# Evaluating Machine Learning Algorithms in Predicting Financial Restatements

Gerhard Klassen
Department of Computer Science
HHU Düsseldorf
Düsseldorf, Germany
klassen@hhu.de

Martha Tatusch
Department of Computer Science
HHU Düsseldorf
Düsseldorf, Germany
tatusch@hhu.de

Weisong Huo
Department of Computer Science
HHU Düsseldorf
Düsseldorf, Germany
weisong.huo@hhu.de

Stefan Conrad
Department of Computer Science
HHU Düsseldorf
Düsseldorf, Germany
stefan.conrad@hhu.de

## ABSTRACT

The identification of financial statements which were willfully or accidentally misstated is important for all involved parties: Investors can expect improved returns, analysts preserve their reputation and auditors avoid costly litigation. In this paper, we chose six state-of-the-art machine learning methods which we analyze in their ability to detect misstatements. In addition to that we investigated the influence of a FeatureBoost algorithm, namely XG-Boost to all of the six machine learning methods. The underlying data is retrieved from Eikon [6], a financial database provided by Refinitiv (former provided by Thomson Reuters). In order to take out our experiments we chose about 9000 US-companies and 757 features per year over ten years. We offer six definitions of ground truth of which three can be calculated with the data extracted from the Eikon database. The other three definitions are created with the help of an external data source provided by Audit Analytics Europe [8]. Our well structured results give an overview on the performance of current machine learning methods in order to identify misstatements.

## CCS Concepts

**•Computing methodologies→Machine learning→Machine learning approaches→Classification and regression trees**

**•Computing methodologies→Machine learning→Machine learning algorithms→Ensemble methods→Boosting**

**•Computing methodologies→Machine learning→Machine learning algorithms→Feature selection**

**•Applied computing→Law, social and behavioral sciences→Economics**

## Keywords

Classification, Ensemble Methods, Financial Restatements

## 1. INTRODUCTION

"One tiny drop changes everything" was the advertising slogan of Theranos, founded in 2003, promising a technology which would detect cancer with only a drop of blood. In 2018 it revealed to be one of the most scandalous fraud cases of the last century. Although the whole scope was not known to publicity immediately, it was assumed that Theranos got a long history of misstating their finances. The SEC confirmed this later with a press release [15]. At this point the harm was already done: Reputations were forever damaged, billions of Dollars were burned and the hope in the advertised technology destroyed. The story of Theranos is not unique. Similar stories could be told about the accounting scandals of Enron, WorldCom, Tyco and many other fraudulent companies. While reading about these cases one question comes to the fore: "Couldn't this have been predicted?". There are many different perspectives and approaches to answer this question. Since various analysts and investors were deceived by the fraudsters for several years, one approach may be the use of artificial intelligence. However, not every AI method is suitable for this task, since the reason for a classification of a misstatement is at least as important as the classification itself. Finally, false detection of misstatements could also cause damage in many ways. Hence, in this paper we do not investigate the performance of neural networks, since these got a black-box character, which is a subject of current research.

Although fraud is one motivation for misstatements, it only represents a small fraction of companies. In fact, most false statements happen due to human made mistakes [16]. While some mistakes are detected and corrected quickly, others cause huge damage similar to fraudulent statements [16].

It is undoubted, that the detection of misstatements is an important field of research for all involved parties. The information that a company misstated a financial statement can make a huge difference in investment decisions. It also got a high impact on the market, especially if a misstatement was done willingly. However, the detection of false statements remains to be a difficult task, especially when trying to detect those automatically. The problem begins with a definition of a misstatement. While detected false statements are forced to be restated, unveiled ones remain hidden. This is a difficulty when applying supervised learning algorithms, which require a labeled training set. In this paper we present six machine learning algorithms of which five are supervised and one is unsupervised. All algorithms are taken out with and without

Draft

XGBoost [3], which is a representative for feature-boost algorithms. In order to train the supervised models we introduce six different definitions of misstatements, all based on restatements. We analyze about 9000 US-firms with 757 features per year from 1998 to 2017. We retrieved our data from Eikon [6], which is provided by Refinitiv (former provided by Thomson Reuters) and Audit Analytics Europe [8].

## 2. RELATED WORK

Although there are several other works which use machine-learning techniques in order to identify misstatements, none of them analyses the impact of feature-boost algorithms. Actually most of them like [4,5] use feature-sets selected by domain specialists. Being aware of the fact, that the knowledge of domain specialists can enhance the results, we added 28 features from [11]. These features had a great impact in the presented work and we assume that they could also have a positive influence in this work. In contrary to [4] and [5] we use way more features and show the impact of a feature-boost algorithm to the results. Other works try to uncover hidden misstatements [1]but do not apply their model to actual restatements. There are also works which present models for fraud detection [10]. In contrary to [10], we do not use neural networks, because of their black-box character. We assume a higher gain from results which potentially can be explained, since this could also explain false-positives. Finally there are also approaches which regard the problem from the perspective of someone who would manipulate a financial statement. One popular work in this field of research is [14]. Roychowdhury makes use of regression equations in his work. Since we want to evaluate the strengths and weaknesses of machine-learning methods in detecting misstatements, the approach of [14] is not really comparable to our approach. All in all there to the best of our knowledge there is no other work which provides an extensive machine-learning and feature-boost evaluation to the presented dataset.

## 3. DATASETS AND DEFINITIONS

In this study we make use of two different data sources. The first data source is Eikon [6] provided by Refinitiv (former provided by Thomson Reuters). From Eikon we retrieved 732 financial figures of 9263 companies from 1998 to 2017. Additionally we added 28 features, which were meaningful in [11]. We regard the financial figures as features and use them as input for the machine learning algorithms.

As stated in the introduction, we use five supervised algorithms which require a training phase. In order to realize the training, we require labeled data. For revealed misstatements, namely those which were restated, we offer six definitions. Three of those are calculated with the help of the Eikon data. The other three definitions are based on data retrieved from Audit Analytics Europe [8].

Since we can only evaluate with unveiled and corrected misstatements we make use of financial restatements. Eikon provides two versions for every financial figure: The actual figure stated by the company and a restated figure. In case a firm corrected a number, the restated figure differs from the actual figure. It must be noted though, the reason for the correction is not given by the database. In order to obtain the values the Python Eikon API offers the parameter *ReportingState*, which can be either set to *Orig* (original) or *Rstd* (restated). Audit Analytics Europe differs two different types of restatements. Those which got a positive effect and those which got a negative effect. In the following section we provide all six definitions of misstatements.

## 4. MODEL DEFINITIONS

In this section we give the six misstatement definitions. For those we solely use restatements, since these are the only misstatements which got revealed and accessible to the public. We define the restatements as follows:

1. **Eikon based definitions**
   a. **all**: If any figure has been restated in a certain year, we label the company to have misstated in this year.
   b. **relevant**: If at least one of the relevant figures has been restated by a company in a certain year, we mark this year as a misstatement for this firm. We consider the following five figures as being relevant: Net income, shareholder's equity, operating cashflow and sales.
   c. **relevant5%**: If at least one of the relevant figures has a restated value which is 5% higher or lower than the actual stated figure, we mark the statement of the to be a misstatement.
2. **Audit Analytics based definitions**
   a. **positive**: The restatement had a positive effect on the originally stated figures.
   b. **negative**: The restatement had a negative effect on the originally stated figures.
   c. **positive or negative**: The original financial statement was restated according to Audit Analytics Europe.

In order to detect the defined misstatements, we make use of six machine learning methods. Additionally, we analyze the impact of XGBoost [3], a feature-boost algorithm to the results. Three of the applied machine learning methods are classic algorithms: The K-Nearest-Neighbor classification algorithm (KNN), the Support Vector Machine (SVM) [17] and the Decision Tree [12]. In the following we denote these models as *simple*. The other three machine learning algorithms are so called *ensemble methods*. In concrete that means, that these are algorithms which combine the results of several classifiers. Therefor we applied the Random Forest [2], the Isolation Forest [18] and AdaBoost [9].

## 5. EVALUATION

In order to evaluate the performance of the machine learning methods we make use of a three-fold cross-validation and use the common measures, namely precision, recall and the f1-score. First we will have a look on the performance of the machine learning algorithms without applying XGBoost.[3]. Then we present the results with XGBoost applied before using the classifiers. In some cases some results would not give further insight, this is why we left those out. This applies especially to the first two restatement definitions of every data source. Note, that the label in the tables represent the two classes *restatement* (=1) and *no-restatement* (=0). Another important remark is that we did not tune the parameters of the machine-learning algorithms. Instead we used the proposed standard parameters from scikit-learn, a python machine-learning library [13].

### 5.1 Evaluation without Feature Boost

In this section we present the results without XGBoost being applied priorly. In Table 1 one can see the results for the first three classic models applied on all 760 features. It can be clearly seen, that the K-Nearest-Neighbor algorithm outperformed the other algorithms, although the amount of false negatives (Type II error)

is extremely high. The high precision of the SVM in this classification task can be explained with the imbalanced dataset. The SVM is actually predicting almost every data point as being a non-restatement.

**Table 1. Results of simple models regarding the restatement definition *all*.**

| Algorithm | Label | Precision | Recall | F1-score |
|---|---|---|---|---|
| KNN | 0 | 0.89 | 0.91 | 0.90 |
| | 1 | 0.66 | 0.61 | 0.64 |
| Decision Tree | 0 | 0.79 | 0.98 | 0.88 |
| | 1 | 0.60 | 0.13 | 0.21 |
| SVM | 0 | 0.78 | 1.00 | 0.87 |
| | 1 | 0.97 | 0.00 | 0.00 |

In Table 2 one can observe the results of the *simple* methods for the restatement definition *relevant*. Although the definition is stricter than the *all* definition, the results can be compared. The K-Nearest-Neighbor algorithm is again outperforming the other methods. It is also the one which actually detects the most misstatements.

**Table 2. Results of simple models regarding the restatement definition *relevant*.**

| Algorithm | Label | Precision | Recall | F1-score |
|---|---|---|---|---|
| KNN | 0 | 0.91 | 0.95 | 0.93 |
| | 1 | 0.63 | 0.49 | 0.55 |
| Decision Tree | 0 | 0.86 | 0.98 | 0.91 |
| | 1 | 0.57 | 0.16 | 0.25 |
| SVM | 0 | 0.84 | 1.00 | 0.91 |
| | 1 | 0.90 | 0.00 | 0.00 |

The last Eikon based definition is also the strictest. The results of the three *simple* algorithms can be seen in Table 3. All algorithms perform worse with this restatement definition, especially the Decision Tree tends to classify all data points as being no restatements. This leads to an extremely high precision and an even higher recall regarding the firm years which were labeled as no restatement.

**Table 3. Results of simple models regarding the restatement definition *relevant5%*.**

| Algorithm | Label | Precision | Recall | F1-score |
|---|---|---|---|---|
| KNN | 0 | 0.93 | 0.98 | 0.95 |
| | 1 | 0.57 | 0.31 | 0.40 |
| Decision Tree | 0 | 0.91 | 1.00 | 0.95 |
| | 1 | 0.00 | 0.00 | 0.00 |
| SVM | 0 | 0.91 | 1.00 | 0.95 |
| | 1 | 0.80 | 0.00 | 0.00 |

Restatements retrieved from Audit Analytics Europe [8] have an even worse detection ratio than the Eikon based definitions. Neither the restatements with a *positive*, nor the restatements with a *negative* effect can be detected well by any of the three *simple* methods. Actually all algorithms tend to classify almost every firm year to be stated correctly. This is why we did not show the results here. The only acceptable result is achieved by the K-Nearest-Neighbor algorithm and the *positive or negative* (Table 4) definition of restatements, although 1916 misstatements were not detected as such.

**Table 4. Results of simple models regarding the restatement definition *positive or negative*.**

| Algorithm | Label | Precision | Recall | F1-score |
|---|---|---|---|---|
| KNN | 0 | 0.94 | 0.99 | 0.96 |
| | 1 | 0.44 | 0.12 | 0.18 |
| Decision Tree | 0 | 0.93 | 1.00 | 0.96 |
| | 1 | 0.00 | 0.00 | 0.00 |
| SVM | 0 | 0.93 | 1.00 | 0.96 |
| | 1 | 0.00 | 0.00 | 0.00 |

Overall the ensemble methods perform better, in particular the Isolation Forest is outperforming every other algorithm. In Table 5 it can be seen, that with the strictest Eikon based definition *relevant5%* the Isolation Forest also outperforms the K-Nearest-Neighbor algorithm. This is also the case for all other Eikon based definitions. The other two ensemble methods show similar performance as the *simple* algorithms.

**Table 5. Results of ensemble models regarding the restatement definition *relevant5%*.**

| Algorithm | Label | Precision | Recall | F1-score |
|---|---|---|---|---|
| Random Forest | 0 | 0.91 | 1.00 | 0.95 |
| | 1 | 0.00 | 0.00 | 0.00 |
| Isolation Forest | 0 | 0.89 | 0.98 | 0.93 |
| | 1 | 0.89 | 0.54 | 0.67 |
| AdaBoost | 0 | 0.92 | 0.99 | 0.95 |
| | 1 | 0.54 | 0.13 | 0.21 |

The Isolation Forest also performs better with the *positive or negative* restatement definition, retrieved from Audit Analytics Europe. Comparing Table 4 and Table 6, one can see the Isolation Forest again outperforms the K-Nearest-Neighbor algorithm. Regarding the *positive* and *negative* definitions of restatements, the Isolation Forest has a similar performance to the *positive or negative* definition.

## 4.2 Evaluation with Feature Boost

In this subsection we present the feature-boosted results of the six machine-learning methods. XGBoost [3] selected only 93 of the 757 features. However, unlike one would expect this does not influence the results significantly. As you can see in Table 7, KNN profits the most by XGBoost, regarding the restatement definition *relevant5%*. Although it is losing one percent of the recall at thenon-restatement firm-years, it is gaining three percent in the classification of misstatements. Regarding the other Eikon based restatement definitions, the results are pretty similar to the one in Table 8 The Audit Analytics Europe definition of a restatement (*positive or negative*) has still a poor detection rate with the *simple*

machine learning methods. In Table 8 one can see that KNN has the maximum gain, which is three percent at detecting misstatements.

**Table 6. Results of ensemble models regarding the restatement definition *positive or negative*.**

| Algorithm | Label | Precision | Recall | F1-score |
|---|---|---|---|---|
| Random Forest | 0 | 0.93 | 1.00 | 0.96 |
| | 1 | 0.33 | 0.00 | 0.00 |
| Isolation Forest | 0 | 0.87 | 0.98 | 0.92 |
| | 1 | 0.75 | 0.26 | 0.38 |
| AdaBoost | 0 | 0.93 | 1.00 | 0.96 |
| | 1 | 0.00 | 0.00 | 0.00 |

.

**Table 7. Results of simple models with XGBoost applied, regarding the restatement definition *relevant5%*.**

| Algorithm | Label | Precision | Recall | F1-score |
|---|---|---|---|---|
| KNN | 0 | 0.93 | 0.97 | 0.95 |
| | 1 | 0.56 | 0.34 | 0.42 |
| Decision Tree | 0 | 0.91 | 1.00 | 0.95 |
| | 1 | 0.00 | 0.00 | 0.00 |
| SVM | 0 | 0.91 | 1.00 | 0.95 |
| | 1 | 0.81 | 0.00 | 0.00 |

**Table 8. Results of simple models with XGBoost applied, regarding the restatement definition *positive or negative*.**

| Algorithm | Label | Precision | Recall | F1-score |
|---|---|---|---|---|
| KNN | 0 | 0.94 | 0.99 | 0.96 |
| | 1 | 0.46 | 0.15 | 0.22 |
| Decision Tree | 0 | 0.93 | 1.00 | 0.96 |
| | 1 | 0.00 | 0.00 | 0.00 |
| SVM | 0 | 0.93 | 1.00 | 0.96 |
| | 1 | 0.00 | 0.00 | 0.00 |

There is also a rather small influence on the ensemble methods. In Table 9, one can see that the Isolation Forest slightly profits by the prior application of XGBoost, while AdaBoost has worse results than without priorly applied the feature-boost algorithm. However, the change is not significant and accounts maximum to only +0.04 for the Isolation Forest and the recall of misstatements and -0.03 for precision of the misstatements for AdaBoost.

Feature-boosting with XGBoost [3] has its highest impact on ensemble methods in combination with the *negative or positive* restatement definition. Comparing Table 6 and Table 10 one can see, that the impact on the precision of the Isolation Forest in detecting restatements is 0.07 higher with XGBoost than without it. Although the amount of detected misstatements is still very low, the precision of detecting them is also higher for the Random Forest, if applying XGBoost first.

**Table 9. Results of ensemble models with XGBoost applied, regarding the restatement definition *relevant5%*.**

| Algorithm | Label | Precision | Recall | F1-score |
|---|---|---|---|---|
| Random Forest | 0 | 0.91 | 1.00 | 0.95 |
| | 1 | 0.00 | 0.00 | 0.00 |
| Isolation Forest | 0 | 0.90 | 0.98 | 0.94 |
| | 1 | 0.91 | 0.58 | 0.70 |
| AdaBoost | 0 | 0.92 | 0.99 | 0.95 |
| | 1 | 0.51 | 0.12 | 0.19 |

**Table 10. Results of ensemble models with XGBoost applied, regarding the restatement definition *positive or negative*.**

| Algorithm | Label | Precision | Recall | F1-score |
|---|---|---|---|---|
| Random Forest | 0 | 0.93 | 1.00 | 0.96 |
| | 1 | 0.45 | 0.00 | 0.00 |
| Isolation Forest | 0 | 0.87 | 0.99 | 0.93 |
| | 1 | 0.82 | 0.26 | 0.39 |
| AdaBoost | 0 | 0.93 | 1.00 | 0.96 |
| | 1 | 0.00 | 0.00 | 0.00 |

# 5. CONCLUDING REMARKS

Our extensive evaluation has shown that the detection of misstatements of any definition presented in this paper is a difficult task. The strictness of the restatement definition has a high impact on the performance of the machine learning algorithms. Especially the KNN algorithm produced worse results, the stricter the restatement definition was. Beside the Isolation Forest, all ensemble methods were also struggling with this classification task. Our assumption is, that the reason for the results is the highly unbalanced dataset. The stricter the restatement definition becomes, the less firm-years are labeled as actual misstatements. This makes some algorithm classify all firm-years as good stated, as this is the majority class.

According to the results, the impact of feature-boosting with XGBoost [3] was rather small. However, if the same results can be achieved with 93 of 757 features this has a high impact on the runtime of the machine-learning algorithms. In addition to that the last experiment with the ensemble methods and the restatement definition *positive or negative* has shown that XGBoost actually can boost the results by a two digit number.

# 5. FUTURE WORK

Detecting restatements is an important task for all involved parties. As this survey has shown, the results have plenty of air at the top. In our opinion, the usage of neural networks is no alternative, since it is hardly possible to get insight to the decision process. In the future we would like to see other machine learning methods to be applied on the presented combination of data. These could be other clustering algorithms, like DBScan [7] or classification algorithms like Naïve Bayes.

## 6. ACKNOWLEDGMENTS

This work was partly supported by the Jürgen Manchot Foundation by funding the research Group *Decision-making with the help of Artificial Intelligence* at HHU Düsseldorf.

## 7. REFERENCES

[1] J Bertomeu, E Cheynel, E Floyd, and W Pan. 2018. Ghost in the Machine : Using Machine Learning to Uncover Hidden Misstatements. (2018), 1–32.

[2] Leo Breiman. 2001. Random Forests. *Machine Learning* 45, 1 (2001), 5–32.

[3] Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16)*. Association for Computing Machinery, New York, NY, USA, 785–794.

[4] Patricia M. Dechow, Weili Ge, Chad R. Larson, and Richard G. Sloan. 2011. Predicting Material Accounting Misstatements*. *Contemporary Accounting Research* 28, 1 (2011), 17–82.

[5] Ila Dutta, Shantanu Dutta, and Bijan Raahemi. 2017. Detecting financial restatements using data mining techniques. *Expert Systems with Applications* 90 (2017), 374–393.

[6] Thomson Reuters Eikon. 2018. Retrieved February 1, 2018 from https://eikon.thomsonreuters.com/index.html. (2018).

[7] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the 2nd Int. Conference on Knowledge Discovery and Data Mining* (1996), 226–231.

[8] Audit Analytics Europe. 2020. Retrieved December 12, 2019 from https://eikon.thomsonreuters.com/index.html. (2020).

[9] Yoav Freund and Robert E Schapire. 1999. A Short Introduction to Boosting. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence. Morgan Kaufmann*, 1401–1406.

[10] Chyan Long Jan. 2018. An effective financial statements fraud detection model for the sustainable development of financial markets: Evidence from Taiwan. *Sustainability (Switzerland)* 10, 2 (2018).

[11] B Brian Lee and William Vetter. 2015. Critical Evaluation of Accrual Models in Earnings Management Studies. *Journal of accounting and Finance* 15, 1 (2015), 62–72.

[12] Dan H. Moore. 1987. Classification and regression trees, by Leo Breiman, Jerome H. Friedman, Richard A. Olshen, and Charles J. Stone.. *Cytometry* 8, 5 (1987), 534–535.

[13] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research 12 (2011), 2825–2830.

[14] Sugata Roychowdhury. 2006. Earnings management through real activities manipulation. *Journal of Accounting and Economics* 42, 3 (2006), 335–370. https://doi.org/10.1016/j.jacceco.2006.01.002

[15] U.S. Securities and Exchange Commission. 2020. Theranos, CEO Holmes, and Former President Balwani Charged With Massive Fraud, Retrieved March 12, 2020 from https://www.sec.gov/news/press-release/2018-41. (2020).

[16] Soenke Sievers and Christian Sofilkanitsch. 2018. Financial Restatements: Trends, Reasons for Occurrence, and Consequences - A Survey of the Literature. *SSRN Electronic Journal* (2018).

[17] Alex J. Smola and Bernhard Schölkopf. 2004. A tutorial on support vector regression. , 199–222 pages.

[18] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. 2008. Isolation Forest In *2008 Eighth IEEE International Conference on Data Mining* (2008).

Draft