Towards Optimal Free Trade Agreement Utilization through Deep Learning Techniques

Johannes Lahann
German Research Center for Artificial Intelligence and Saarland University
johannes.lahann@dfki.de

Martin Scheid
German Research Center for Artificial Intelligence and Saarland University
martin.scheid@dfki.de

Peter Fettke
German Research Center for Artificial Intelligence and Saarland University
peter.fettke@dfki.de

Abstract

In recent years, deep learning based methods achieved new state of the art in various domains such as image recognition, speech recognition and natural language processing. However, in the context of tax and customs, the amount of existing applications of artificial intelligence and more specifically deep learning is limited. In this paper, we investigate the potentials of deep learning techniques to improve the Free Trade Agreement (FTA) utilization of trade transactions. We show that supervised learning models can be trained to decide on the basis of transaction characteristics such as import country, export country, product type, etc. whether FTAs can be utilized. We apply a specific architecture with multiple embeddings to efficiently capture the dynamics of tabular data. The experiments were evaluated on real-world data generated by Enterprise Resource Planning (ERP) systems of an international chemical and consumer goods company.

1. Introduction

Free trade agreements (FTA) have been one of the most prominent issues in the world economy. The number of FTA has constantly been increasing since the early 1990s. As of January 2012, the World Trade Organization (WTO) notified of about 500 regional trade agreements, including agreements on goods and services [1]. According to the Dictionary of International Trade FTA is defined as an arrangement that establishes unimpeded exchange and flow of goods and services between trading partners, regardless of national borders of member countries [2]. FTA are considered to be one of the most effective policy tools for enhancing trade volume between member countries. Particularly, in recent years, when multilateral trade liberalization through World Trade Organization (WTO) has slowed, bilateral trade liberalization through FTA has played a central role in increasing world trade. FTA utilization is expected to affect the performance of firms in several ways. The use of FTA reduces the market price of the firms' products in importing countries. Such reductions increase demand for their products. If the price elasticity of demand is greater than unity this leads to a rise in product imports.

If it comes to the optimal utilization of FTA from a firm's perspective, Schaub et al. describe three major challenges that need to be considered [3]. First, the design of the production and distribution network is crucial for the utilization of FTA. Due to rules of origin (ROO), companies must adapt their sourcing pattern of parts of materials in order to be able to take advantage of FTA utilization for the finished products. Second, companies must ensure that they comply with the compliance process within the framework of FTA utilization. This ranges from the collection of information from the FTA to ensuring that goods that satisfy ROO will do so in the longer term. The third challenge is concerned with the allocation of functional and geographical responsibilities that are specifically dedicated to the utilization of FTA. Most of the reasons why companies do not use FTA are related to the FTA compliance process. These range from the collection of information regarding FTA to ensuring the goods satisfy the ROO on a consistent basis. In general, such a process must meet two requirements. First, it must be ensured that a company complies with the rules and regulations that apply to a particular FTA, and second, a cost or time saving must be achieved by using the FTA.

This work focuses on the third argument. Internationally active companies possess numbers of transactions through multiple Enterprise Resource Planning (ERP) systems, making it extremely difficult to overlook all relevant information that need to be taken into account for effective FTA utilization. Recently, machine learning and especially deep learning technologies have achieved great success in the analysis and interpretation of large amounts of data in various domains such as computer vision [4] and speech recognition [5, 6]. Inspired by those developments,

URI: https://hdl.handle.net/10125/63918
978-0-9981331-3-3

in this work, we explore the potentials of machine learning in order to improve the utilization of FTA. The contribution of this paper are threefold:

1. This study presents the utilization of FTA as an interesting and relevant use case both for research and practice. The use case offers great potential for the effective application of machine learning, data mining, and optimization techniques.

2. This study shows that machine learning models can be trained to decide whether FTA can be exploited for export transfer transactions. To our knowledge, no prior work applied machine learning models in this domain.

3. In addition, the approach has been extended to a multi-class problem in which different reasons have been identified explaining why FTA cannot be utilized for a certain export transfer transaction.

The paper follows the "exaptation" (extend known solutions to new problems) type of Design Science research knowledge contribution by adapting existing techniques (deep learning classifier), which had success in various domains, to build innovative solutions to improve FTA utilization [7].

The remainder of the work is structured as follows: Section 2 presents related work on ERP data analysis. In section 3 a more detailed description of the FTA utilization use case is given. Section 4 introduces a deep learning based approach for effective FTA utilization. In particular, a binary classification task to detect optimizable transaction as well as a multi-class classification task, that further investigates different reasons for the decision are formulated. The experimental setup and results of the proposed approach are presented in section 5. Section 6 concludes the paper with a discussion and summary.

2. Related Work

In this section, we focus on related work that applied data mining, optimization or machine learning methods on data collections extracted from ERP systems. In 2006, Bay et al. were the first to perform data analysis methods on ERP data [8]. They used Naive Bayes method to identify suspicious general ledger accounts. Their approach was complemented by McGlohon et al, who applied link analysis to identify (sub)-groups of high-risk general ledger accounts [9]. Kahn et al. utilized activity pattern recognition methods to create transaction profiles of SAP ERP users in order to detect suspicious user behavior and violations of duties [10]. Islam et. al used a matching algorithm to evaluate audit logs from SAP R/3 systems to identify fraud scenarios [11]. Another approach was taken by Debreceny and Gray, who applied Benford's law to find unusual combinations of numbers and unusual temporal patterns [12]. Argyrou et al. evaluated self-organizing maps to identify suspicious journal entries of a shipping company [13]. In particular, they calculated the Euclidean distance of a journal entry and the code-vector of a self-organizing maps best matching unit. In subsequent work, they applied extreme value theory to estimate optimal sampling thresholds of journal entry attributes [14]. Jans et al. applied latent class clustering on SAP ERP purchase order transactions [15]. Transactions significantly deviating from the cluster centroids are flagged as anomalous and are proposed for a detailed review by auditors. The approach was further enhanced by the integration of process mining techniques to detect deviating process flows in procure to pay processes of firms [15]. More recently, Schreyer et al. used deep autoencoder networks to detect anomalies in journal entries [16]. They evaluated an adapted reconstruction error of the autoencoder as an anomaly assessment. In a similar fashion, Paula et al. applied autoencoder based anomaly detection to investigate fraud in Brazilian exports and anti-money laundering [17].

According to our review, the majority of existing references draw on traditional data analytics and rather than machine learning and, in particular, deep learning techniques. As a result and in agreement with [18, 16], we see a demand for learning-based approaches capable to detect unknown scenarios. Also, to the best of our knowledge, this work presents the first machine learning approach in the context of FTA utilization of export transfer transactions.

3. Use Case

In the conducted study we worked with a large international company of the chemical and consumer goods industry. Through its global presence there are exports and imports of several thousand products a day. These shipments are made by different logistics companies all over the world and tracked with different IT systems. It has several locations all over the world which operate independently of one another. Efficient utilization of available FTA can lead to considerable potential savings and thus, plays an important role in the company's strategy. Therefore, the company invests large manual efforts to achieve the highest possible coverage of FTA-utilization. New transactions must be checked in a manual process to see whether there are

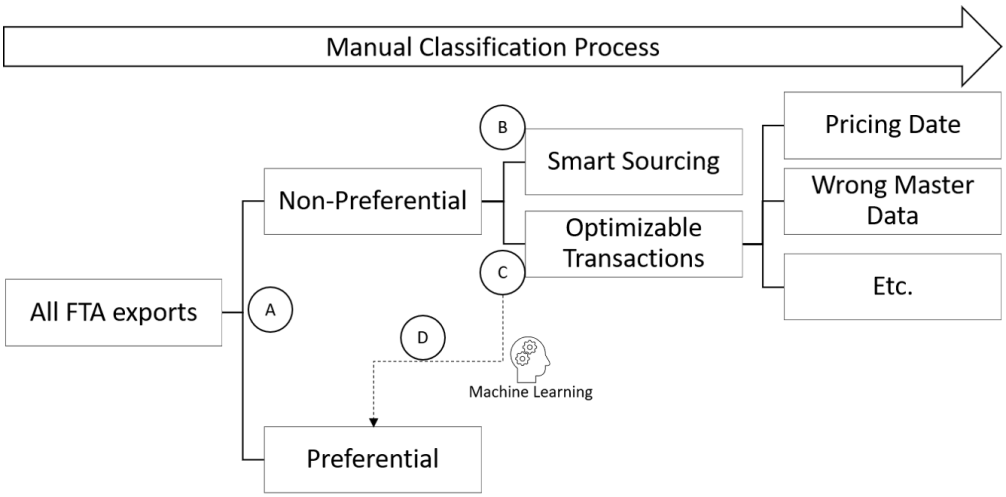

Figure 1. Motivating Example: FTA-utilization of trade report transactions.

saving potentials through the utilization of FTA. At the same time, legal changes in the determination of FTA or changes in the company's sourcing strategy can result in existing FTA no longer being utilizable. This leads to a large amount of non-preferential transactions, where the FTAs where not used. That has to be checked by domain experts in a manual process on a regular basis to evaluate the faulty usage and non-usage of FTAs. Figure 1 illustrates the decision-making process that leads to the use or non use of FTAs.

The FTA exports can be divided into preferential transactions (A), i.e. for a given transaction FTA are already utilized, and non-preferential transactions, i.e. for a given transaction no FTA are utilized.

That decision alone is often not trivial and requires careful consideration of the underlying product data. This means that the composition of the product also plays a role, as does the length of the individual components which is also stored in the products' master data. For example, this also includes whether the components themselves are preferential or not. With non-preferential transactions two groups have to be distinguished: The first group consists of transactions that offer no simple options to utilize existing FTA. These transactions would require a substantial adjustment of the sourcing strategy (B). This procedure is called smart sourcing and is not the focus of this paper. The other group includes transactions that can fulfill the conditions of existing FTA by making minor adjustments like adapting the master data or the pricing data (C). This step can lead to products being preferential, for example, as they fall below limit values. We argue that the selection of transactions on which FTA can be utilized with minor effort is improvable through the application of machine learning algorithms to reduce the necessary manual effort significantly (D). To simplify the explanation of the proposed approach, we would like to clarify the following terminology.

Preferential transaction: A transaction is called preferential when available FTA are utilized efficiently.

Non-preferential transaction: A transaction is called non-preferential when no FTA is utilized, yet. However, this does not mean that no available FTA can be utilized.

Optimizable transaction: A non-preferential transaction is called optimizable when available FTA can be utilized.

Non-optimizable transaction: A non-preferential transaction is called non-optimizable when available FTA cannot be utilized at acceptable effort.

4. Proposed Approach

We propose a deep learning based approach to identify optimzable transactions as defined in section 3. The approach is based on the idea that transactions for which FTA have already been used exist and have similar characteristics to transactions where no FTA have been utilized yet. We consider a binary classification problem to distinguish between optimizable and non-optimizable transactions. Next to that, we explore the reasons why a transaction is non-optimizable by defining a mutli-class classification problem. To train our machine learning models we make

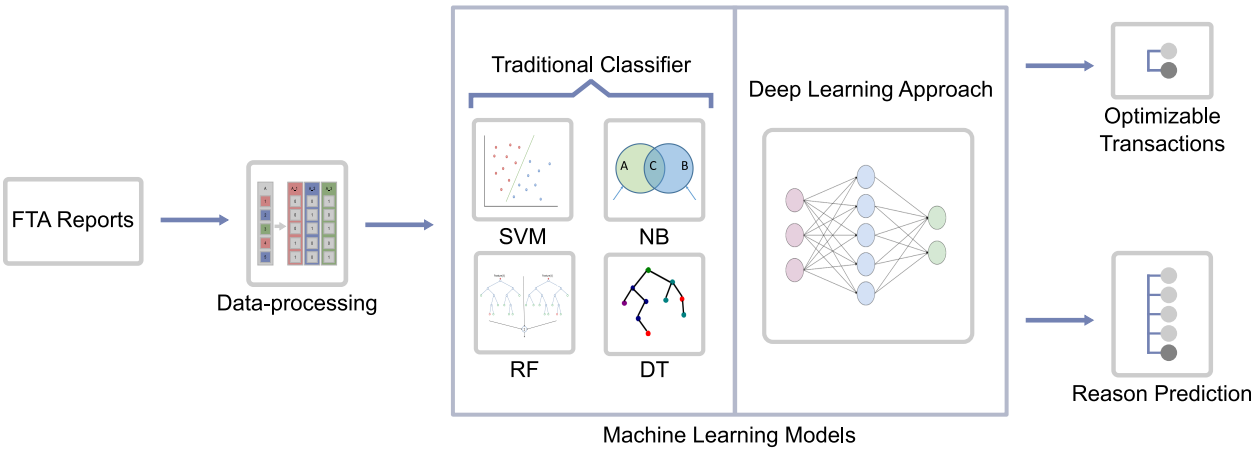

Figure 2. Different Stages of the Proposed Approach

use of a data export of a large chemical and consumer goods company that was manually inspected and labeled in terms of efficient FTA utilization. Figure 2 illustrates the stages of the proposed approach.

4.1. Data-Processing

Prior machine learning studies have shown that data preparation, comprising various stages such as data cleaning, encoding, dimensionality reduction or feature extraction, can have significant effects to the predictive ability of machine learning classifiers [19, 20]. We choose encoding schemes depending on the type and characteristics of the different data fields. In particular, we had to consider data fields of categorical and continuous values as well as a special date column.

Table 1. Additional columns that together represent the date column.

Attribute	Type
TimeYear	int64
TimeMonth	int64
TimeWeek	int64
TimeDay	int64
TimeDayofweek	int64
TimeDayofyear	int64
TimeIs_month_end	bool
TimeIs_month_start	bool
TimeIs_quarter_end	bool
TimeIs_quarter_start	bool
TimeIs_year_end	bool
TimeIs_year_start	bool

Date column A date attribute contains various hidden pieces of information, e.g. the day of the week, the day of the month, wether the day is on the weekend or a holiday, etc. This information can have a high impact on how to evaluate a trade transaction. A deep neural network might be able to learn this hidden information by itself, but would require a large amount of additional data. We chose not to try to learn these kinds of information but used the pandas library to add additional data fields. In particular, we replaced the date column by a number of additional columns that are listed in table 1. The additional columns were then treated as categorical values.

Categorical values Our data set mainly contains ID and Key attributes, i.e. they are represented through categorical data. In [21], various existing categorical variable encoding techniques for neural network classifiers were compared. One-hot encoding is a widely used coding scheme for categorical data. It compares each level of the categorical variable to a fixed reference level. It transforms a single variable with n observations and d distinct classes to d binary variables with n observations, each, as shown in Figure 3. Since with one-hot encoding a new attribute is added for each distinct value of an attribute, this leads to an attribute explosion, i.e. if an attribute can consist of 3 different values, e.g. the numbers 3.249, 3 and 2.99, a new attribute is generated for each of these values. This makes successful training of machine learning models more difficult because of the large amount of new generated data-attributes. In our approach, we solve this problem by adding one embedding layer per attribute. Section 4.2 describes the implementation of the embedding layers.

Continuous values To normalize the continuous data attributes, we use min-max scaling. In this encoding, the

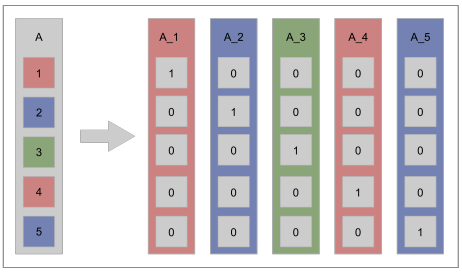

Figure 3. One-hot encoding for categorical data

data is scaled to a fixed interval between zero and one. In contrast to normal standardization, we will end up with smaller standard deviations, which suppresses the effect of outliers.

4.2. Neural Network Architecture

The model consists of one embedding layer for each of the categorical variables, followed by a dropout layer for all categorical variables. The continuous variables are processed through a single batch normalization layer. Afterwards, the encoded continuous and categorical variables are concatenated together and processed through two blocks of affine functions:

1. Rectified linear units (ReLU) and
2. batch normalization and dropout.

Last, they are processed through a final linear layer before they are fed through a sigmoid or softmax layer depending on the type of classifcation problem. The architecture is based on the tabular model of the fastai library[1] and represents best practices from current research in the field of deep learning.

Embedding Layers As neural networks operate on continuous data, categorical inputs must be appropriately encoded. One common option is one-hot encoding as described in section 4.1. However, this has two major drawbacks. First, the dimension of the training data explodes, since for each categorical column there are new columns equal to the cardinality of the attribute added. Second, the model cannot easily capture dependencies between the instances of one attribute, since the attribute is represented as a sparse vector. Both these problems can be solved through the introduction of embedding layers. The embedding layer maps the one-hot representation of categorical values into an n-dimensional embedding space using a so-called embedding matrix. An embedding matrix is essentially a look-up matrix of dimensions $c \times m$

[1] https://docs.fast.ai/

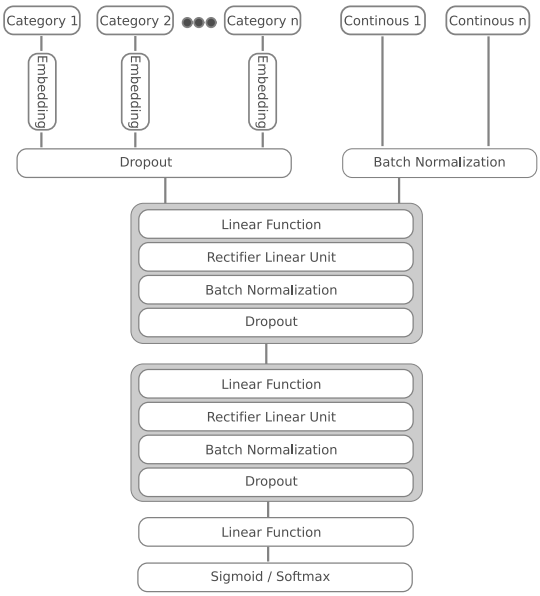

Figure 4. Neural Network Architecture

where c is the cardinality of the attribute and m is the chosen dimensionality of the embedding. By choosing $m < c$ we can reduce the dimensionality of the data. At the same time, the parameters of the embedding matrix can be trained by the neural network. Thus, the neural network can choose a representation by itself that allows it to efficiently model the dependencies between the values of the attribute.

Hyperparameter Selection For hyperparameter selection, we built up on the defaults of the fastai library. In particular, we adapted the number of neurons, the learning rate, the number of epochs, the amount of dropout, and the amount of weight decay. We trained the neural network for 15 epochs, slowly decreasing the learning rate form 0.02 to 0.00005. We used 1000 neurons for the first linear layer and 500 neurons for the second of the two blocks. We used 5% embedding dropout and 20% weight decay. Additionally, we used standard adam optimization and cross-entropy loss.

4.3. Comparison with Traditional Machine Learning Classifier

To measure the feasibility of the proposed deep learning based approach, we compare its performance with support vector machines (SVM), random forests (RF), naive bayes (NB), and single decision trees (DT), which are among the most powerful and most widely-used algorithms for classification problems [22]. The support vector machine and the naive bayes

represent the baseline for the experiments conducted. The decision tree and the random forest algorithms have the advantage that learned allocation rules are easier to comprehend through visualizations. Furthermore, they have proven to be often superior to other machine learning models when it comes to tabular data. Another advantage of tree-based models is that they do not rely so much on data-processing and hyperparameter tuning. For the hyperparameter selection of the traditional machine learning models, we rely on the defaults of the scikit-learn library.

5. Evaluation and Discussion

To gauge the effectiveness of the proposed deep learning approach, we conducted a range of experiments comparing its performance with several established machine learning models. The performance of the models is measured in terms of multiple commonly applied classification metrics.

5.1. Setup

All experiments were performed in two phases on two different computers, since the data-processing stage had different computational requirements than the model training stage. For the data-processing phase we used a machine with 6 TB disk space, 64 GB memory and 20 Intel XEON CPUs E502-2630 v4 with 2.20 GZ. As data manipulation software we used the python packages pandas and numpy. The deep learning models were implemented with the fastai package on top of pytorch. As development environment we used jupyter notebooks. For the traditional machine learning classifier as well as the evaluation metrics, the scikit-learn package was used.

5.2. Dataset

For the experiments, proprietary data from an international company in the consumer goods industry were used. In compliance with strict data privacy regulations, all journal entries have been anonymized using an irreversible one-way hash function during the data extraction process. In a multi-level selection process, the information from multiple tables from various SAP GTS (SAP Global Trade System) systems was collected in an Azure Cloud, converted into a uniform format, and linked together. Both IT experts and custom experts were involved to map all technical requirements and to consider all relevant criteria for the FTA utilization when making the selection of attributes. The dataset comprises 15 different attributes containing general transaction properties, e.g. the customer, the material, the departure and destination country. Table 2 lists a more detailed description of the dataset. In total, 75,828 transactions have been considered. For all these transactions, it has been determined whether a transaction is optimzable or non-optimizable and what the underlying reason for the decision is. The determination has taken place either through a manual or a semi-automated, rule-based approach. The data contains 26,661 optimizable and 53,541 non-optimizable transactions.

Table 2. Description of data attributes

Attribute	Description
PreferenceFlag (card. 2)	Indicates whether an FTA is already being used or not.
BillInterComp (card. 2)	Indicates whether it is a inter company transaction or to an external customer.
DepCountry (card. 19)	The departure country of the transaction.
CommodityCode (card. 519)	Identifies products or groups of products in international commerce.
BusinessUnit (card. 4)	Determines the busiiness unit of the transaction.
MaterialGroup (card. 3096)	The components of the product are assigned to different material groups according to their properties.
DestCountry (card. 41)	The destination country of the transaction.
SoldToCustomer (card. 274)	The customer the good is sold to.
ShipToCustomer (card. 368)	The supplied customer the good is send to.
SalesDocument (card. 4851)	Determines the invoice of the transaction.
PlantKey (card. 94)	Specifies the id of the plant involved.
CondValue (mean 1343, std. 9686)	Determines the price of the transaction.
Time	Day in 2017
Optimizable (card. 2)	Determines if the transaction is optimizable.
Reason (card. 16)	Describes the reason, why the transaction can or can not be optimized.

5.3. Evaluation Metrics

To compare the performance of the classification algorithms, we computed the average accuracy, average precision, average recall, F-Measure and area under the ROC curve as evaluation metrices. We formulated a binary classification problem to detect optimizable transactions and a multi-class classification to detect the underlying reason why no FTA can be utilized. Table 3 shows the formulas of the used metrics. In the table

Table 3. Supervised Learning Metrics.

Metrics	Formula
Accuracy	$\frac{1}{n}\sum_{i=1}^{l} s_i \frac{tp_i+tn_i}{tp_i+tn_i+fn_i+fp_i}$
Precision	$\frac{1}{n}\sum_{i=1}^{l} s_i \frac{tp_i}{tp_i+fp_i}$
Recall	$\frac{1}{n}\sum_{i=1}^{l} s_i \frac{tp_i}{tp_i+fn_i}$
F-Measure	$\frac{1}{n}\sum_{i=1}^{l} s_i \frac{precision \times recall}{precision+recall}$
AUC	$\frac{1}{n}\sum_{i=1}^{l} s_i \int_0^1 tpr_i d(fpr_i)$

fp, tp, fn, tn depict the number of false positives, the number of true positives, the number of false negatives and the number of true negatives, respectively. fpr and tpr are the false positive rate and the true positive rate. The parameter i specifies in a multi-class classification with respect to which class fp, tp, fn, tn, fpr and tpr are calculated. For example, tp_i are the true positives for class i, i.e. the number of events of class i that have been correctly classified by the machine learning model as being of class i. In the multi-class classification scenario all measures are computed for each individual class. Afterwards, the per-class results are weighted with the true class size and summed up. Due to the size of the data set, we refrained from a cross validation evaluation and rather used a simple splitting into training, validation and test set, using 80% of the data for the training set and 10% each for validation and test set.

5.4. Results of the Binary Classification

General Results Table 4 presents the results of the binary classification task of detection optimizable transactions. The results show that an efficient mapping from the feature space to the output space can be learned. The tree based methods as well as the deep learning based methods were able to reach high scores up to 0.9 for most of the measured metrics whereas the SVM and the NB performed significantly worse. The deep learning based approach reached the best performance in terms of precision, accuracy, and F-Score. The deep learning approach and the decision tree performed equally well in terms of recall. Non of the machine learning models had a solid AUC-score. The highest AUC-score was reached by the naive bayes algorithm. We argue that the AUC-score is of less importance in this situation as a classifier does not have to perform well for all possible thresholds, but if needed a specific threshold can be selected respectively.

Table 4. Comparison of classification algorithms to detect optimizable transactions.

	Precision	Recall	Acc	F-Score	AUC
SVM	0.74	0.22	0.71	0.34	0.38
NB	0.38	0.82	0.51	0.52	**0.60**
DT	0.91	**0.91**	0.94	0.91	0.50
RF	0.92	0.90	0.94	0.91	0.44
ANN	**0.94**	**0.91**	**0.95**	**0.92**	0.39

Confusion Matrix Figure 5 shows the confusion matrix of the neural network for the binary classification problem. Although, both classes are not completely balanced, it can be seen that the neural network did not perform significantly worse on one class. However, the confusion matrix emphasizes that the neural network was trained with the cross-entropy loss, which takes all classes at equal weight into account. If we are more interested in finding optimizable transactions, it could be useful to use a custom loss function that stronger penalizes false negatives.

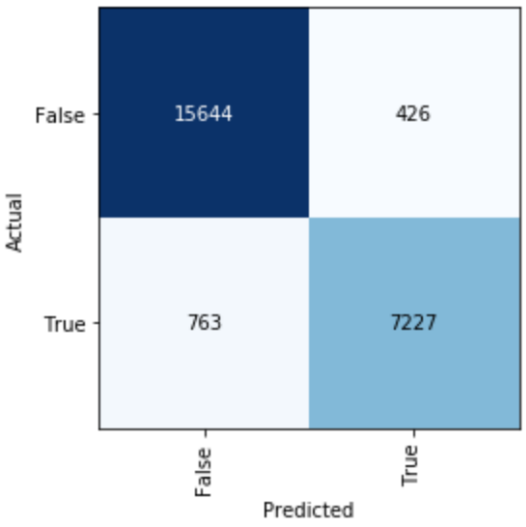

Figure 5. Confusion matrix for Detecting Optimizable Transactions

Feature Importance In order to measure the importance of a single attribute, the accuracy on the entire data set was measured after completion of the training. Two measurements were carried out. For the

first measurement, the original values of the attributes were used. In the second measurement, the values of the attribute were randomized. The change in the achieved accuracy allows conclusions to be drawn about the importance of the attribute for the prediction. A large deviation indicates a high feature importance and a small deviation indicates a low feature importance of a specific attribute. Figure 6 presents the feature importance diagram for the 10 most important attributes. The attributes with the highest feature-importance were the commodity code and the customer. From a practical perspective, this makes absolutely sense. A commodity code is a six digit number that identifies products or groups of products in international commerce. Customs authorities use these numbers to determine the duty and taxes for specific goods. Since FTA's usually cover certain materials or products, we should expect the commodity code to have a high impact on the FTA utilization. The same applies to the customer. A customer is located in a certain country and only buys a specific type of products. Both components have a major impact on the possibility of FTA utilization. In contrast, the relatively high influence of the day of the year looks conspicuous and was reported back to the company.

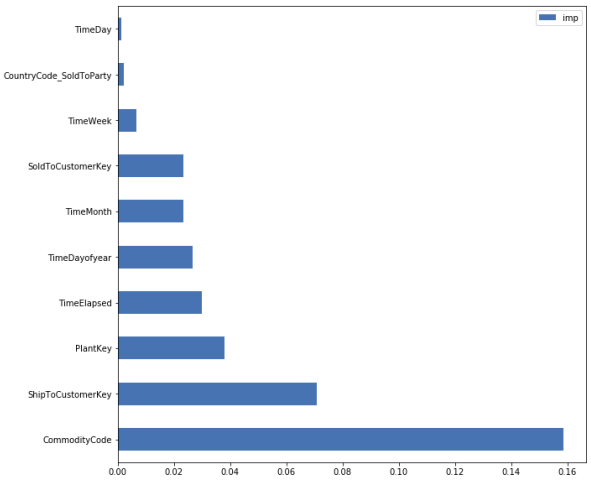

Figure 6. Feature Importance Visualization for Detecting Optimizable Transactions

5.5. Results of the Multi-Class Classification

General results Table 5 presents the results of the multi-class classification task of predicting the reasons why a transaction has been considered optimizable or non-optimizable. Similarly to the binary classification problem, the tree based methods and the deep learning approach perform reasonably, whereas the naive bayes and the support vector machine achieve only moderate results. The deep learning based approach achieved the highest score in all measured metrics. For the neural network a value of 0.90 was observed for all metrics.

Table 5. Comparison of classification algorithms that predict the reason why a FTA is or is not applicable.

	Precision	Recall	Acc	F-Score
SVM	0.38	0.49	0.49	0.40
NB	0.38	0.20	0.20	0.19
DT	0.88	0.89	0.89	0.88
RF	0.88	0.88	0.88	0.88
ANN	**0.90**	**0.90**	**0.90**	**0.90**

Confusion Matrix Figure 7 presents the confusion matrix of the multi-class classification task. In total, 15 different reasons were defined that influenced the decision of a transaction being optimzable. Five of them led to non-optimizable transactions and 10 reasons led to optimizable transactions. The confusion matrix shows that the reasons do not occur equally frequently. Instead, there is an imbalance in the data. Nevertheless, the neural network was able to forecast all reasons similarly well.

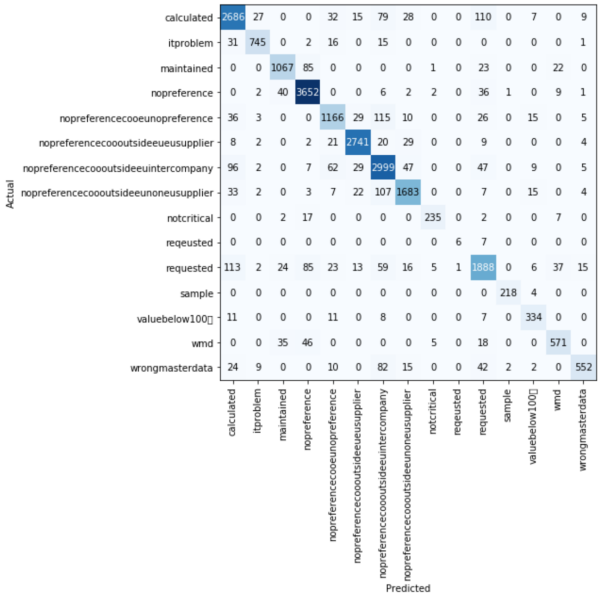

Figure 7. Confusion matrix for Predicting the underlying Reason, that led to optimizable or non-optimizable transactions

Feature Importance The calculation of the feature importance for multi-class classification followed the same procedure as in the binary classification. Figure 8

lists the diagram. Overall, there are a lot of coincidences with the feature importance of the binary classification. For example in both diagrams, the same ten attributes were selected and the commodity code had the highest impact. This is not surprising as the multi-class classification task can be interpreted as a refinement of the binary classification. However, there are also some deviations. For example, the month in which the transaction was executed had the second most influence on the assessment of the transactions. The impact of the month was not much smaller than that of the commodity code. This leads to the assumption that transactions executed in particular months were only assigned to certain reasons. This relationship appears unusual at first glance and requires further investigation by domain experts. Another conspicuous detail is the fact that the customer has clearly become less influential compared to the binary classification.

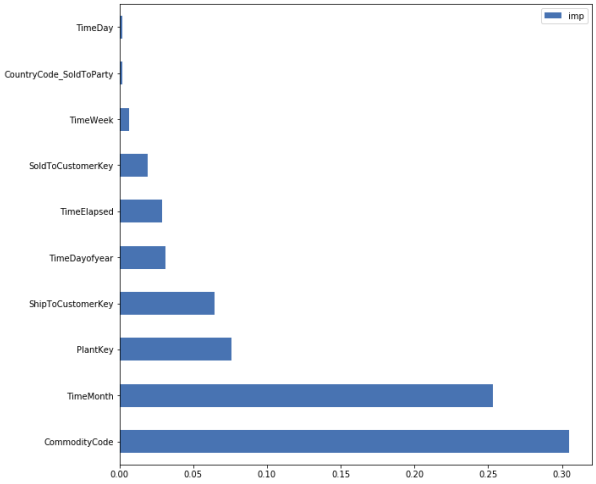

Figure 8. Feature Importance Visualization for Predicting the Reason

6. Conclusion and Outlook

In this work, a machine learning application for effective FTA utilization for export transfer transactions has been presented. We argue that the use case of efficient FTA utilization is an interesting application domain for machine learning and data mining methods. Hence, we developed the application of machine learning in the context of FTA utilization. The evaluation of an ERP data export of a consumer goods industry has shown that supervised classification algorithms are suitable to reveal optimizable transactions based on typical characteristics of a trade transaction. The results of the evaluation have shown that tree based methods are a viable base line for tabular data, which can achieve excellent performance with little hyper parameter optimization. We also confirmed that a deep learning based approach can surpass traditional machine learning approaches.

It has to be pointed out that the possibilities of parameter optimization, and generally, model improvement are certainly not exhausted, yet. It is still possible to improve the performance of the model with hyperparameter optimization or deeper neural network architectures. In addition, the "Black Box" character of deep learning approaches is a relevant topic in this context and needs to be further investigated. Since the decisions have to be legally represented, a detailed justification is necessary. The use case also provides an interesting scenario for the application of online learning methods. On the one hand, there are only labels for a small proportion of transactions. On the other hand, the legal regulations and the export strategy of the company change over time, so that machine learning models have to evolve to still be able to make useful predictions.

References

[1] A. Ulloa and R. Wagner, "Why don't all exporters benefit from free trade agreements? Estimating utilization costs," tech. rep., IDB Working Paper Series, 2012.

[2] W. Goode, World Trade Organization, and University of Adelaide Centre for International Economic Studies, *Dictionary of trade policy terms*, vol. 4. Cambridge University Press Cambridge, UK, 2007.

[3] M. Schaub, *Utilization of Free Trade Agreements:(FTA's) by Companies Trading in Goods*. PhD thesis, Verlag nicht ermittelbar, 2012.

[4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105, 2012.

[5] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed Representations of Words and Phrases and their Compositionality," in *Advances in Neural Information Processing Systems 26* (C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, eds.), pp. 3111–3119, Curran Associates, Inc., 2013.

[6] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," *CoRR*, vol. abs/1301.3, 2013.

[7] S. Gregor and A. R. Hevner, "Positioning and Presenting Design Science Research for Maximum Impact," *MIS Quarterly*, vol. 37, no. 2, pp. 337–355, 2013.

[8] S. Bay, K. Kumaraswamy, M. G. Anderle, R. Kumar, and D. M. Steier, "Large scale detection of irregularities in accounting data," in *Proceedings - IEEE International Conference on Data Mining, ICDM*, pp. 75–86, IEEE, dec 2006.

[9] M. McGlohon, S. Bay, M. G. Anderle, D. M. Steier, and C. Faloutsos, "SNARE: A Link Analytic System for Graph Labeling and Risk Detection," in *Proceedings*

of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '09, (New York, NY, USA), pp. 1265–1274, ACM, 2009.

[10] R. Q. Khan, M. W. Corney, A. J. Clark, and G. M. Mohay, "Transaction mining for fraud detection in ERP Systems," *Industrial Engineering and Management Systems*, vol. 9, no. 2, pp. 141–156, 2010.

[11] A. K. Islam, M. Corney, G. Mohay, A. Clark, S. Bracher, T. Raub, and U. Flegel, "Fraud detection in ERP systems using Scenario matching," *IFIP Advances in Information and Communication Technology*, vol. 330, pp. 112–123, 2010.

[12] F. Benford, "The Law of Anomalous Numbers," in *Proceedings of the American Philosophical Society*, vol. 78, pp. 551–572, 1938.

[13] A. Argyrou, "Auditing Journal Entries Using Self-Organizing Maps," *AMCIS 2012 Proceedings 16*, jul 2012.

[14] A. Argyrou, "Auditing Journal Entries Using Extreme Value Theory," *In Proceedings of the 21st European Conference on Information Systems*, jul 2013.

[15] M. Jans, N. Lybaert, and K. Vanhoof, "Internal fraud risk reduction: Results of a data mining case study," *International Journal of Accounting Information Systems*, vol. 11, pp. 17–41, mar 2010.

[16] M. Schreyer, T. Sattarov, D. Borth, A. Dengel, and B. Reimer, "Detection of Anomalies in Large Scale Accounting Data using Deep Autoencoder Networks," sep 2017.

[17] E. L. Paula, M. Ladeira, R. N. Carvalho, and T. Marzagão, "Deep learning anomaly detection as suppor fraud investigation in Brazilian exports and anti-money laundering," in *Proceedings - 2016 15th IEEE International Conference on Machine Learning and Applications, ICMLA 2016*, pp. 954–960, IEEE, dec 2017.

[18] S. Wang, "A comprehensive survey of data mining-based accounting-fraud detection research," in *2010 International Conference on Intelligent Computation Technology and Automation, ICICTA 2010*, vol. 1, pp. 50–53, IEEE, may 2010.

[19] S. B. Kotsiantis and E. Al, "Data Preprocessing for Supervised Learning," *International Journal of Computer Science*, vol. 1, no. 2, pp. 111–117, 2006.

[20] I. H. Witten, E. Frank, and Mark A. Hall, *Data Mining: Practical Machine learning*. Morgan Kaufmann, 2011.

[21] K. Potdar, T. S., and C. D., "A Comparative Study of Categorical Variable Encoding Techniques for Neural Network Classifiers," *International Journal of Computer Applications*, vol. 175, no. 4, pp. 7–9, 2017.

[22] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, S. Y. Philip, and Others, "Top 10 algorithms in data mining," *Knowledge and information systems*, vol. 14, no. 1, pp. 1–37, 2008.