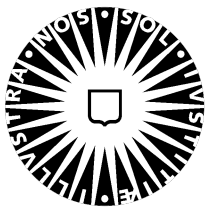

Universiteit Utrecht

Detecting Erroneous Research

Comparison of Shallow and Deep Learning Models

Joep Franssen (5659400)

Master thesis
Applied Data Science
First supervisor: Dr. Javier Garcia Bernardo
Second supervisor: Dr. Ayoub Bagheri
July 1, 2022

ABSTRACT

In the academic field, researchers are expected to publish papers on a frequent basis to stay relevant. This focus on 'quantity production' negatively affects the research quality and results in more erroneous studies. Published erroneous studies can have severe consequences as it could contribute to harmful changes in health policies or other sensitive domains. We aimed at tackling this problem by building a shallow and deep learning model that can detect erroneous research. For this binary classification problem, these models are trained on retracted and non-retracted papers. Furthermore, we split up papers in different content sections to see if some sections are more useful for predicting erroneous research. We compared the accuracy, precision, recall and F1-scores for four paper sections for a test and an external dataset. The performance measures indicate that both shallow and deep learning models can be used to detect erroneous studies to a certain extent and can potentially help journals to detect erroneous research. Furthermore, the performance measures strongly differed among paper sections and between the test and external dataset.

CONTENTS

1 INTRODUCTION

1.1 BACKGROUND AND RESEARCH QUESTION

Researchers are expected to publish regularly in journals to stay relevant and succeed in their academic careers. This phenomenon, which is often referred to as 'publish or perish', causes difficulties concerning the quality of research. Most importantly, researchers tend to focus more on quantity than quality, which might lead to errors, or even fraud, in the research methodology (De Rond and Miller, 2005).

Furthermore, journals do not only consider the scientific value of research but also consider other metrics like the estimated attention and engagement (Simonsohn et al., 2015). Because of this, some good studies with outcomes that might not excite many people might not be published. On the contrary, a low quality or even erroneous study that proposes a possible cancer medicine might be published rather easily, as these topics and findings generate much traction. This publication bias might even encourage researchers to, for example, 'adjust' their significance scores (p-hacking) to get a publication (Head et al., 2015). All in all, you could argue that this 'publish or perish' phenomenon might lead to more erroneous research.

Detecting erroneous research using shallow and deep learning models might contribute to solving this problem. Researchers committing fraud or errors are good at masking their approach, so it requires thorough investigation before errors can be detected (Markowitz and Hancock, 2016). What makes this problem even more challenging is that erroneous research ranges from studies having only small textual errors to studies making up fake results or pleading plagiarism (Marcus and Oransky, 2018). Fortunately, some models are reasonably good at detecting linguistic features, discussed topics or meta information that might indicate a research is erroneous. These models could assist editorial teams to find potentially erroneous research before publication. This would improve the scientific value of journals, since the current retraction process is inadequate for many high-quality journals (Trikalinos et al., 2008). As these models will not be flawless, they will not replace human investigation as a whole. However, they could mark potentially erroneous research for further human investigation.

The present study aimed to explore the following questions: how well can shallow and deep learning models perform in detecting erroneous research? And, secondly, are certain paper sections more useful for predicting erroneous research? For instance, the references list could be less indicative for erroneous research compared to the title and abstract section. For these tasks, we developed a shallow Naive Bayes model and a deep learning BERT model.

1.2 RELATED WORK

There has not been much research conducted on creating classifiers for detecting erroneous papers. However, in other fields, elaborate research has taken place in distinguishing fraud from genuine. Bank transactions have been explored in order to detect fraudulent credit card transactions. Given the importance of trust in monetary institutions, many studies have been dedicated to this topic. Khatri et al. (2020) developed a credit card fraud detector which was trained on numerical features like elapsed time between first and current transaction as well as the amount of transferred money.

In linguistics, fake news detection is a thoroughly discussed topic. On social media platforms like Facebook and Twitter, fake news spreads quickly and even domain experts have difficulties in distinguishing fake from real. Using quantitative methods, Ahmad et al. (2020) were able to detect fake news with more than 90% accuracy using several classifiers. They extracted different textual features from the articles using an LIWC tool and used these as input to the models. A similar study using raw text instead of numerical input was done by Jwa et al. (2019). They tried to distinguish fake from real news by analyzing the relationship between the headline and the body text using BERT. They found that the deep-contextualising nature of BERT is best suited for this task and improved the F-score over shallow learning models.

Regarding errors in papers, research has been conducted on finding characteristics of erroneous papers. Markowitz and Hancock (2016) found that erroneous papers appear to have significantly higher levels of linguistic obfuscation, including lower readability and higher rates of jargon than genuine studies. Moreover, these erroneous papers tend to use more references to make it look more genuine. Assessing the credibility of research was the main topic in the research by Alipourfard et al. (2021). They created the 'Systematizing Confidence in Open Research and Evidence' (SCORE) program which aims to assess the credibility of research claims with much greater speed and much lower cost than is possible at present. This approach differs from the present study as it assesses the credibility per claim but not regarding the text as a whole. Therefore, it might miss subtle linguistic cues that are left out of the claims which could still be indicative for erroneous research.

Research in the data science field has become much easier as elaborate datasets are now available. Regarding the present study, The Retraction Watch Database (Ribeiro and Vasconcelos, 2018) is an extensive database containing over 20,000 retracted papers. Besides, the Web of Science database provides access to multiple databases that provide reference and citation data from academic journals, conference proceedings, and other documents in various academic disciplines (Chadegani et al., 2013). These well-structured datasets can be used as input for more profound models like BERT for rather complex tasks like erroneous research detection.

2 DATA

To build an error detection classifier, we used two resources to collect sufficient retracted and non-retracted papers. We sampled the retracted papers from the Retraction Watch Database. This is a project that was started in 2010 by Adam Marcus and Ivan Oransky (Marcus and Oransky, 2018). They have collected over 20.000 retracted papers from a wide range of journals. Furthermore, we scraped the Web of Science database to get a similar amount of non-retracted papers.

To avoid label leakage due to classification on the basis of journal names, we sampled retracted and non-retracted papers from the same journals. In total, we scraped over one thousand retracted and non-retracted papers. Using the PyMuPDF package, we converted these PDF documents into raw text and saved these raw textual output to a dataframe (PyPI, 2022).

As we aimed to explore the predictive power for error detection among different paper sections, we split each paper in parts. After manually analyzing the structure of a set of retracted and non-retracted papers, we found that most papers contained a title and abstract, some middle part including the introduction, methodology and results, a discussion and/or conclusion and some references. Therefore, we decided to split the raw text in parts, namely: 1) title & abstract 2) main content (introduction till discussion/conclusion) 3) discussion/conclusion 4) references. Many of the scraped papers had different structures, but we decided to only include papers having these four sections. Regarding the third part, most papers only contained a discussion or a conclusion so we decided to split on the first appearance of either the word 'Conclusion' or 'Discussion'. However, on some occasions, this split resulted in a text section containing the content of both the conclusion and discussion.

Subsequently, we balanced the retracted and non-retracted papers for each journal so that the classifier would not be solely learning topic words or journal names, which would result in label leakage as described earlier. The final distribution of papers per journal are stated in table 1. We decided to separate the journals in two groups: a train/test dataset (364 papers from five journals) and an external dataset (264 papers from two journals). The performance on the external dataset indicated whether the patterns that the classifier finds are generalizable and not only suited to the journals it had been trained on. For the external dataset, we selected two reasonably general journals to assess the classifiers performance on a widespread set of topics. The Plos One journal covers primary research from any discipline within science and medicine. Besides, RSC Advances covers research on all aspects of the chemical sciences.

Although Bricken (2022) claims that text documents do not need much preprocessing for a BERT transformer to work well, we decided to apply some very

Journal Name	**Amount of Papers**	**Dataset**
Arabian Journal of Geosciences	94 (47R and 47NR)	Train/test
Journal of Cellular Biochemistry	156 (78R and 78NR)	Train/test
Journal of Fundamental and Applied Sciences	24 (12R and 12NR)	Train/test
Oncotargets and Therapy	26 (13R and 13NR)	Train/test
International Journal of Electrical Engineering Education	64 (32R and 32NR)	Train/test
Plos One	120 (60R and 60NR)	External
RSC Advances	144 (72R and 72NR)	External

Table 1: Journal names and the assigned dataset. In total 364 papers were used for the train/test dataset and 264 papers were used as the external dataset. R = Retracted papers, NR = non-retracted papers.

shallow preprocessing to avoid the chance of label leakage. Besides that, we also removed noise. As many retracted papers were often less recently published, they contained less recent year references. We found that these years were indicative for a paper being retracted. As we considered this as a confounder, we decided to remove all the numbers, including year references, using Regex. Furthermore, using the Spacy package, we removed the proper nouns like the author names and journal names. We also removed spaces and tabs as these can be considered as noise.

Regarding the data availability and ethicality of usage of this dataset, Retraction Watch states that the data is made available from The Center For Scientific Integrity, the parent nonprofit organization of Retraction Watch, subject to a standard data use agreement (Marcus and Oransky, 2018). Retraction Watch only provides the DOIs of retracted papers and some publicly accessible metadata like the subject, associated institution, author name, journal, retraction date and the reason for retraction. Besides, we could access the Web of Science database using our University Utrecht account to find non-retracted papers. As we do not publish any sensitive information in the present study, like names of researchers who got retracted, we do not violate any ethical rules.

3 METHODS

After we collected and preprocessed the data, we configured two different models. As stated in the research question, we aimed to to compare the performance of a shallow and deep learning model. For this study, we decided to use a shallow Naive Bayes classifier and a more advanced BERT transformer model.

3.1 BERT MODELS & NAIVE BAYES CLASSIFIERS

BERT models understand text by training on the Next Sentence Prediction (NSP) and Masked Language Model (MLM) mechanisms. Regarding Next Sentence Prediction, a model is presented with two sentences and has to decide which sentence follows the other one. The other task (MLM) consists of masking certain words in sentences and requires the model to reconstruct them based on their surroundings (Acheampong et al., 2021). By applying these two unsupervised techniques, BERT models get a deeper understanding of words in relation to their context (de Vries et al., 2019). Given the increased complexity of these models, they also require more training time and higher GPU performance. BERT is considered to be a deep learning transformer model and very good at tasks like topic classification and sentiment detection. In several studies with complicated classification tasks, like the study of González-Carvajal and Garrido-Merchán (2020), BERT models have shown to perform significantly better than shallow learning models like SVM, Random Forests or Naive Bayes.

Where BERT is considered to be a deep learning model, Naïve Bayes is a shallow learning algorithm that utilizes Bayes' rule together with a strong assumption that the attributes are conditionally independent given the class. In the context of the present study, each word can be considered as an independent feature and each feature contributes to the final classification, but the Naive Bayes model does not look at the correlation between each word. Hence it is called naive. Regarding computational complexity, training time is linear with respect to both the number of training examples and the number of attributes. This makes this model computationally way more efficient and faster than BERT (Webb et al., 2010).

3.2 PERFORMANCE MEASURES

To compare the Naive Bayes classifier to the BERT transformer, we used several performance measures. Given the confusion matrix, we calculated the accuracy (1), precision (2), recall (3) and F1 score (4). The formulas are stated below where TP = True Positives, FP = False Positives, TN = True Negatives and FN = False Negatives.

1. Accuracy = (TP + TN)/(TP + TN + FP + FN)
2. Precision = TP / (TP + FP)
3. Recall = TP / (TP + FN)
4. F1 Score = 2 * ((precision * recall) / (precision + recall))

Accuracy is a good general indicator of how well the performance is. However, this approach is rather limited in case of unbalanced data. As an example, in credit

card fraud detection tasks, the amount of fraudulent cases is rather limited so a classifier that is biased to labeling every transaction as genuine will still result in a high accuracy score. We can overcome this issue by including measures that attach more importance to the amount of false positives and false negatives. As an example, the F1 score is a measure that calculates a harmonic mean of the precision and recall and is therefore a frequently used performance measure (Rahman, 2021).

Regarding the error detection problem, you could argue that a miss of an erroneous paper has serious consequences. If people read and believe erroneous research, it might have a significant impact. Especially in some domains like medical healthcare this could even cost lives. Therefore, it is very important that the classifier is able to detect the erroneous papers very well. It is acceptable if a better true positives detection comes with a higher number of false positives, as the editorial teams should rather investigate more potential erroneous papers than missing an erroneous one. Therefore, we considered the recall score for the erroneous/retracted group an important measure. This is the amount of correctly detected erroneous papers divided by itself plus the amount of missed erroneous papers.

3.3 CLASSIFIERS INPUT

We aimed to explore whether certain paper sections are more indicative for erroneous research. Therefore, we split up the papers in four sections. In figure 1, we visualized the number of words for each of the four predefined sections. Papers with over 1000 words for a section are bundled together in the last bar. The BERT transformer can take a textual input up to 512 tokens (including a CLS and SEP token) while the Naive Bayes model does not have this token limitation. As shown, some sections have many papers containing way more than 512 tokens. Feeding longer documents, as is the case for the present study, will result in a cut-off after the token threshold. Strategies for dealing with longer texts are explored by Sun et al. (2019). Initial exploration with several different approaches showed that the head/tail strategy worked best as the start and end of a paper section often contain the most informative cues. Therefore, we concatenated the first and last 256 tokens in case a paper section contained more than 512 tokens and fed that to the model as input.

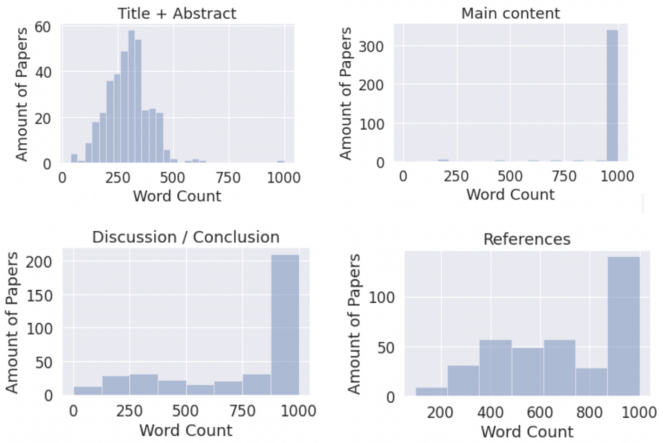

Figure 1: Word count per paper section. Papers with over 1000 words are accumulated in the last bar.

Regarding the train and test dataset, we followed the distribution that was used in a similar study (Pappagari et al., 2019). The test set contained 34% of the data and the train set was composed of all the other papers. Furthermore, we created an

external dataset that contained papers from two different journals. In this manner, we could compare how well the classifiers would perform on topics from journals it was not originally trained on. This could give more insights in the generalizability of the models.

3.4 BERT TRANSFORMER MODEL CONFIGURATION

In the present study, we used the DistilBERTForSequenceClassification model with the pretrained DistilBERT Base Cased. DistilBERT is a reduced BERT model that is 40% smaller, while retaining 97% of its language understanding capabilities and being 60% faster (Sanh et al., 2019). The applied model adds a linear layer on top of the pooled output which makes it suited for (binary) classification tasks like error detection (Huggingface, 2022). Tuning hyperparameters and finding the optimal configuration would be too time consuming and very computationally expensive. Therefore, we derived the configuration mainly from the research of Pappagari et al. (2019). Instead of using the default BERT model, we used the DistilBERT model given its faster processing times and lower GPU performance requirements. The present study's model has six hidden layers and twelve attention heads. Besides, we fitted the model with batch size 32 and five epochs. The initial learning rate is set to 0.001 and is reduced by a factor of 0.95 if validation loss does not decrease for three epochs. The Transformer is trained with the default BERT version of Adam optimizer with an initial learning rate of 5e-5. We reported the accuracy in all of our experiments. We chose a model with the best validation accuracy to calculate accuracy on the test set and external dataset (Pappagari et al., 2019).

3.5 NAIVE BAYES MODEL CONFIGURATION

We ran the Naive Bayes classifier with the default parameters. Besides, we used a TfidfVectorizer without any predefined maximum number of features. TF-IDF is a measure that expresses how original a certain word is by comparing how often a word appears in a certain document compared to how often it appears in all the documents in the dataset. In case a word appears a few times in a very limited set of documents, it seems to be very unique and the TF-IDF score will be relatively high for that word (Soucy and Mineau, 2005). However, stop words often appear in a lot of documents and as a result the TF-IDF score for a stopword will be comparatively low. These insights can be useful for any classification task. The output of the TfidfVectorizer will be the input to the Naive Bayes classifier which, in turn, can find patterns of certain words being more predictive for retracted papers or for non-retracted papers.

4 RESULTS

After collecting and preprocessing the dataset, we ran the developed models. The outcomes of the models, and the Notebook scripts, are attached in the appendix. We compare the performance of the models for the test and external dataset and analyze how they perform among different paper sections.

4.1 PERFORMANCE ON THE TEST DATASET

The performance of both classifiers on the test set are attached in Appendix 6.1. The test set contained unseen papers derived from the same set of journals as the train dataset. Given that BERT used half of the test set for validation purposes, we also halved the test set for the Naive Bayes classifier. In this way, both classifiers were tested for 63 papers.

Given that we had the same amount of retracted and non-retracted papers for each journal, so a balanced dataset, the accuracy was a good performance measure to compare both models. The accuracy for all four defined paper sections was better for the BERT model compared to the Naive Bayes classifier. Furthermore, it was interesting that the BERT transformer strongly outperformed the Naive Bayes classifier with over 25% detection difference when it came to erroneous research detection on the basis of the 'References' (fig. A4). However, the references section was the least predictive part for erroneous research as both classifiers had the worst performance for this paper section. Regarding the other paper sections, the 'Title + Abstract' (fig. A1), 'Main content' (fig. A2) and 'Conclusion/Discussion' (fig. A3), the BERT transformer also outperformed the Naive Bayes classifier strongly with over 10% higher accuracy.

When comparing the other performance measures like precision, recall and the combination of the two (F1-score), we found very similar and high scores for the BERT transformer for all these measures. However, different patterns appeared for the Naive Bayes classifier. The precision was more than 20% higher than the recall for the retracted group for the 'Conclusion/Discussion' (fig. A3) and 'References' (fig. A4). This was a result of the fact that the Naive Bayes classifier had a tendency (bias) to classify a document as non-retracted. The recall of the retracted group is the most important metric to take into consideration for this task, as a potential miss of an erroneous paper might come with severe consequences. This recall performance could be improved by further hyperparameter tuning or this could indicate that Naive Bayes classifiers are less suited to this classification task than BERT.

4.2 PERFORMANCE ON THE EXTERNAL DATASET

In general, the performance on the test dataset was better than the external dataset when considering the accuracy, recall, precision and F1-scores. This makes sense given that the train and test data are derived from the same set of journals and are about the same topics. Because of this, the classifier can detect the labels on the basis of topics that are more common in retracted and non-retracted papers. Compared to the performance on the test dataset, the accuracy seemed to be roughly 10% lower for both models and for all paper sections (Appendix 6.2).

Regarding the differences between the two models, the accuracy outcomes of both models were very comparable except for the 'References' (fig. B4), where we found 13% higher accuracy for the Naive Bayes classifier. Regarding the precision and recall measures, the BERT model had an imbalanced performance for both classes for only the 'Title + Abstract' (fig. B1) section. In contrast, the recall and precision performance were even more imbalanced between the classes for the Naive Bayes model. Especially for the 'Title + Abstract' (fig. B1) and 'Main content' (fig. B2), the precision was roughly 30% lower than the recall for the retracted group. For these sections, we saw a stronger bias towards predicting a paper to be retracted. Although this imbalance indicate a worse fit of the model, the tendency to overpredict the retracted class is desired over underpredicting in the context of this problem. As stated, a high recall should be pursued to avoid any misses of erroneous papers.

To assess the performance of the classifier, it might be more interesting to look at the external data set given that this measure tells us more about the generalizability of the model. If the classifier is able to find generalizable cues that indicate erroneous research over just detecting certain topic words it has been trained on, it would mean that a classifier is useful in practice for predicting erroneous research in more settings and for diverse topics.

5 CONCLUSION AND DISCUSSION

Regarding the research question, the accuracy indicated that both deep and shallow learning models were able to detect erroneous research to a certain extent. As expected, for both models, the accuracy was better for the test than the external dataset. This makes sense as these papers are derived from the same set of journals the classifiers were trained on. Regarding the test dataset, the Naive Bayes classifier performed worse on all paper sections than the BERT model. However, this pattern did not hold for the external dataset. The Naive Bayes classifier even outperformed the BERT model for the conclusion/discussion (fig. B3) and the references (fig. B4).

Considering the precision and recall, the BERT model had very balanced outcomes across almost all paper sections for the test and the external dataset. The Naive Bayes classifier was more vulnerable to be biased towards predicting either one class resulting in a stronger imbalance between precision and recall within classes. Further parameter tuning could be a solution to overcome this bias, although it might also be inherent to the limitation of this shallow learning model for such a complicated task. Similarly, the BERT transformer could also be improved by running the models with different values for the parameters. However, as running these models are very time consuming and computationally expensive we decided to use previous similar studies for determining the parameters.

A possible limitation of the present study could be that the models learn to relate class labels (either retracted or non-retracted) to topics rather than relying on deeper semantic and syntactic features which is called label leakage. However, the fact that the performance on the external dataset, containing widespread topics, was also better than chance, tends to indicate that the models do not fully rely on only topic detection but also on deeper linguistic cues.

Although we tried to collect as many papers as possible, we eventually used a rather limited dataset consisting of seven different journals. We divided the data in one train/test dataset containing papers from five journals and an external dataset consisting of papers from two journals. After preprocessing, we had 628 papers left. It would be better if we could have used more papers and journals to see if the findings would be similar as for this small study. Further expansion of the Retraction Watch Database would be useful to increase the study size for future studies. In the context of the present study, we could not find any numbers relating to percentages of erroneous research. Therefore, we decided to sample a balanced set of retracted and non-retracted journals although the actual distribution of retracted and non-retracted papers might be different in reality.

Regarding the BERT transformer, we used the head and tail strategy to deal with the token size limitation. We based this approach on a study by Sun et al. (2019). However, we did not compare this approach to other strategies to see which strategy would result in the best performance. This could also be an interesting angle for further research in this error detection task. Furthermore, we could argue that other transformers might be more suited to this task. Another approach could be to use other transformer models like the Longformer which does not have the token amount restriction that BERT has (Beltagy et al., 2020). Future models could be proven to be more useful for these kinds of tasks where we are dealing with long texts. Therefore, it is very likely that performance on error detection tasks will increase over the coming years. Given that the data science field is dynamic, the present study could be rapidly outmoded by more sophisticated models but the present paper's insights can be a useful starting point for future studies.

BIBLIOGRAPHY

Acheampong, F. A., H. Nunoo-Mensah, and W. Chen (2021). Transformer models for text-based emotion detection: a review of bert-based approaches. *Artificial Intelligence Review 54*(8), 5789–5829.

Ahmad, I., M. Yousaf, S. Yousaf, and M. O. Ahmad (2020). Fake news detection using machine learning ensemble methods. *Complexity 2020*.

Alipourfard, N., B. Arendt, D. M. Benjamin, N. Benkler, M. Bishop, M. Burstein, M. Bush, J. Caverlee, Y. Chen, C. Clark, et al. (2021). Systematizing confidence in open research and evidence (score).

Beltagy, I., M. E. Peters, and A. Cohan (2020). Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Bricken, A. (2022). Does BERT Need Clean Data? Part 2: Classification. https://towardsdatascience.com/does-BERT-need-clean-data-part-2-classification-d29adf9f745a. [Online; accessed 28-June-2022].

Chadegani, A. A., H. Salehi, M. M. Yunus, H. Farhadi, M. Fooladi, M. Farhadi, and N. A. Ebrahim (2013). A comparison between two main academic literature collections: Web of science and scopus databases. *arXiv preprint arXiv:1305.0377*.

De Rond, M. and A. N. Miller (2005). Publish or perish: Bane or boon of academic life? *Journal of management inquiry 14*(4), 321–329.

de Vries, W., A. van Cranenburgh, A. Bisazza, T. Caselli, G. van Noord, and M. Nissim (2019). Bertje: A dutch bert model. *arXiv preprint arXiv:1912.09582*.

González-Carvajal, S. and E. C. Garrido-Merchán (2020). Comparing bert against traditional machine learning text classification. *arXiv preprint arXiv:2005.13012*.

Head, M. L., L. Holman, R. Lanfear, A. T. Kahn, and M. D. Jennions (2015). The extent and consequences of p-hacking in science. *PLoS biology 13*(3), e1002106.

Huggingface (2022). DistilBERT. https://huggingface.co/docs/transformers/model_doc/distilBERT. [Online; accessed 28-June-2022].

Jwa, H., D. Oh, K. Park, J. M. Kang, and H. Lim (2019). exbake: Automatic fake news detection model based on bidirectional encoder representations from transformers (bert). *Applied Sciences 9*(19), 4062.

Khatri, S., A. Arora, and A. P. Agrawal (2020). Supervised machine learning algorithms for credit card fraud detection: a comparison. In *2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pp. 680–683. IEEE.

Marcus, A. and I. Oransky (2018). The Retraction Watch Database [Internet]. http://retractiondatabase.org/. [Online; accessed 28-June-2022].

Markowitz, D. M. and J. T. Hancock (2016). Linguistic obfuscation in fraudulent science. *Journal of Language and Social Psychology 35*(4), 435–445.

Pappagari, R., P. Zelasko, J. Villalba, Y. Carmiel, and N. Dehak (2019). Hierarchical transformers for long document classification. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 838–844. IEEE.

PyPI (2022). PyMuPDF. https://pypi.org/project/PyMuPDF/. [Online; accessed 28-June-2022].

Rahman, R. (2021). Supervised machine learning algorithms for credit card fraudulent transaction detection: A comparative survey.

Ribeiro, M. and S. M. Vasconcelos (2018). Retractions covered by retraction watch in the 2013–2015 period: prevalence for the most productive countries. *Scientometrics 114*(2), 719–734.

Sanh, V., L. Debut, J. Chaumond, and T. Wolf (2019). Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Simonsohn, U., J. P. Simmons, and L. D. Nelson (2015). Better p-curves: making p-curve analysis more robust to errors, fraud, and ambitious p-hacking, a reply to ulrich and miller. *Simonsohn, Uri, Joseph P. Simmons, and Leif D. Nelson (2015),"Better P-Curves: Making P-Curve Analysis More Robust To Errors, Fraud, and Ambitious P-Hacking, A Reply To Ulrich and Miller (2015)," Journal of Experimental Psychology: General 144*, 1146–1152.

Soucy, P. and G. W. Mineau (2005). Beyond tfidf weighting for text categorization in the vector space model. In *IJCAI*, Volume 5, pp. 1130–1135.

Sun, C., X. Qiu, Y. Xu, and X. Huang (2019). How to fine-tune bert for text classification? In *China national conference on Chinese computational linguistics*, pp. 194–206. Springer.

Trikalinos, N. A., E. Evangelou, and J. P. Ioannidis (2008). Falsified papers in high-impact journals were slow to retract and indistinguishable from nonfraudulent papers. *Journal of clinical epidemiology 61*(5), 464–470.

Webb, G. I., E. Keogh, and R. Miikkulainen (2010). Naïve bayes. *Encyclopedia of machine learning 15*, 713–714.

6 APPENDIX

6.1 NAIVE BAYES & BERT PERFORMANCE ON THE TEST SET

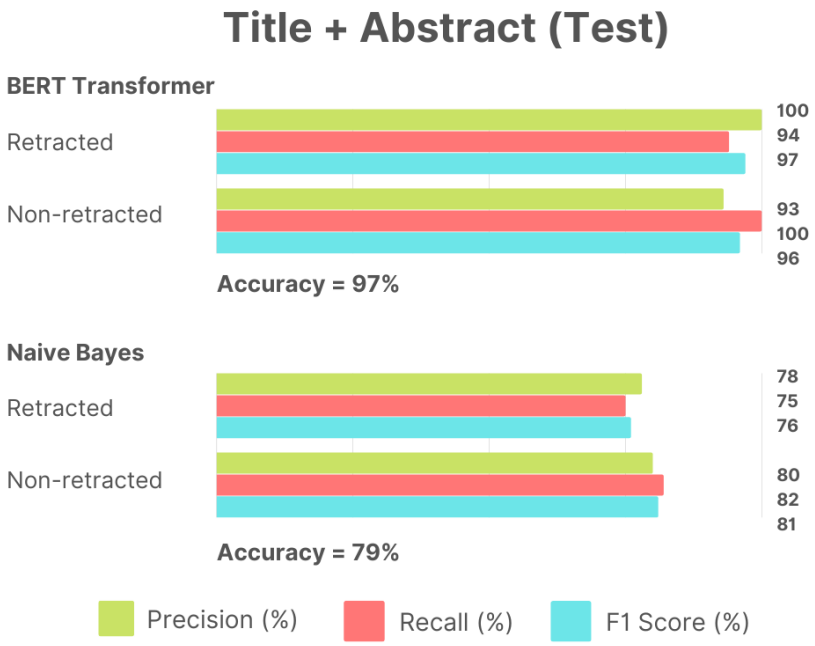

Figure A1: BERT & Naive Bayes performance on 'Title + Abstract' of the test dataset

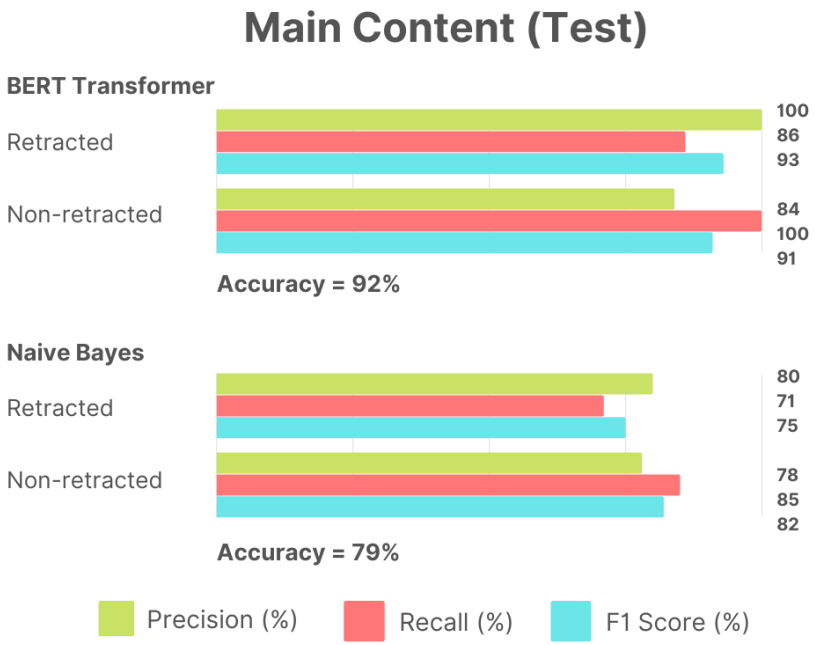

Figure A2: BERT & Naive Bayes performance on 'Main content' of the test dataset

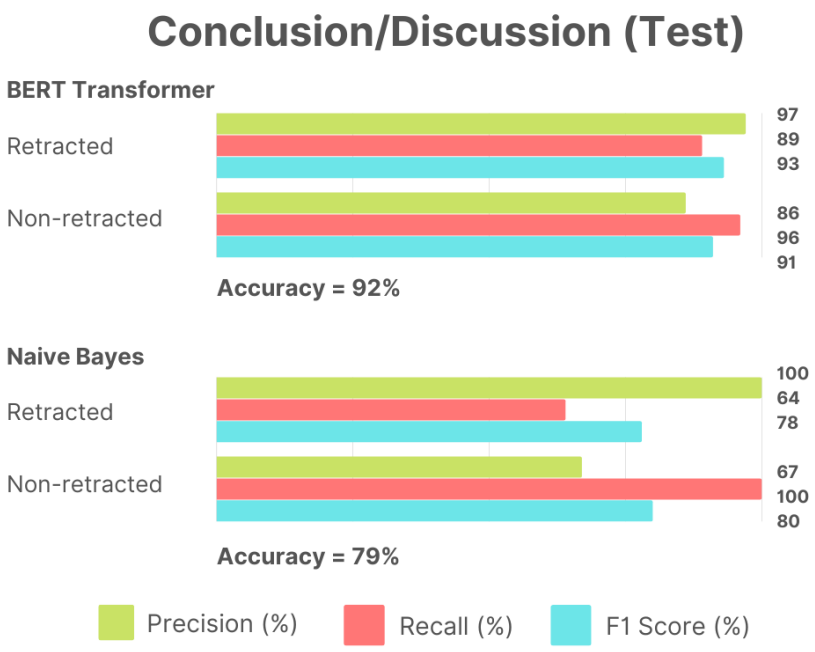

Figure A3: BERT & Naive Bayes performance on 'Conclusion/Discussion' of the test dataset

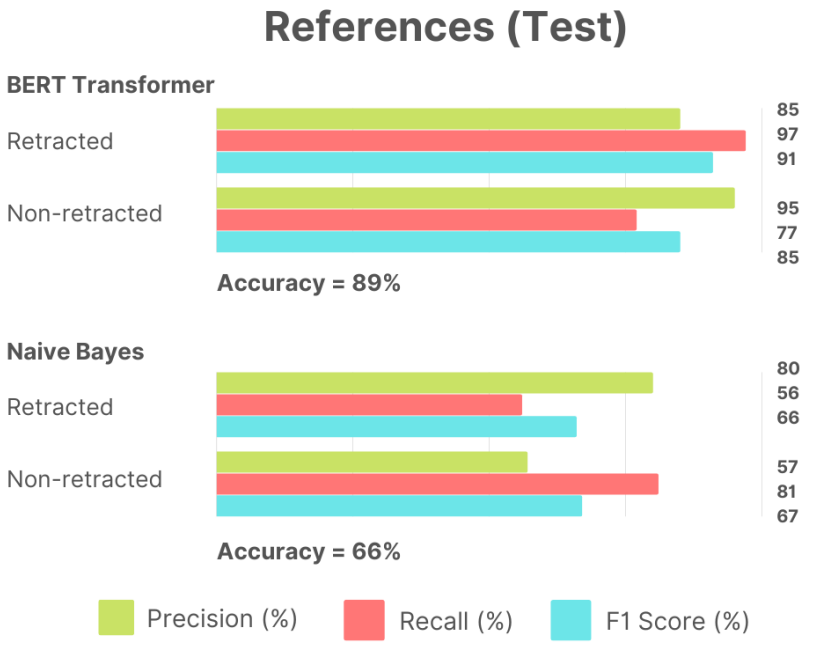

Figure A4: BERT & Naive Bayes performance on 'References' of the test dataset

6.2 NAIVE BAYES & BERT PERFORMANCE ON THE EXTERNAL DATA SET

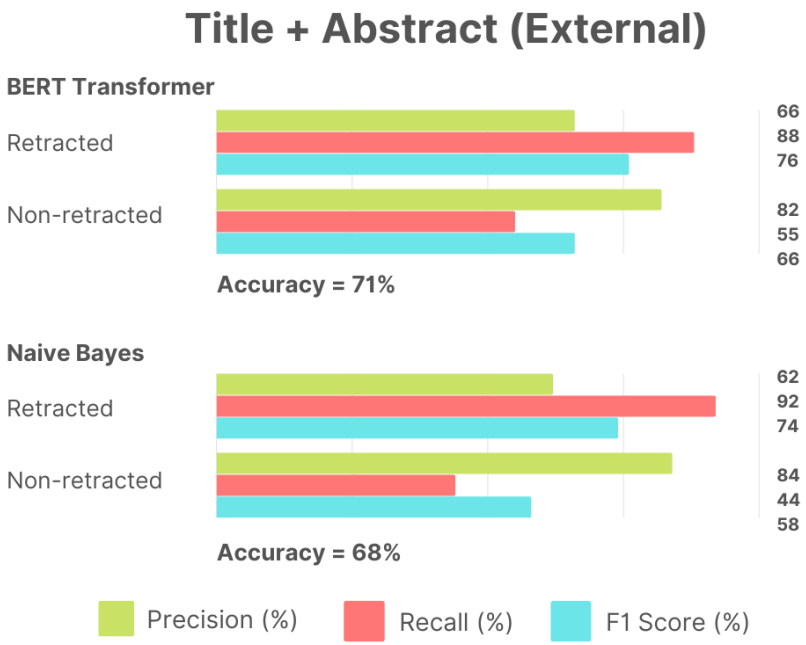

Figure B1: BERT & Naive Bayes performance on 'Title + Abstract' of the external dataset

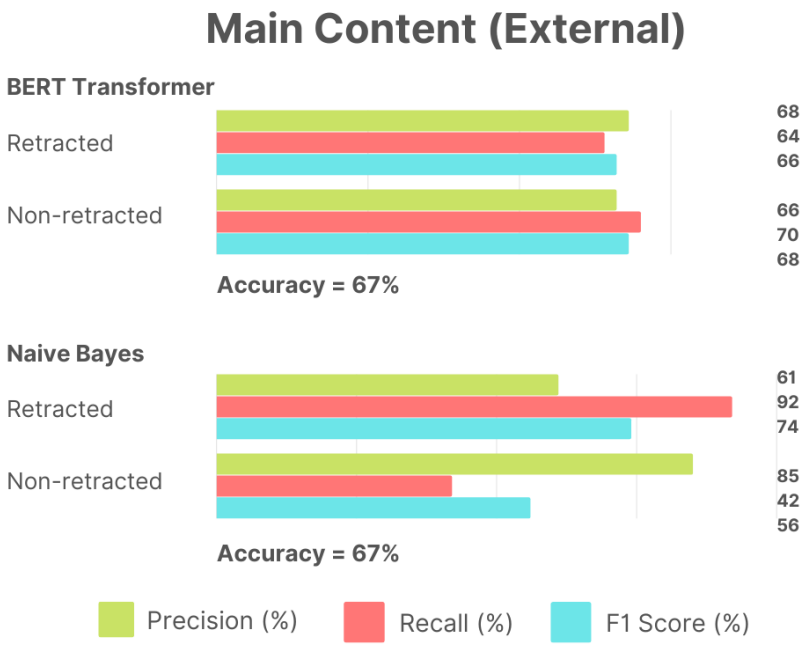

Figure B2: BERT & Naive Bayes performance on 'Main content' of the external dataset

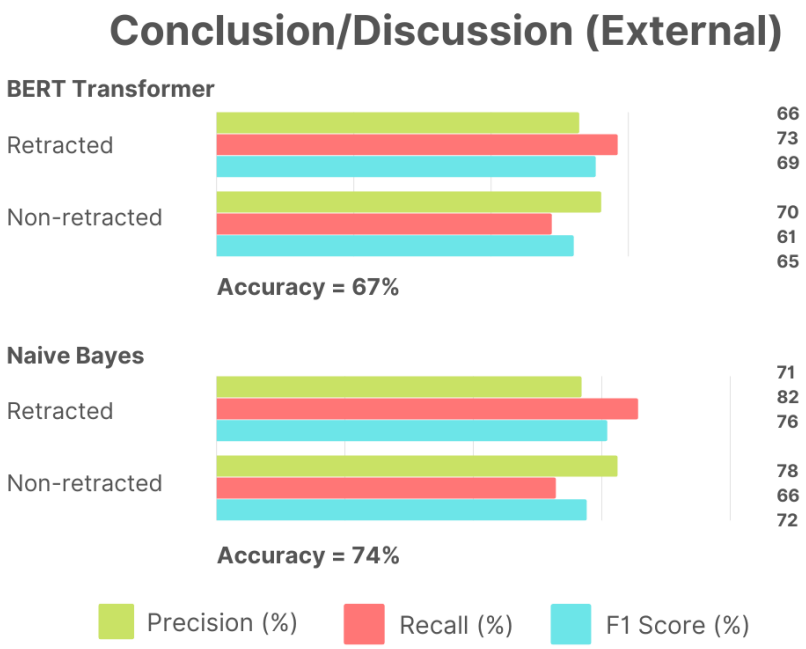

Figure B3: BERT & Naive Bayes performance on 'Conclusion/Discussion' of the external dataset

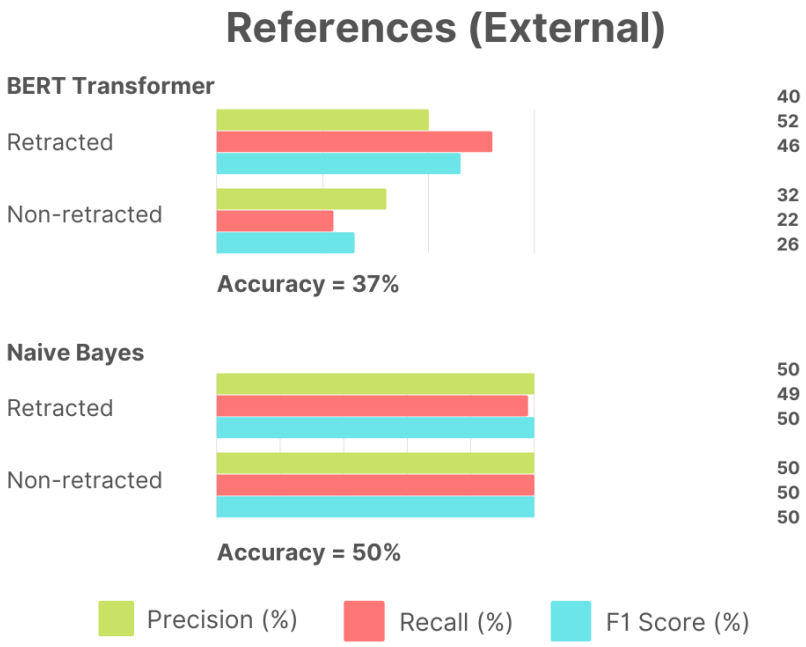

Figure B4: BERT & Naive Bayes performance on 'References' of the external dataset

6.3 NOTEBOOK SCRIPTS

The following Python Notebooks are accessible in my Github repository.
- Preprocessing notebooks
- Naive Bayes Classifier & BERT Model notebooks
- Notebooks for visualizing the word count distribution for different paper sections

Github repository: https://github.com/joepfranssen/error-detection-thesis