Scalability of the random forest

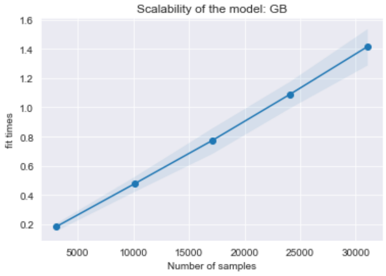

Figure 18: Scalability of the gradient booster

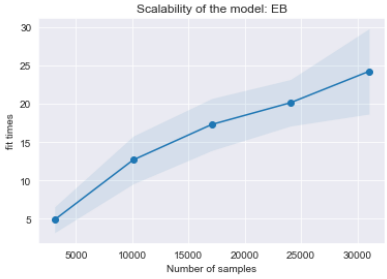

Figure 19: Scalability of the explainable booster

5.3.3 *Effect sizes*

To assess whether the correct features were fed into the model post-hoc, one can turn to a logistic regression to statistically analyse whether the correct choices were made. The results of the logistic regression which present that dependent variables alongside its effect sizes and corresponding p-values are displayed in figure 4. Almost all variables are significant at the 1% level, except for the reversed flag (β = 1.395, p = 0.018) and reporting period (β = 0.422, p = 0.017), which are significant at the 5% level. The debit credit and the late posting feature show to be insignificant, suggesting no apparent relation between these and the fact whether a product or service on a line item contains VAT. Additionally, the model shows an R^2 of 22.4% which means that around 22% of the data's variability is explained by the variability of the features used. Although this is generally described as weak, one must recall that VAT is often not only determined by characteristics of the supplier – which are stated on the invoice – but dependent on the performance of the receiver itself (Belastingdienst, 2022). Therefore, one cannot unilaterally conclude the R^2 in this case to be low.

Table 4: Logistic regression for feature importance

Feature	Coefficient (β)	Std error
Constant	-1.448***	(0.395)
Debit credit	-0.064	(0.368)
Reversed flag	1.395**	(0.591)
Log related	-2.731***	(0.175)
Product group	-0.007***	(0.001)
Product type key	-0.308***	(0.088)
Reporting period	0.422**	(0.177)
Late posting	0.182	(0.152)
First SIC digit	-0.773***	(0.252)
First two SIC digits	0.135***	(0.031)
R^2	0.224	
Observations	11,085	

Significance is displayed at the ***1%, **5% and *10% levels.

5.3.4 *Performance per subclass*

Lastly, the performance of the classifiers on subclasses within the data will be discussed. Recall that the aim of this research is to classify whether invoices that have been deemed VAT-free actually do have VAT stated on them. However, as ERP data is recorded per line item, VAT-free line items could receive a positive label when they are stated on an invoice which contains both VAT-free as well as VAT-carrying goods. From a practical perspective, this is not necessarily bad as we want those - VAT-free - line items to be positively classified as well as these are stated on VAT-carrying invoices. However, this does imply that classifiers learn from negative instances. Therefore, we need to assess these cases separately and thus, the performance of the four classifiers is assessed based on these subclasses. The data is divided into three subclasses: line items which contain VAT and have a positive label accordingly, line items which do not contain VAT yet do have received a positive label, and all other line items which do not contain VAT and have received a negative label accordingly. The results of this analysis is displayed in table 5. The random forest classifier is found to perform best in identifying the first subclass of line items (VAT-carrying invoices with a correct positive label) as displayed in column 4 (0.503%). More surprisingly, columns 4, 5, and 6 show that the other classifiers are better in identifying line items which do not contain VAT but are stated on VAT-carrying invoices than invoices which do contain VAT. This suggests the classifiers to learn from features which are not solely characteristic to VAT-carrying line items.

Table 5: Performance per subclass

	No-skill	DTC	RFC	GBC	EBC
VAT-free & positive label	0.977	0.552	0.414	0.471	0.471
VAT-carrying & positive label	0.014	0.428	0.503	0.297	0.234

DTC: Decision Tree Classifier, RFC: Random Forest Classifier, GBC: Gradient Boosting Classifier, EBC: Explainable Boosting Classifier

6 DISCUSSION

The following section is divided into three parts. First, a brief summary of our findings will be given in addition to placing these into context. Next, the practical implications in the business domain will be elaborated on. Lastly, limitations of the research will be discussed and some recommendations for future research will be given.

6.1 *Discussion of the results*

The goal of this research was to examine whether several machine learning techniques were able to identify whether VAT is prevalent on incoming invoices which were at first ought to be VAT-free. These machine learning techniques were a decision tree, a random forest, a gradient boosting, and an explainable boosting classifier. Using several threshold and ranking metrics we were able to identify the random forest classifier to be perform best in identifying VAT from the inputted data. This contradicts Freitas (2004) and Mori and Uchihira (2019) which explain the accuracy interpretability trade-off as a negative linear relationship between these two entities. The more complex a model gets, i.e. less interpretable, the more accurate a model becomes. When looking solely at F2-scores, one could argue in favour of the trade-off with only the gradient booster as obvious outlier. However, when taking the other threshold and ranking metrics into account, the random forest's performance is too high and the gradient booster's performance too low for the trade-off to hold. There are two possible explanations for this finding: limited hyperparameter tuning and class imbalance. Limited hyperparameter tuning follows from using a grid search to find the optimal configuration of the algorithms. A grid search is an extensive search of all combinations of hyperparameter values one puts in. However, this implies that only the optimal configuration within the set of inputted parameters will be found. Therefore, its quality is completely dependent on the values the operator puts in. Additionally, the class imbalance is only accounted for in the hyperparameters of the decision tree and random forest. Although boosters generally do not need to have a hyperparameter for this as the boosting goes on as long as the performance improves, it is possible that the boosters have found a local minimum instead of a global minimum. Lastly, this research adds to contemporary literature as we have proposed two novelties. The first one is the identification of VAT instead of the identification of line items and the second one is classifying invoices based on ERP data instead of attempting to extract features from the invoice itself.

6.2 *Practical implications*

Of equal importance to the accuracy interpretability trade-off is the time saved employing machine learning methods instead of manual labour, when looking from a business perspective. Remember, the VAT reclamation is currently a manual process of tax advisors going through all invoices looking for VAT before assessing whether this VAT can be reclaimed. It is a very time-consuming and, therefore, very costly process as their earnings are based on whether they find reclaimable VAT (Horsthuis et al., 2020). The most important metric for tax advisors would be recall as it specifies the percentage of all positive cases identified amongst all positive cases. Assuming VAT to be normally distributed amongst the invoices, a recall of 0.9 would imply that 90% of all VAT will be found. Depending on the total document value of all invoices combined, one is able to quantify the monetary loss due to this 10% lack in accuracy on the positive class. However, this is compensated by the monetary gain of not having to go through all invoices manually. We observe the upside potential to be quite large - high likelihood of finding the majority of VAT -, while the downside risk is minimised - missing VAT is not very costly as not a lot of manual hours would be lost. Therefore, tax advisors would benefit employing a random forest approach to these types of inquiries.

6.3 *Limitations & recommendations for future research*

This research contributes to current literature by finding that the accuracy interpretability trade-off does not hold for VAT identification on incoming invoices in addition to serving as a proof of concept for using machine learning methods to identify VAT. However, there are certain limitations to this research which could be addressed in future research. The first two limitations are of methodological nature, while the latter two are related to the data set.

First of all, as mentioned before, hyperparameters are needed to be tuned for optimal performance of the classifiers. Employing a grid search is computationally very expensive, yet needed to assess every possible configuration of the model. However, it is limited by the values one incorporates into the grid search. Therefore, it is possible that some configurations are not tested, including the most optimal one. Secondly, SMOTE and random undersampling are employed to remedy the huge class imbalance. Although SMOTE is the state-of-the-art method for oversampling, it does add 'fake' instances to the model through which the classifiers learn. In addition, random undersampling removes instances from the sample, therefore, losing information. Future research could find optimal values for the degree to which one should over- and undersample.

Next, we are taking a critical look at the data used to perform our research. The data consisted of a sample of invoices which were deemed VAT-free. The data used in this research consisted of the first 4,595 invoices shared by the client, meaning these 4,595 invoices are only a subset of all VAT-free deemed invoices they possess. Therefore, we are not able to assess whether the employed data set is actually representative of the complete data. This means we are unable to generalise our findings to the complete sample, let alone generalise across companies. Future research should attempt to reproduce this research using a sample of invoices from multiple companies in order to be able to make conclusions regarding generalisation. Lastly, the metadata was ordered on a per line item basis, meaning that a single invoice consisting of multiple line items (per service or good supplied) will lead to more observations than there are invoices. Subsequently, some VAT-free line items are labeled as VAT-carrying as these line items are featured on an invoice which includes VAT-carrying line items. While this is in principle not a problem as the aim of this research is to identify all VAT-carrying invoices, this does mean that our classifiers learn from wrongly labeled data. Although we account for this in the error analysis, future research would benefit from making this distinction at the start of the research.

7 CONCLUSION

This research aims to examine whether various machine learning classifiers, such as a decision tree, a random forest, a gradient booster, and explainable booster, are able to identify VAT on incoming invoices which were at first deemed VAT-free. Additionally, we have aimed to discover which of these previously mentioned classifiers performs best on our data. Because a distinction between VAT-carrying and VAT-free invoices has been made prior to our research scope, we have dealt with an extremely imbalanced data set. Using ERP data ordered on a per line item basis, we found the random forest to perform best in terms of multiple threshold and ranking metrics. This contradicted the accuracy interpretability trade-off of Freitas (2004) and Mori and Uchihira (2019) which prescribes the more complex a model gets, the more accurate it becomes. In our case, the gradient and explainable booster are more complex in terms of interpretability and were therefore expected to perform better. Additionally, this research proposed two novelties to the literature of invoice classification. The first one is the ability to identify VAT on incoming invoices and the second one is the use of ERP data. As only invoices of one company were used in this research, there are questions regarding the generalisability of this research. Therefore, future research should aim to reproduce this research with a combined data set of invoices from multiple companies.

REFERENCES

Bardelli, C., Rondinelli, A., Vecchio, R., & Figini, S. (2020). Automatic electronic invoice classification using machine learning models. *Machine Learning and Knowledge Extraction*, *2*(4), 617–629.

Bartoli, A., Davanzo, G., Medvet, E., & Sorio, E. (2010). Improving features extraction for supervised invoice classification. In *Artificial intelligence and applications.*

Belastingdienst, D. (2022). *Welke btw mag ik aftrekken?* Retrieved 2022-05-06, from `https://www.belastingdienst.nl/wps/wcm/connect/nl/btw/content/welke-btw-mag-ik-aftrekken`

Cervantes, J., Garcia-Lamont, F., Rodríguez-Mazahua, L., & Lopez, A. (2020). A comprehensive survey on support vector machine classification: Applications, challenges and trends. *Neurocomputing*, *408*, 189–215.

Chen, J. Y., Procci, K., Boyce, M., Wright, J., Garcia, A., & Barnes, M. (2014). *Situation awareness-based agent transparency* (Tech. Rep.). Army research lab aberdeen proving ground md human research and engineering directorate.

Davenport, T., & Harris, J. (2007). Competing on analytics: The new science of winning boston. *MA: Harvard Business School Publishing*.

Davenport, T. H. (1998). Putting the enterprise into the enterprise system. *Harvard business review*, *76*(4), 121–131.

Freitas, A. A. (2014). Comprehensible classification models: a position paper. *ACM SIGKDD explorations newsletter*, *15*(1), 1–10.

Goossenaerts, J. B., Zegers, A. T., & Smits, J. M. (2009). A multi-level model-driven regime for value-added tax compliance in erp systems. *Computers in industry*, *60*(9), 709–727.

Hamza, H., Belaïd, Y., Belaïd, A., & Chaudhuri, B. B. (2008). Incremental classification of invoice documents. In *2008 19th international conference on pattern recognition* (pp. 1–4).

Hastie, T., Tibshirani, R., Friedman, J. H., & Friedman, J. H. (2009). *The elements of statistical learning: data mining, inference, and prediction* (Vol. 2). Springer.

Horsthuis, M., Zegers, A., & Haenen, R. (2020). *Reverse indirect tax audit using advanced analytics.* KPMG.

Jadhav, S. D., & Channe, H. (2016). Comparative study of k-nn, naive bayes and decision tree classification techniques. *International Journal of Science and Research (IJSR)*, *5*(1), 1842–1845.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning* (Vol. 112). Springer.

Keen, M. (2007). Vat attacks! *International tax and public finance*, *14*(4), 365–381.

Khan, A. (2020). *Comparison of machine learning approaches for classification of invoices* (Unpublished master's thesis).

Lahann, J., Scheid, M., & Fettke, P. (2019). Utilizing machine learning techniques to reveal vat compliance violations in accounting data. In *2019 ieee 21st conference on business informatics (cbi)* (Vol. 1, pp. 1–10).

Larsson, A., & Segerås, T. (2016). *Automated invoice handling with machine learning and ocr.*

Mercado, J. E., Rupp, M. A., Chen, J. Y., Barnes, M. J., Barber, D., & Procci, K. (2016). Intelligent agent transparency in human–agent teaming for multi-uxv management. *Human factors*, *58*(3), 401–415.

Miller, T. (2019). Explanation in artificial intelligence: Insights from the social sciences. *Artificial intelligence*, *267*, 1–38.

Mori, T., & Uchihira, N. (2019). Balancing the trade-off between accuracy and interpretability in software defect prediction. *Empirical Software Engineering*, *24*(2), 779–825.

Nori, H., Jenkins, S., Koch, P., & Caruana, R. (2019). Interpretml: A unified framework for machine learning interpretability. *arXiv preprint arXiv:1909.09223*.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Pijnenburg, M., Kowalczyk, W., Dijk, H.-V., van der Hel-van, D. E., et al. (2017). A roadmap for analytics in taxpayer supervision. *Electronic Journal of e-Government*, *15*(1), 19–32.

Schapire, R. E. (1999). A brief introduction to boosting. In *Ijcai* (Vol. 99, pp. 1401–1406).

Shalev-Shwartz, S., & Ben-David, S. (2014). *Understanding machine learning: From theory to algorithms.* Cambridge university press.

Sorio, E., Bartoli, A., Davanzo, G., & Medvet, E. (2010). Open world classification of printed invoices. In *Proceedings of the 10th acm symposium on document engineering* (pp. 187–190).

Su, J., & Zhang, H. (2006). A fast decision tree learning algorithm. *Faculty of Computer Science, University of New Brunswick*, *6*, 500–505.

Tarawneh, A. S., Hassanat, A. B., Chetverikov, D., Lendak, I., & Verma, C. (2019). Invoice classification using deep features and machine learning techniques. In *2019 ieee jordan international joint conference on electrical engineering and information technology (jeeit)* (pp. 855–859).

van den Biggelaar, S., Janssen, S., & Zegers, A. (2008). Vat and erp: What a cio should know to avoid high fines. *Compact*, *10*(2), 15–21.

van Loo, L., Zegers, A., & Haenen, R. (2015). Data analytics applied in the tax practice. *Compact*, *1*, 3–11.

Zhang, H. (2004). The optimality of naive bayes. *Faculty of Computer Science, University of New Brunswick*, *1*(2), 6.

Śmietanka, B.-O. M., A. (2020). *Study and reports on the vat gap in the eu-28 member states.* European Commission, Directorate-General for Taxation and Customs Union.

APPENDIX A

Table 6: List of all features in the ERP data including their corresponding type

Feature	Type
Instance id	Nominal
Accounting document number	Nominal
Debit credit	Nominal
Reversed flag	Nominal
Document source	Nominal
Fin related	Nominal
Log related	Nominal
Tax related	Nominal
Legal entity key	Nominal
Legal entity id	Nominal
Legal entity country id	Nominal
Posting date	Nominal
Document date	Nominal
Reporting date	Nominal
Calendar period	Nominal
Fiscal year	Nominal
Product key	Nominal
Product id	Nominal
Product group	Nominal
Product type key	Nominal
Product taxable transaction	Nominal
Vendor key	Nominal
Vendor id	Nominal
Vendor name	Nominal
SIC	Nominal
Vendor site key	Nominal
Vendor site id	Nominal
Vendor country id	Nominal
Vendor country EU flag	Nominal
Receiving warehouse key	Nominal
Receiving warehouse id	Nominal
Receiving warehouse country id	Nominal
Supplying country	Nominal
Entry user key	Nominal

Table 6 – continued from previous page

Feature	Type
PTP key	Nominal
Header key	Nominal
Net document value doc	Continuous
Document value doc	Continuous
Tax value	Continuous
Currency doc	Nominal
Net document value le	Continuous
Tax value le	Nominal
Currency le	Nominal
Goods flow	Nominal
Invoice flow	Nominal
Tax fin log	Nominal
Months late	Continuous
Quarters late	Continuous
Reporting period	Nominal
Monthly declaration used flag Declaration used flag	Nominal
Net document value reporting	Continuous
Tax value reporting	Continuous
Tax value calculated reporting	Continuous
Tax value calculated le	Continuous
Tax value difference reporting	Continuous
Tax value difference le	Continuous
Interest lost reporting	Continuous
Currency type reporting	Nominal
Exchange rate	Continuous
Late posting	Nominal
Interest lost	Continuous
Late grouping	Nominal
Period late	Nominal
Total tax value reporting	Continuous
Total document value reporting	Continuous
Labels	Nominal

APPENDIX B

Table 7: Features extracted from the Chi^2 test for feature importance with their description included

Feature	Description
Debit credit	Shows whether invoice should be credited or debited
Reversed flag	Shows whether the invoice was cancelled
Log related	Categorical variable indicating whether the line item is po, pi, or tl
Product group	Categorical variable (256 categories) indicating the category of the good or service delivered
Product type key	Categorical variable showing to which type key a product belongs
Reporting period	Shows in which reporting period (4 quarters) the invoice has been received
Late posting	Binary variable (yes / no) indicating whether the invoice has been posted into the ERP system late
First SIC digit	Categorical variable (9 categories) indicating the industry in which the supplier is active
First two SIC digits	Categorical variable (83 categories) indicating the subindustry in which the supplier is active

APPENDIX C

Table 8: List of tested and adopted values of the hyperparameter tuning

Hyperparameter	Tested values	Adopted value
Decision tree		
criterion	[gini, entropy]	gini
min samples split	[5, 25, 50]	5
min samples leaf	[7, 9, 11, 13, 15]	9
class weight		0:0.01, 1:0.99
max depth		None
Random forest		
n estimators	[50, 150, 250]	50
max features	[auto, sqrt]	auto
min samples split	[50, 100, 150]	100
min samples leaf	[5, 7, 9, 11, 13]	5
class weight		0:0.01, 1:0.99
max depth		None
Gradient booster		
n estimators	[5, 25, 50]	5
learning rate	[5, 10, 50, 100]	5
min samples split	[5, 25, 50]	5
min samples leaf	[3, 5, 7, 9]	3
max depth		None
Explainable booster		
interactions	[18, 28, 38]	18
learning rate	[0.2, 2, 20]	0.2
min samples leaf	[7, 13, 19]	7