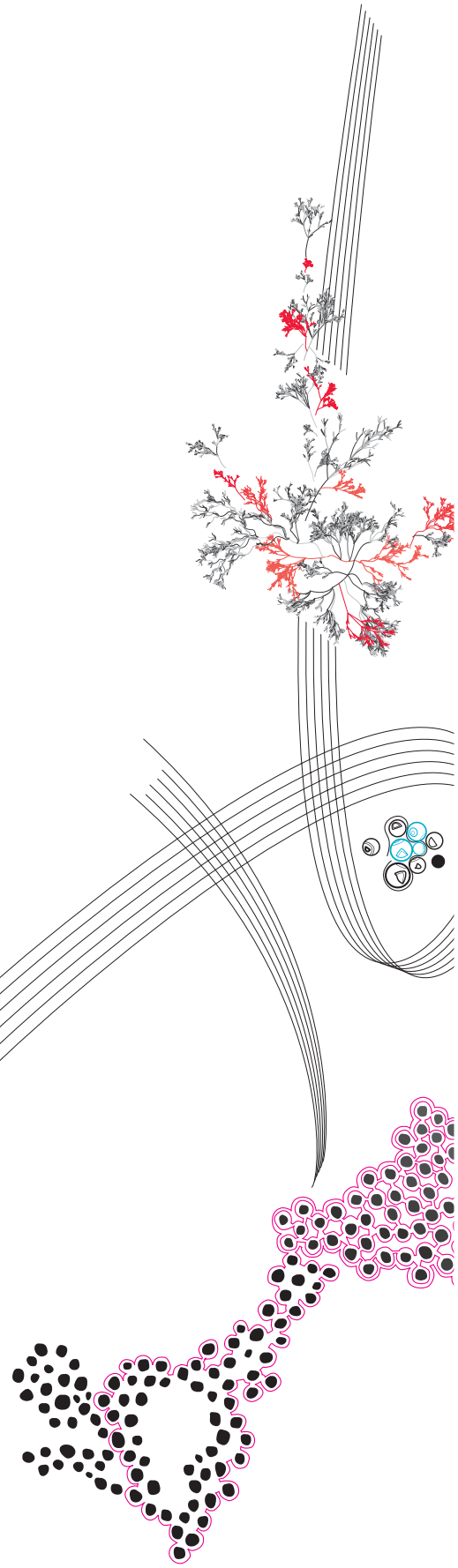

MSc Computer Science
MSc Thesis

Classifying Companies Based on Textual Webpage Data

A Comparative Analysis

Jordy Weening

Supervisors:
dr. M. Poel
dr.ing. G. Englebienne

March, 2024

Department of Computer Science
Faculty of Electrical Engineering,
Mathematics and Computer Science,
University of Twente

UNIVERSITY OF TWENTE.

Contents

Acknowledgements

At the start of this document, I would like to thank my supervisors/members of the graduation committee, Mannes Poel and Gwenn Englebienne, for their careful reading and thoughtful feedback throughout my research.

Secondly, I express my gratitude towards my supervisors at E-Active, Gerben Bosch and Marijn Otte, for their continuous enthusiasm and insightful ideas.

Lastly, I want to thank my fiancée for motivating me through the complete process, and providing me with useful tips during our bi-weekly lunch breaks.

— Jordy

Abstract

Growth of the World Wide Web consistently causes innovative ideas of companies to promote themselves and market their products. Large corporations invest many resources to achieve top spots in search queries, making it infeasible for small business owners to compete. Q-info.com, a web platform created by E-Active, offers new solutions for these companies. With their platform, it becomes easy and affordable to attract customers, sell products and manage their finances. However, Q-info.com has the same problem of getting businesses to find their platform. With hundreds of industry-specific sites, they employ a new strategy to attract small businesses. This research is done to answer the question: *What macro-precision, recall and F1-score performance is achievable with NB, SVM and BERT classifiers, determining the industry of a company using the textual data from its website?* With this information, E-Active can set up a system to classify a company by its website, and consequently invite it to that specific site on Q-info.com. This research was able to achieve macro-averaged performances of 81% in precision, and 78% on recall and F1-score. These best results were shown using an SVM classifier, predicting industries on a cleaned dataset with 178 distinct classes. This study compared the different models, tuning them to optimize precision. Additionally, a voting ensemble has been implemented to study the combined predictive power of three classifiers. Data cleaning was done by removing records, incorrectly predicted by each model, using 10-fold cross validation, this resulted in a maximum performance increase of 25 percentage points.

Keywords: Webpage classification, Naive Bayes, SVM, BERT

Chapter 1

Introduction

With the growth of the World Wide Web, as a small business, it becomes increasingly difficult to stand out between competitors. Business directories, sites that list businesses, can help position a company within a specific audience. *"Q-info.com"* [17], which is such a directory, provides its services specifically to small and local companies, offering them affordable and straightforward solutions to grow their venture. On Q-info.com, every company has their own unique profile page, which they can use to promote their organization and attract customers, specifically within their local area. Other than branding, Q-info.com offers a host of different features to support small business owners. A complete accounting system, product and subscription management, and a review module are just some of the ingredients that make up the platform. Developed by E-Active in 2006 [16], the website *"klik-info.nl"* [14] was launched together with *"klik-info.be"* [15], respectively the Dutch (NL) and Belgian (BE) domains. This was later, in 2015, expanded with a United-Kingdom (UK) site *"company-info.co.uk"* [18]. In 2022, the platform was rebranded to the current domain Q-info.com, which initiated many improvements the following years. Q-info.com spans hundreds of industry specific subdomains to sort businesses into their own category. Examples of such sites are *accountant-info.co.uk* for accountants, or *massage-info.co.uk* for masseurs.

1.1 Problem Statement

Generating traffic is an important part of managing an online platform. For Q-info.com, this comes two-fold, both businesses and consumers have to recognize it as a place to find one another. E-Active has chosen to focus on attracting companies, because many features of the platform are designed for this type of customer. While currently there is a healthy amount of active companies, for further development to be worthwhile, Q-info.com necessitates growth. E-Active has observed that interesting companies for their industry-specific sites is more effective than employing a generalized marketing approach on Q-info.com. However, the feasibility of implementing this on a large scale requires a dedicated system. Knowledge about the industry of a company is required to specifically target it. This is not a trivial task, as there is no general place to find this information. The information that is readily available, is the website of the business. The goal of E-Active is to use information found on the website of a business to determine its industry. This classification of webpages, has been done in the past based on numerous justifications. Recently, studies have shown the use case of detecting phishing based on the content of a page [26, 41]. Other attempts have shown the possibilities of classifying webpages into a domain, such as sports [43], or have shown the possibility of finding documented sources of

flash flood events [45]. These different implementations indicate the prospects of extracting information from webpages.

1.2 Research Questions

Intuitively, multiple approaches are possible to solve a webpage classification problem. As mentioned before, different solutions have been suggested in different research projects. To find a fitting solution for this specific problem, a comparative study will be carried out showing different models and experiments, with emphasis on finding the best performing model from a select set. To do this, and expand upon the current knowledge, this research aims to answer the following research questions:

> ***RQ 1**: What macro-precision, recall and F1-score performance is achievable with NB, SVM and BERT classifiers, determining the industry of a company using the textual data from its website?*
>
> > ***RQ 1.1**: What performance increase is achievable using an ensemble of classifiers compared to the individual models?*
> >
> > ***RQ 1.2**: What is the correlation between document size and the number of True Positives (TPs) and False Positives (FPs)?*
> >
> > ***RQ 1.3**: What is the correlation between training set size and classification performance?*
> >
> > ***RQ 1.4**: What is the approximated percentage of mislabeled and spam records in the dataset?*
> >
> > ***RQ 1.5**: How does performance of the classifiers change when removing mislabeled records, compared to the performance of classification with the original dataset?*

The supporting questions were formulated to give a comprehensive view of methods that could improve classification performance. Each question is intended to analyze a separate factor that was identified as interesting to research. Now, for each question a brief explanation will be given what its purpose is, and why it is important to answer.

RQ 1.1
This research will focus on three individual classifiers, as stated in the main research question. Because differences in classifiers could lead to better performance in specific parts of the dataset, an ensemble of these classifiers could increase the overall performance [34].

RQ 1.2
The dataset that is used in this research (as will be shown in section 2.3) entails a directory of company websites with their respective industry. Because each website is unique, the size of its content falls within a large range. Consequently, the hypothesis is made that a correlation could be present between the size of a document and the performance of a classifier. When a large correlation is found, it will be possible to make a statement about the confidence of classification, based purely on the amount of text that is available on the website.

RQ 1.3
Intuitively, larger training sets result in improved classification performance. As will be shown in section 2.3, the dataset that is used in this research is not balanced, resulting in industries that have many records, and industries with only a few records. Analyzing the correlation between this training size and classification performance will help to determine minimum training size requirements. These requirements could lead to a manual expansion of the training set for specific industries.

RQ 1.4
It is common for datasets to contain anomalies, such as mislabeled records or incorrect data. For this research, it is valuable to find the amount of inaccurate records, such that conclusions can be drawn about the effect of these on classifier performance.

RQ 1.5
Because RQ 1.4 analyzes the percentage of anomalies in the dataset, it is interesting to find the effect on classification performance when these are removed. This could lead to a suggestion of more research into anomaly detection and dataset cleaning.

1.3 Scientific Contribution

The scientific contribution of this research comes two-fold. It will compare webpage classification techniques, specifically on a large set of industries. As will be shown in section 2.1, current research efforts have only shown methods for few classes (2-20), which is far from the number of different labels in this research (178). Additionally, this research will show methods to perform anomaly detection on a dataset of webpages, and conclude that the classifiers in this research can be employed to improve the quality of a dataset and improve on the precision shown by state-of-the-art anomaly detection ensembles.

1.4 Requirements and Scope

1.4.1 Requirements

This thesis is a collaborative effort with E-Active, aimed at answering the research questions stated before. To meet the satisfaction of the client, certain criteria have been established for a classification pipeline. In this section, we outline the essential performance metrics the implemented system should achieve to be operationally effective.

In a production setting, the pipeline will mainly handle batch processing. While these batches can be large, there is no real-time requirement. Typically, processing within a few weeks is considered acceptable, which is also a base requirement to show any results within the timeframe of this research project.

For E-Active, certainty of correct classifications is crucial. When a company is categorized within an industry, confidence is key. Therefore, our focus is on maximizing TPs and, more importantly, minimizing FPs. This highlights the importance of the precision statistic, which decreases as FPs increase, see equation 1.1.

$$Precision = \frac{TP}{TP + FP} \tag{1.1}$$

Having a high precision is desirable, but this statistic has no correlation with the number of positive records in the dataset, meaning that a single TP could result in a high

precision score, as long as there are no FPs. For this reason, we find the recall measure in equation 1.2, where FN is used to indicate the number of False Negatives. Because of this FN factor, the recall statistic is influenced by the number of positive records in the dataset.

$$Recall = \frac{TP}{TP + FN} \tag{1.2}$$

Combining these measures, we find the F1-score as the harmonic mean, defined in equation 1.3. The F1-score is not biased towards either precision or recall, and while precision is valued higher than recall in this research, having bad performance in recall is not satisfactory. Merely optimizing for precision will reduce recall performance; therefore, the F1-score is used to show the performance of both statistics in a single metric.

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{1.3}$$

Given that future plans of Q-info.com necessitate platform growth, the client specifically emphasizes optimization for individual industries, instead of overall performance. In cases where specific industries fail to achieve the defined requirements, they may be excluded from production. A minimum precision threshold of 90% per class is set, ensuring E-Active can confidently employ the classifier in determining the industry of a company. Because improvement in precision has a negative effect on recall, a threshold of 70% is set for the recall statistic, giving the possibility to substantially scale the platform. Since only being able to correctly label a few companies is not satisfactory.

The dataset comprises two levels of labels, a top-level category and a bottom-level industry. The primary interest of E-Active is the classification of industries, as categories prove too broad for practical use. An added benefit of the system would be language independence. The capability to classify a company with a website in any language would facilitate the expansion of Q-info.com, for example into Germany. However, this is not a core requirement, and thus the initial emphasis lies on English sites. Furthermore, translation to English tends to be more straightforward than to Dutch. This approach sets the stage for potential future adaptations.

Summarizing, we find the following requirements:

1. The pipeline will primarily handle batch-processing, processing within a few weeks is acceptable.
2. TPs and FPs are the most important metrics, thus optimizing for the precision statistic.
3. Optimize classifying industries instead of categories.
4. Optimize for a subset of industries instead of overall performance.
5. 90% precision on an industry is mandatory for production use.
6. 70% recall on an industry is mandatory for production use.
7. Emphasize classifying English websites.

1.4.2 Scope

Because this study is done with a time constraint, and many possibilities for research exist on this topic. A selection has been made which parts of the subject are researched, and which parts are left out. As was already established before, requirements have already been set by the client. From these, it becomes clear that English written sites should be used exclusively. Furthermore, the research questions that have been defined, limit this research to three classification algorithms, notably: NB, SVM and BERT. The decision has been made to solely use text-based classification, this is done based on two reasons. First, because the used classifiers all have shown good results on text-based classification problems, which is important because comparison is an essential part of this research. Second, using other features such as images, or having to crawl the internet for cross-referencing hyperlinks (other sites linking to the company), would have higher computational requirements, both in terms of bandwidth and storage capacity. This is not feasible on the systems available for this research.

1.5 Document Outline

The rest of this document will have the following structure. Chapter 2 will highlight the related work and state-of-the-art solutions, and explains the dataset used in this research. Chapter 3 explains the methodology and experiment setup for this research, including hardware and software configurations. Chapter 4 shows the results. Chapter 5 highlights the limitations of this research. Final conclusions, and points of interest for further research, are given in chapter 6.

Chapter 2

Background

2.1 Related Work

The field of webpage classification is continually growing. New technological advances enable more ways to access the web, which leads to higher demand and expectations. Because of this, it becomes increasingly important to serve the correct pages to the user, dependent on its search query, interests, or other factors. Advancements in this field are made continuously, as shown by Hashemi in 2020 [24], who compared many studies in this field. Highlighting different machine learning techniques that show promising results with F1-scores up to 99%. Important to note, is the low number of classes (2-20, 2 being the most common) in the studies compared by Hashemi. Common datasets, used as benchmarks, include the WebKB project [48, 50], which offers datasets of either 4, 7 or 20 different classes [5]. Another dataset that is commonly used is the DMOZ corpus [2] which does offer a large set of classes, although most research only uses a subset of the dataset, omitting a large portion of the classes [33, 35, 42]. The study that Hashemi performed, found popular classification methods based on how many times they occurred in state-of-the-art research. It specifically established that, among others, Naive Bayes (NB) and Support Vector Machines (SVM), were popular [24]. Using NB, an F1-score of 95% was shown using information found on sibling pages [33]. SVM has been used in several studies, showing performances of up to 93% in F1-score [22, 29, 47]. Recently, in 2023, the use of Bidirectional Encoder Representations from Transformers (BERT) has resulted in decent performance, as shown by Nandanwar et al. [35], who managed to achieve 96% and 84% F1-scores on the 4-class WebKB and 13-class DMOZ datasets respectively. In 2022, a study was performed to compare different ensembles of classifiers on a range of 2-class datasets, these ensembles showed a maximum performance increase of 10 percentage points compared to the base model [34]. Now, some further details are given on these specific concepts.

2.1.1 NB

Based on Bayes' theorem [6], NB offers a probabilistic method of classifying data using supervised learning. The algorithm is built on the hypothesis that individual features are independent of each other. Using equation 2.1, it calculates the probability of a document belonging to class $C_i \in C$ (C being the set of industries) given feature set X [11, 33]:

$$P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)} \tag{2.1}$$

NB has several advantages: it is fast, easy to implement, there is no need for large training sets, and it can be used for both discrete and continuous data [8]. A disadvantage is the assumption of independence between features, thus, correlation between features will not be taken into account. NB relies on the probability $P(X|C_i)$, if we end up with a zero probability for some $P(x_k|C_i)$, where $x_k \in X$, it will return a zero probability for $P(X|C_i)$. To remedy this issue, Laplace smoothing can be used, adding a smoothing factor to both the nominator and denominator, making sure zero-division does not happen [11].

2.1.2 SVM

The SVM algorithm aims to construct a hyperplane that maximizes the separation between two classes. An example of this is shown in figure 2.1. Implementing this on a dataset, two problems can arise. The first issue arises when classification needs to be done on more than two classes. This can be solved either by a One-vs-Rest (OvR) or a One-vs-One (OvO) classification strategy. Taking n as the number of unique classes in the dataset, OvR yields n classifiers and OvO yields $n\frac{(n-1)}{2}$ classifiers. A second concern is the nonlinear nature of real-world data. Figure 2.1 shows a classification problem where a straight line perfectly separates the two classes, in real-world scenarios it is unlikely that this is possible. A kernel function can be used to map the data into a higher dimension, facilitating nonlinear decision boundaries. There are numerous kernel functions $K(x, y)$, the most notable being the RBF, polynomial and linear. Although perfectly separating the data is difficult, research has shown that textual data is often close to linearly separable, thus requiring a linear kernel [21, 23, 27, 51].

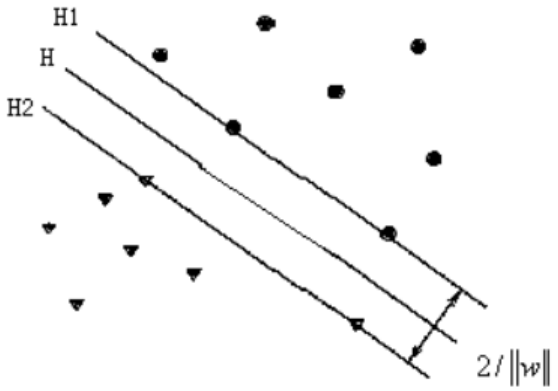

FIGURE 2.1: Optimal separating hyperplane [51].

2.1.3 BERT

In 2019, researchers from Google introduced BERT, a pretrained model that can be fine-tuned on custom datasets and machine learning problems. BERT is pretrained using an unsupervised approach and can be fine-tuned using a supervised approach. Because of this, a BERT model can be used in different configurations, such as document classification or question answering. Initially, two models were presented, $BERT_{BASE}$ and $BERT_{LARGE}$, having 110- and 340 million parameters respectively [13]. Because of its open-source nature, numerous fine-tuned BERT models have been created and are publicly available for use [1]. This research will use three of these models, notably: $BERT_{BASE}$, $BERT_{LARGE}$ and XLM-R (XLM-RoBERTa). Where XLM-R is a model developed in 2020 by Facebook

specifically trained for multilingual datasets [12]. Since this research focuses on English webpages, the assumption is made that this strategy will scale well to other languages, should this be a requirement in the future. Figure 2.2 shows a BERT representation fine-tuned for document classification. Each input sequence starts with a $[CLS]$ symbol and every sentence is separated using a $[SEP]$ token. Because BERT is pre-trained, the preprocessing steps are fixed to the model that is used. As an example, unlike NB and SVM, there is no need for stopword removal, because maintaining sentence structure is important for a BERT classifier.

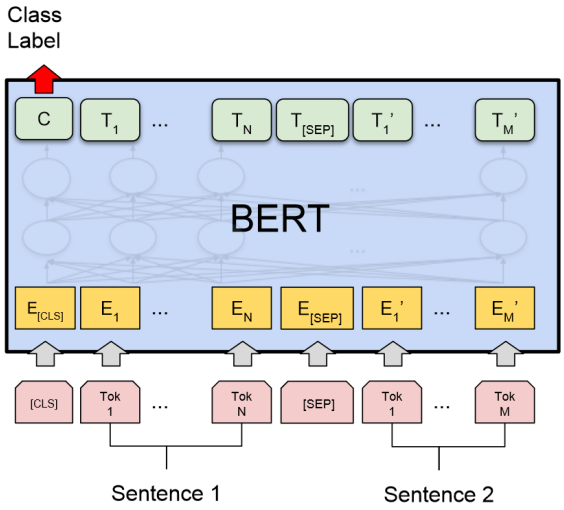

FIGURE 2.2: BERT fine-tuned for document classification, using tokenized input and outputting a class label [13].

2.1.4 Voter

In addition to the individual models, research has shown that using an ensemble of classifiers could improve classification results [34]. This is done based on the assumption that each individual model might outperform the others on specific classes or subsets of the data. We make a distinction of three different voting strategies: *hard, soft,* and *intermediate.*

Hard Voting

Also suggested by Kovačević et al. [29], a *hard* (or *majority*) voting classifier decides on a class based on the majority of votes. Each model in the ensemble makes a classification, and the voter decides upon a class by the majority. In the case of a tie, the deciding factor is alphabetical order of the classes.

Soft Voting

Because *hard* voting does not offer a proper statistical way to decide upon ties, and this situation being a frequent occurrence, a *soft* voting classifier has been suggested [34]. For this voting scheme, every model provides a probability value for each class, after which the mean is taken and the highest average class probability is chosen. This results in equation 2.2, where: c is the predicted class, n is the number of classifiers, and p_{ij} is the probability of class i predicted by classifier j.

$$c = \underset{i}{\operatorname{argmax}} \frac{1}{n} \sum_{j=1}^{n} p_{ij} \tag{2.2}$$

Intermediate Voting

Besides the well established *hard* and *soft* voting methods, we suggest an intermediate solution between these. Initially, a *hard* voting strategy is used, when no majority is found, the decision is made based on the precision scores of the original training step. This method finds a balance between cross-model confidence, and individual model confidence.

2.2 Used Concepts

Now, some concepts that are used in this research are briefly explained. It is important to note that this section will not contain extensive background information, and more reading is required if more details are desired.

2.2.1 Stopword Removal

As explained by Kaur et al. [28], stopwords are common words in a dictionary that hold little to no meaning about the subject of the text. Examples of these are: *the, a, an.* Removing these words in the preprocessing step could improve performance of the classifier because it reduces noise in the data.

2.2.2 Stemming

Stemming is the process of converting a word into its stem, this process helps reduce the size of the dictionary while preserving the meaning of each word. This can improve recall and precision performance when used in a machine learning pipeline [44].

2.2.3 Lemmatization

An alternative to stemming is lemmatization, where stemming reduces a word to its root or stem, lemmatization tries to find the dictionary form of a word by removing inflectional endings. This often leads to a more accurate representation of a word without some issues that stemming has, although it is often more computationally complex [9].

2.2.4 N-Grams

N-grams are formed by splitting a document into slices such that each slice has N consecutive words [20]. These slices can then be used in feature extraction. In this research, only 1-grams (unigrams) and 2-grams (bigrams) are used.

2.2.5 TF-IDF

Term Frequency Inverse Document Frequency (TF-IDF) is a measure that gives importance of a word to a document, that is part of a larger corpus. The full equation $tfidf(t, d, D)$ for term t, document d and corpus D is shown in equation 2.3. Where $f_{t,d}$ is the number of times term t occurs in document d [30, 24].

$$tfidf(t, d, D) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}} \cdot \log \frac{|D|}{|\{d \in D : t \in d\}|} \tag{2.3}$$

2.2.6 Chi2 Feature Selection

Based on the χ^2 statistic, the chi2 feature selection algorithm is used to choose the most relevant features from a large set. This makes training a classifier more efficient, since only a small set of important features are used as input, instead of the complete document [25].

2.2.7 Micro- and Macro-Averaged Statistics

Research often discriminates between micro and macro performance [47, 49, 50]. Micro-averaged performance is calculated by aggregating the counts of TPs, FPs and FNs across all classes and then calculating the performance measures. Macro-averaged performance is determined by first calculating the performance for each class individually, after which the results are averaged. So micro-averaging gives similar weight to each record and macro-averaging gives similar weight to each class. In this research, the focus is on optimizing individual class results, consequently macro-averaging is used.

2.2.8 McNemar's Test

In 1947, McNemar proposed a test to compare the predictive accuracy of two models [32] (later extended by Edwards [19]). This test formulates the null hypothesis that none of the two models shows better performance than the other. Thus, the alternative hypothesis being that the performances are not equal. Calculating the test (χ^2) statistic is done using equation 2.4, where B and C can be extracted from the contingency table of the models' predictions (see table 2.1). This statistic can then be converted to a p-value, which is compared to the chosen significance level α to determine if the null hypothesis can be rejected.

		Model 2	
		Correct	**Incorrect**
Model 1	**Correct**	A	B
	Incorrect	C	D

TABLE 2.1: Example contingency table distinguishing between correct and incorrect predictions of two models (variable names used in this example).

$$\chi^2 = \frac{(B - C)^2}{B + C} \tag{2.4}$$

2.3 Dataset

This research project makes use of a private dataset provided by E-Active. In this section, first, a top-level description will be given about the dataset and its potential issues. Then we continue upon this, and try to remedy some peculiarities that are found. Since Q-info.com is a business directory, the dataset contains information about all companies on the platform. The information of a business that is available for this research entails: *Name,*

Website address, Industry, Category, Country. Where the content located at the website address is used as input, and the industry is the desired output. A top-level category is also provided, subdividing into the industries. In total there are 363,191 records with 12 top-level categories, 220 distinct industries and 3 countries. A few fictional examples of these are shown in table 2.2. Some observations result from this extraction, notably the data contains some diversity in formatting. Several URL structures are used, from which, some are invalid (see *http::www.jassal.nl*). Furthermore, the database contains empty data. Another thing to notice is the records being labeled in Dutch, some of these Dutch industries will be referenced to in this thesis; when of actual importance to the reader, the translated version will be used instead. When looking at more examples of the dataset, more anomalies arise. Not all will be emphasized here, instead we conclude that extensive preprocessing is a mandatory step and continue upon this in the next section. A company registered on Q-info.com automatically becomes part of the dataset, there are some spam filters in place, but E-Active has suspected them not being fault proof. They consequently hypothesize that numerous records in the dataset are either spam, have a non-existent web-address or have an incorrectly labeled industry. The client estimates that 1-2% of the records are incorrectly labeled and that a maximum of 15% of the records can be considered spam (either by having a website that does not exist or having content on the website that does not represent an industry on Q-info.com). This estimation is based on expert knowledge, and there has not been an empirical analysis of this before this research.

Name	**Website address**	**Industry**	**Category**	**Country**
Restaurant Havik	*https://www.restauranthavik.nl*	eetgelegenheid	eetgelegenheid	NL
Deo Sure		woninginrichting	dienstverlening	UK
Motor Peter	*http://motorpeterberg.be*	motor	winkel	BE
Jassal	*http::www.jassal.nl*	tuinarchitect	dienstverlening	NL

Table 2.2: Fictional examples of records in dataset.

2.3.1 Data Preprocessing

Having a base dataset provided by E-Active is paramount to develop a classification system. The dataset, as outlined, is sourced from businesses upon their registration and includes vital information such as the website addresses. It is imperative to note that this data, while valuable, lacks verification, potentially leading to integrity concerns. To address this, preprocessing measures have been implemented based on specific criteria that will be elucidated in the subsequent sections.

Out of the initial dataset of approximately 300,000 records, several issues were identified that need to be addressed. From these, requirements can be determined, and the data can be preprocessed. Since the data comes from businesses, there are various mistakes in important attributes such as the site address. As a result, some of these sites cannot be accessed and human errors are found within the records. While some of these errors might be easy to identify and fix, others might be less trivial. Another issue with the dataset is that multiple records exist with the same website. Having companies listed as two different industries can lead to complex classification scenarios, this also needs to be addressed. Additionally, because Q-info.com has existed for a few years, some data might no longer be relevant. Companies might have gone out of business, resulting in sites that are either empty or up for sale. Intuitively, for these cases, there is no correlation between the site and the industry. Lastly, a common issue with any online platform is

spam. E-Active has highlighted two industries that contain a high percentage of incorrect data. These industries, accountants, and associations from the UK, have been targeted by spammers, and consequently, the client has advised against using data from these specific sites.

In summation, the following five requirements have been identified:

1. No duplicate records
2. Ensuring reachability of URLs
3. Exclusion of pages labeled as "for sale"
4. Removal of pages with insufficient content
5. Exclusion of data from accountants and associations in the UK

Now a short explanation for each requirement is given, specifically how they were enforced in the preprocessing stage, and the effect it has on the dataset.

No duplicate records

Duplicate records occur in various forms in the dataset, thus requiring different approaches for resolution. Some instances have straightforward solutions, while others require a deliberative process. There are numerous cases where a company is registered two or more times on the platform. By mapping these occurrences to tuples (a, b), with $a, b \in C$ and C being the set of industries, we determine the frequency of each occurrence. For cases where $a = b$, one record is simply discarded, and the other kept, resulting in the removal of about 23,000 records. In situations where $a \neq b$ a three-fold decision-process is applied:

1. Keep record with industry a
2. Keep record with industry b
3. Discard both records

This mapping is illustrated in appendix B.1, where for other combinations, that are not specified, option 3 is chosen. This selection has been decided upon by the client, who used their domain-specific expertise to determine cases where a general decision could be applied. At the end of the process, the dataset is reduced by approximately 35,000 records.

Ensuring reachability of URLs

Classifying the industry of a company, relies on data from the businesses' website. When analyzing the base dataset, there are several instances where the input data is slightly incorrect. Notably, the site address, which is important for this research, has many records with inaccuracies. Common occurrences include substitution of a comma for a period, or incorrect top-level domains such as ".nl.nl". After resolving such issues, the site HTML is requested using the urllib3 python library [3]. Despite numerous techniques to maximize the number of successful responses, such as configuring the correct user-agent and disabling SSL checks, about 65,000 sites either failed to respond, or returned an error status.

Exclusion of pages labeled as "for sale"

When companies go out of business, often their domain ends up "for sale". A default page is put up to notify visitors about the availability of the domain. Needless to say, these pages should not be included in classification. Since it is not feasible to manually check every page, some smart filtering steps can be applied to test if a page is for sale. E-Active has established a list of common phrases which occur on these kinds of sites, comparing the textual content with this list, 4,000 records can be removed. Appendix B.2 shows the list that was used.

Removal of pages with insufficient content

The dataset of scraped pages, while having only successful responses, still has some records that are not usable for classification. We identify these by the data returned from the response. Analysis by random selection shows that responses with only little data returned, lack important or key facts about the company and its industry. Therefore, a minimum size requirement of 25kb has been set. This margin was chosen, based on the observation that beyond this boundary, some meaningful information could be seen in the files. 13,000 files are within this range, and thus removed from the dataset.

Exclusion of data from accountants and associations in the UK

E-Active has identified a source of spammers on the accountants and associations industries for UK companies. Many companies in these industries are either nonexistent or incorrectly labeled. For this reason, these 5,650 companies have been omitted from the dataset.

2.3.2 Data Storage

After all processing steps have been completed, some 160,000 records remain. The essential attributes we store are: *Market, Category, Industry, Id, Site HTML, Website address and Name*. Notably, the HTML data is stored as a text file within the data directory organized by market, category, and industry. This results in the following directory tree:

```
data
├── annotations.csv
├── UK
│   ├── Advies
│   │   ├── Bedrijfsadvies
│   │   │   ├── 208359
│   │   │   ├── 208838
│   │   │   └── ...
│   │   └── ...(See Bottom-Level in appendix A.3)
│   └── ...(See Top-Level in appendix A.3)
├── NL
│   └── ...
└── BE
    └── ...
```

The *annotations.csv* document defines the link between files and annotating data. This CSV-file contains the columns: *Id, Filename, Market, Category, Industry, Website address, Name*. The "filename" attribute denotes the location of the corresponding HTML data file, ensuring the availability of relevant content.

2.3.3 Data Analysis

Having established a dataset that fits the requirements, we now show some visualizations of the different dimensions and aspects of the data. In table 2.3, the different dimensions of the dataset are shown. In total, 161,685 records remain, spread across 220 industries. These industries fall into 1 of 12 categories, this is shown in appendix A.3.

Dimension	**Size**
Countries	3
Top-level categories	12
Bottom-level industries	220
Records	161,685

TABLE 2.3: Dimensionality of the dataset.

As noted before, the platform is currently active in three different countries. As shown in figure 2.3, there is no uniform distribution across these different markets. Where the UK has more than 80,000 active companies, we find less than 20,000 in Belgium. A cause of this could be the focus on Flanders (the Dutch-speaking part of Belgium), which eliminates a lot of potential companies, and the smaller size of Belgium compared to the United Kingdom and The Netherlands.

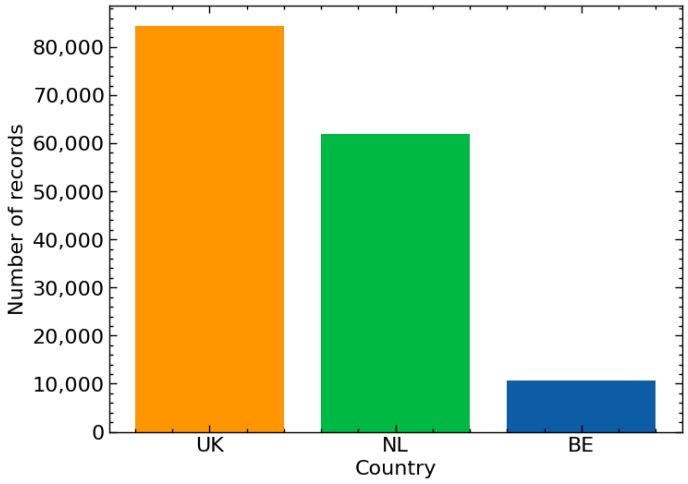

FIGURE 2.3: Distribution of active companies across the three countries.

The platform is divided into 12 categories, to show the distribution across these, figure 2.4 shows the number of sites in each category. Besides showing the number of sites, it also shows how much of a category is occupied by a specific market. We conclude that the UK has the largest share in most categories, except for *techniek* and *uiterlijk*. We again conclude that Belgium is not prominently present, especially in the lower performing categories. A next step is looking into the distribution of records within a category. We find similar results, where the UK is predominantly present. A complete overview of the data is shown in appendix A.1.

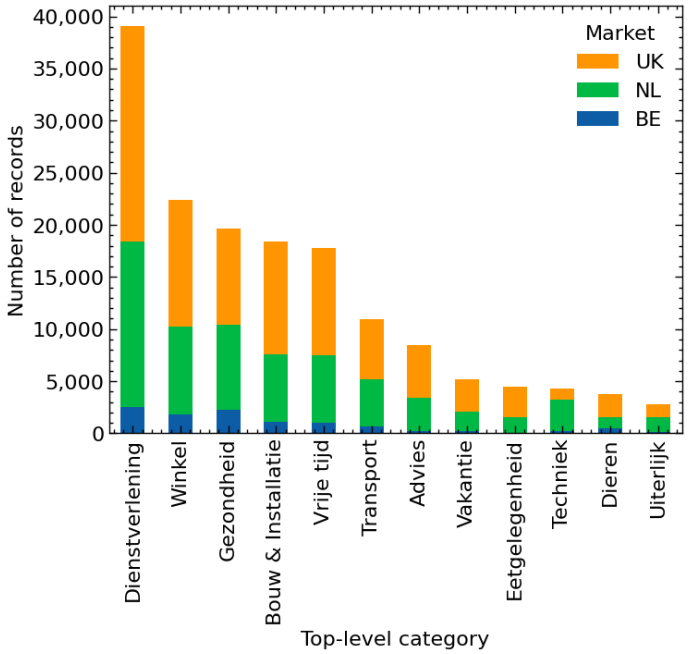

FIGURE 2.4: Distribution of top-level categories.

During preprocessing, it was noted that documents in the dataset should have a minimum size. In figure 2.5 a histogram is shown displaying the different document sizes. To mitigate the influence of outliers, for this plot the data has been truncated at 1,000kb.

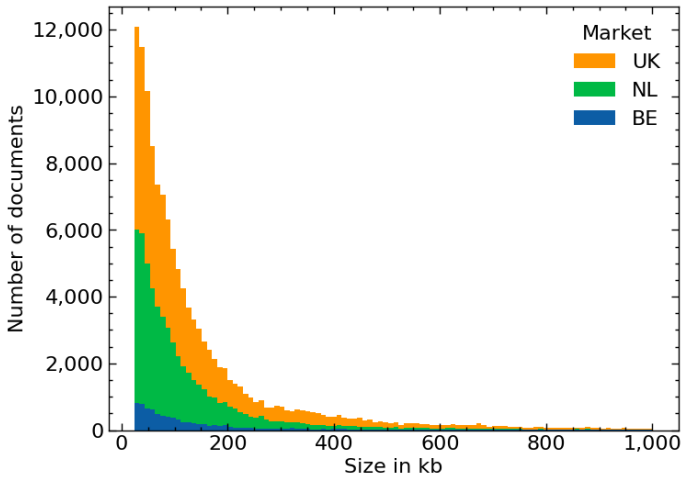

FIGURE 2.5: Distribution of file sizes across markets (truncated at 1,000kb).

2.4 Training Dataset

As defined in section 1.4.1 and 1.4.2, this research will focus on textual classification of English sites. Therefore, a mirror dataset has been created, removing all content that is deemed out-of-scope. The new dataset, containing only companies on the UK market, has been stripped of all HTML tags using the beautifullsoup python package [40]. The HTML tag removal requires a reconsideration of dataset requirement 4, since a page made up solely

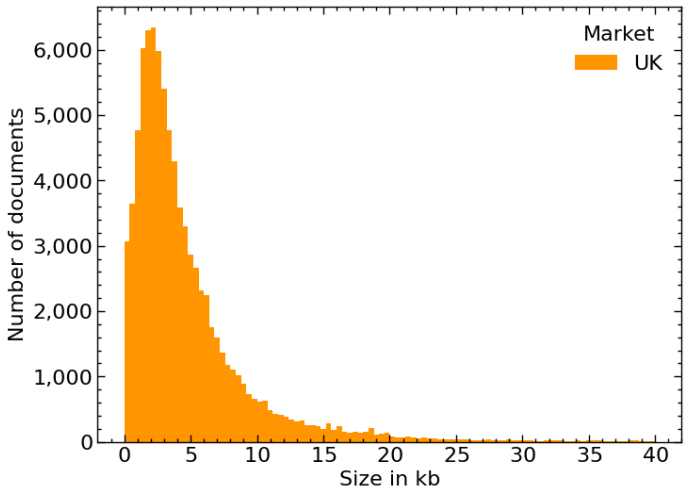

FIGURE 2.6: Distribution of file sizes in mirror dataset (truncated at 40kb).

of images will result in an empty record in the mirror dataset. To decide on a margin, a random selection of documents at different sizes was taken, and their content was analyzed. It was found that documents with less than 100 bytes of data had no correlating information with the industry. Hence, a new filtration step is done to remove any document containing less than or equal to 100 bytes of data. In order to verify that the chosen margin has been appropriately selected, a further analysis of the impact of file size on classification performance will be conducted, as mentioned in sections 3.3.2 and 4.2.1. The new size distribution is shown in figure 2.6, where the outliers (documents larger than 40kb) have been removed to accurately visualize the dataset. The new dimensions of the dataset are shown in table 2.4. Because of the focus on the UK market, roughly 84,000 records remain, spread across 178 industries. The full distribution of businesses across industries and categories on the UK market is shown in appendix A.2. From the full visualization, it becomes clear that an imbalance is present across the industries. Additionally, when looking at the remaining labels, it should be noticed that they are not mutually exclusive (a restaurant may also do takeaway, or vice versa). This last observation will have some impact on performance, however, it is difficult, and beyond the scope of this research, to determine the significance of this. The rest of this document will assume use of the mirror dataset, meaning it only contains textual data without HTML tags.

Dimension	**Size**
Countries	1
Top-level categories	12
Bottom-level industries	178
Records	84,476

TABLE 2.4: Dimensionality of the mirror dataset.

Chapter 3

Methodology

3.1 Experiment Setup

Three different classification models have been decided upon to be used in this study. Increasing in complexity, these models are NB, SVM and BERT. These models were picked because the literature shows promising results for them [29, 33, 35, 42, 47]. For both NB and SVM, TF-IDF features are used, BERT however, is supplied with a tokenized version of the documents. Additionally, a voting ensemble is implemented using the three models. Each implementation requires a unique pipeline with unique hyperparameters, the optimization process of this is done using a sequential tuning strategy, with macro-precision being the metric to optimize. Now, for each classifier, it will be explained how they are implemented and what hyperparameters are tuned.

3.1.1 NB

The NB model that is implemented, consists out of a set of layers. Stepping through the process, we find: preprocessing, feature selection, feature extraction and finally, a classification layer. These layers have diverse implementations and can be tuned based on the dataset and classification problem. A selection of parameters has been made that are experimented on to find the best performing model. In table 3.1 these ranges are shown. Because of the discrete nature of the data, a multinomial implementation is chosen. Laplace smoothing is used to avoid issues when features are not present in the training set, resulting in a zero probability.

Layer	Option	Values
Preprocessing	**Stopword removal**	Without, With
	Lexical normalization	None, Stemming, Lemmatization
	NGrams	Unigrams, Bigrams, Both
Feature extraction	**TF-IDF min document frequency (Nr of documents)**	0, 5, 10, 20, 30
	TF-IDF max document frequency (In terms of %)	10, 80, 90, 100
Feature selection	**KBest features**	$K \in \{100, 500, 1000, 1500, 2000, All\}$
	Feature scoring method	chi2
Classification	**Classification algorithm**	Multinomial Naive Bayes

Table 3.1: Options experimented with on NB classifier.

3.1.2 SVM

Similar to the NB implementation, the SVM model will consist out of the layers: preprocessing, feature selection, feature extraction and a classification layer. This same structure accommodates similar hyperparameters to tune. Additionally, some SVM specific parameters are experimented on, all are shown in table 3.2. As classification strategy, the OvR method will be used. As shown in section 2.1.2 there is a stark difference in the number of classifiers required for OvR and OvO. Because of the many classes in our problem, this decision has been made.

Layer	Option	Values
Preprocessing	**Stopword removal**	Without, With
	Lexical normalization	None, Stemming, Lemmatization
	NGrams	Unigrams, Bigrams, Both
Feature extraction	**TF-IDF min document frequency (Nr of documents)**	0, 5, 10, 20
	TF-IDF max document frequency (In terms of %)	90, 100
Feature selection	**KBest features**	$K \in \{100, 500, 1000, 1500, 2000, 2500, All\}$
	Feature scoring method	chi2
Classification	**C value**	0.1, 1, 10
	loss	squared_hinge, hinge
	penalty	l2
	dual	True, False
	Tolerance	1e-2, 1e-3, 1e-4
	Class weight	None, balanced
	Degree (for polynomial kernel)	3, 4
	Kernel	Linear, Polynomial, RBF

Table 3.2: Options experimented with on SVM classifier.

The NB and SVM classifiers will be implemented in python. For stopword removal, the English stopword dictionary provided by the NLTK package [10] will be used. It is common to additionally remove domain-specific stopwords [7], however in this research, the domain is not well-defined because of the diversity of companies, consequently this step is omitted. Stemming or lemmatization will be implemented using the Porter stemming algorithm, or the WordNet lemmatizer respectively, both provided by the NLTK package [10]. Tokenization into unigrams, bigrams or both is then done and resulting TF-IDF vectors are extracted. To reduce the set of features, the best features are selected using the chi2 scoring algorithm. As a last step, classification is performed on the remaining features. Feature extraction, selection and classification, are done using the Scikit-Learn package [37].

3.1.3 BERT

Table 3.3 shows the options that have been experimented on. Due to time constraints and the computational cost of training a BERT model, the selection of hyperparameters to tune is smaller compared to the other models in this study. Because Devlin et al. [13] gives a suggestion of hyperparameters to use, we choose a batch size of 16 (because 32 is not computationally feasible on the system). While a maximum of 4 epochs is expected in

fine-tuning, a limit is set to 7, although the process will end when there is no improvement in validation loss during 2 epochs.

Layer	Option	Values
Feature selection	**Maximum sequence length**	512, 200, 100
Classification	**Learning rate (Adam)**	5e-5, 3e-5, 2e-5
	Warmup proportion	0.05, 0.1
	Model	xlm-roberta-base, bert-large-uncased, bert-base-uncased

TABLE 3.3: Options experimented with on BERT classifier.

The BERT model will be implemented using the PyTorch machine learning package [36]. Additionally, the transformers package [46] is used for tokenization and optimization using the Adam algorithm. Learning rate is scheduled to start at 0, then increase linearly for the warmup proportion up to the specified learning rate. Then, linearly decreasing again each training step (batch) until it is 0.

3.2 Train, Validation, Test Split

A three-way split has been carried out on the dataset to train, validate and test the models following scientific standards [39]. To create the split, for each label, 80% of the data was randomly selected as training data. The remaining data was again, for each label, randomly split into two groups (both 10% of the original data) validation and test respectively. Hyperparameter tuning is done based on the classification performance on the validation data. Final results are shown based on the test data.

3.2.1 Decision Criterion

Because a range of configurations result from the set of hyperparameters that have been chosen, now a criterion is defined making it possible to choose the best performing configuration. The best configuration is determined based on a combination of the macro-averaged precision and recall scores. Because a balance of the two is required, we find the F1-score using equation 1.3 (this is not the true micro- or macro-averaged F1-score, but will give enough information to decide on a configuration). Initially, we decide based on this harmonic mean. Then, if a tie between configurations occurs, precision, and recall are valued individually in this order. If these additional steps also result in a tie, the configuration with the least constraining parameters is chosen. The hyperparameter tables 3.1 to 3.3 show the values increasing in order from least constraining to most constraining, from left to right (ordered by intuition).

3.2.2 Voter

After establishing the base models, the best configuration for each model will be decided upon based on the previously stated decision criterion. We then implement a voting ensemble using the three different strategies *hard, soft* and *intermediate* shown in section 2.1.4. Consequently, the best strategy is chosen based on the same decision criterion, initially based on the F1-score calculated using equation 1.3, when multiple strategies show a tie, the best model is chosen based on the precision and recall statistic, in that order.

3.3 Visualizations

We now define a set of visualizations that will aid in comparing the models and show the results achieved.

3.3.1 Pipeline

As defined in section 3.2.1, a criterion has been established to decide which configuration of hyperparameters is best. From these, a pipeline is created which shows all aspects of a model. For each model (NB, SVM, BERT and Voter) a diagram will be created to show the established pipeline.

3.3.2 Effect of Document Size

Having a dataset that is sourced from public websites, results in a broad distribution of document sizes. Previously shown in section 2.3, the document sizes range from 0.1kb to 40kb (outliers not accounted for). Intuitively, more textual content contains more information about the class of a document. To validate this hypothesis and show a possible correlation, the relationship between TPs, FPs and document size will be analyzed by plotting the data onto a bar chart. The analysis will be done based on the results gathered from the test data.

3.3.3 Effect of Training Set Size

Machine learning, relies on training data. Because the dataset used in this research is not uniformly distributed across the different classes, it is interesting to find out if there is a correlation between the amount of training data and its performance. Furthermore, a difference can be present across classifiers, where one could need more data than another. To analyze this, a range of scatter plots will be created showing the test data performance metrics of an industry respective to the amount of input data.

3.4 Dataset Mislabeling Detection

Previously, the assumption has been made that several anomalies in the dataset might influence the classification performance, see section 2.3. Furthermore, the methods that have been used thus far, are considered unsatisfactory in removing these types of irregularities, since they are unable to verify incorrect labels, or identify all spam sites. To refine the dataset, a technique is used to detect mislabeled data [38]. Using the established configurations of NB, SVM and BERT based on the specified criterion in section 3.2.1. We use 10-fold cross validation to get a classification on every record in the dataset. This is done using the following procedure: First, the train and validation sets are combined. Then, each item is assigned a random number $i \in \{1, \ldots, 10\}$ that indicates the fold where it will be part of the validation set. Now that a classification for each individual record can be achieved, the common incorrect results (companies for which each model fails to classify the actual label) are found. Unlike the proposed technique [38], in our research, the cross validation is only run once instead of 10 times. This is done because of the computational cost of running a BERT classifier. To test the assumption, 200 random samples are taken from the common incorrectly classified results (assuming there are at least 200 common incorrect results). Then through a web portal, illustrated in figure 3.1, these 200 websites

are displayed subsequently, after which the user can decide on one of the displayed options (left to right):

- The actual label is correct.
- One of the model predictions is correct.
- All shown industries are valid for the company (i.e. the business is active in multiple industries).
- None of the displayed industries is valid for the company.
- The website is either spam, non-existent, or returns an error.

Using the results from this random sampling approach, the percentage of anomalies in the data is statistically estimated. When a significant amount of anomalies is present (>= 25% of the sample test), the classifiers are retrained using a subset of the data (omitting the common incorrect results) and the difference in performance is analyzed.

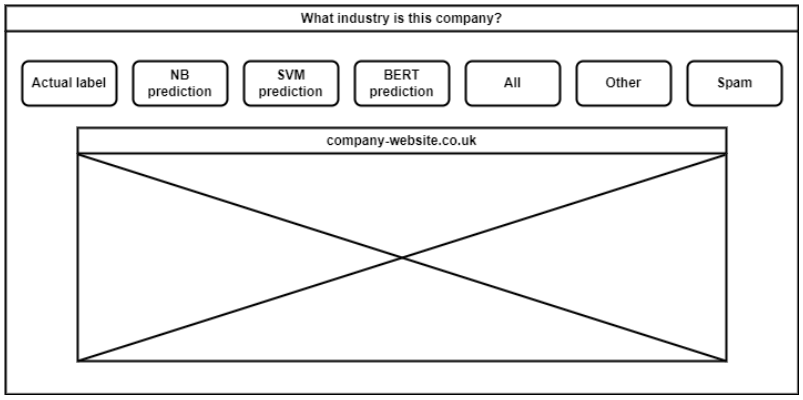

FIGURE 3.1: Wireframe of decision system used to hand label common incorrect results.

3.5 Metrics

Analyzing the performance and comparing the different models is done using a multitude of statistical measures. A difference between this thesis and other state-of-the-art research is the focus on per-class performance instead of the overall metrics [31, 34, 49]. The main needs of E-Active are to achieve excellent performance in a selection of classes, where most literature tries to improve averaged statistics. Because of this, a distinction will be made between overall statistics and per-class performance, where both will highlight the common (macro-averaged) metrics: precision, recall, F1-score and accuracy. Additionally, as was noted in section 3.4, 10-fold cross validation is performed. To validate that this process has been implemented correctly, the mean and standard deviation values of the cross validation statistics are also provided. These metrics are expected to improve slightly on the base test and validation sets, since the models will be trained on 81% of the original data instead of 80%, which is the base train set. To test if a statistically significant difference is found, the McNemar's test is used, because it works with limited data (see section 2.2.8 for the explanation). The null hypothesis is defined as (H0): *There is no significant difference in performance between the two classifiers*, the alternative hypothesis is defined as (H1): *There*

is a significant difference in performance between the two classifiers. The null hypothesis is rejected when the resulting p-value is less than the chosen significance level α of 0.05. As a final remark, any time a percentage increase is mentioned, it specifically refers to an increase in percentage points, unless otherwise stated.

3.6 Hardware and Software

Experiments are run on systems following the specification in table 3.4. Two systems are used to train the models, with the main difference being the presence of a GPU. Since BERT is considerably larger than the other models, it is not feasible to run it without using a GPU.

Model	CPU	RAM/Swap	GPU	Software package
NB	Intel Xeon 2nd gen	4 GB/16 GB	None	Scikit-Learn [37]
SVM	Intel Xeon 2nd gen	4 GB/16 GB	None	Scikit-Learn [37]
BERT	Intel i5-12400F	32 GB	GeForce RTX 3060	PyTorch [36]

Table 3.4: Hardware and software specification.

Chapter 4

Results

This chapter will show the relevant results and findings from the experiments defined in the previous chapter. Section 3.1 mentioned that the different models are tuned based on a set of hyperparameters such that the best model can be chosen. These chosen models will first be highlighted, after which, their results are compared. As noted before, the most important metric to optimize is macro-precision. However, other metrics such as F1-score and accuracy are used to show a more comprehensive view of the performance.

4.1 Model Implementations

For the models NB, SVM, BERT and the Voting ensemble, the best-found configuration will now be shown and explained.

4.1.1 NB

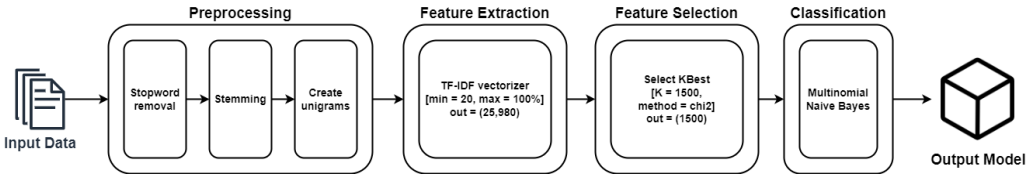

FIGURE 4.1: Naive Bayes pipeline, options shown between "[ ]", output features shown between "( )".

Recall table 3.1 where we defined the hyperparameters that have been experimented on using a Naive Bayes classifier. After running all experiments that are shown in appendix C.1, the model that showed the best results based on the decision criterion (see section 3.2.1) is shown in figure 4.1. Specifically, we find the model performing best using preprocessing with stopword removal, stemming and unigram creation. Using only TF-IDF features present in a minimum of 20 documents, and finally, selecting the best 1,500 features completes the pipeline for NB.

4.1.2 SVM

Appendix C.5 shows all configurations of hyperparameters that have been experimented on for the SVM classifier. The resulting pipeline of best parameters is shown in figure 4.2. When comparing it to the NB pipeline, we find several differences. Where NB achieved the best results using stemming, SVM was found to perform best using a lemmatized form

of the input. Furthermore, SVM shows that it is less susceptible to noise because the minimum document frequency is 5 compared to 20 for NB. This results in a much larger batch of features, from which 2,500 are extracted as a final input. The model was found to perform best using a linear kernel, which was already hypothesized in section 2.1.2. The SVM specific options were found to perform best in their default configuration, according to the Scikit-Learn specification [4].

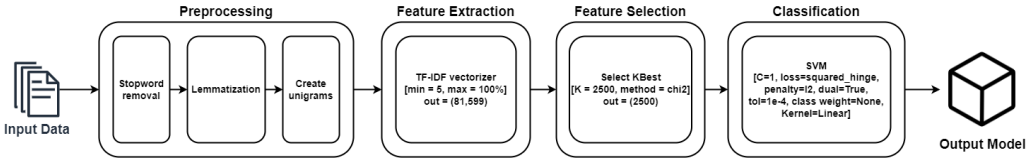

FIGURE 4.2: SVM pipeline, options shown between "[ ]", output features shown between "( )".

4.1.3 BERT

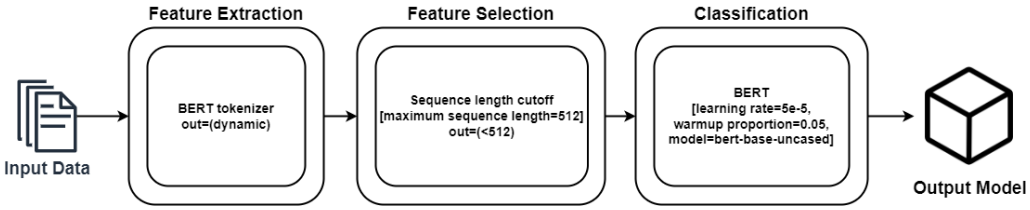

FIGURE 4.3: BERT pipeline, options shown between "[ ]", output features shown between "( )".

Figure 2.2 shows the best performing BERT pipeline resulting from the experiments documented in appendix C.9. Using the maximum sequence length of 512 tokens, a learning rate of 5e-5 and warmup proportion of 0.05, the best performance was achieved on the $BERT_{BASE}$ model. Figure 4.4 shows the progression of the validation, training and test loss during training, and additionally the boundary where the model was detected to start overfitting. As expected [13], this occurs after three epochs, although validation and test loss do not decrease significantly after the first epoch. The results in the rest of this document are based on the model after three epochs of training.

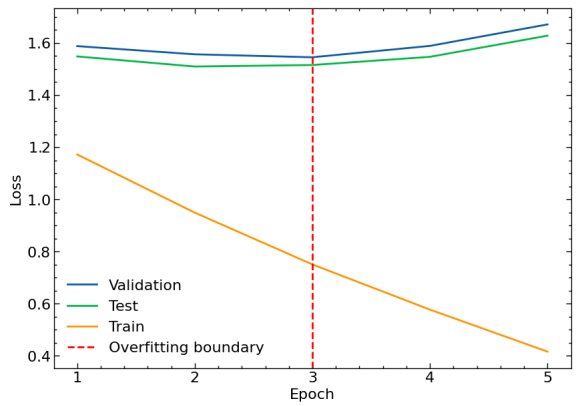

FIGURE 4.4: Validation, train and test loss of BERT model during training.

4.1.4 Voter

Recall that three voting strategies have been experimented on to combine the different classifiers into an ensemble. The results of these individual experiments can be found in appendix C.13. The best results were found using a *soft* voting strategy. Shown in figure 4.5, each classifier supplies the voter with the individual class probabilities, then the best class is chosen.

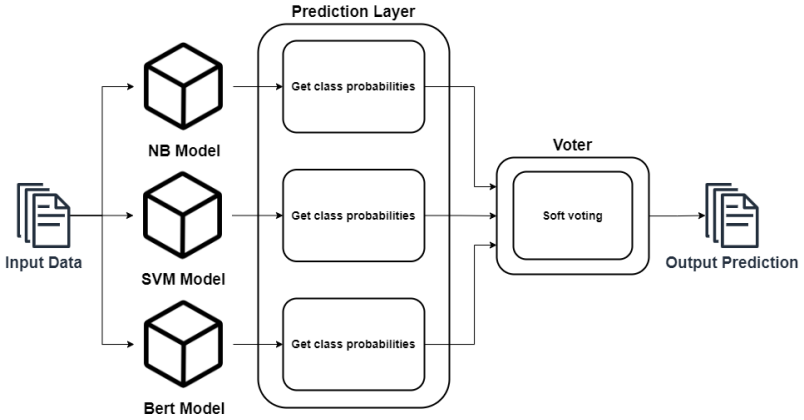

FIGURE 4.5: Voting pipeline using an ensemble of NB, SVM and BERT classifiers.

4.2 Comparison

Now that four models have been established, the results will be analyzed and compared. We first analyze the general statistics, after which, the relevant class-specific results will be shown. Lastly, general tests explained in 3.3 are carried out. Table 4.1 shows the performance measures of each best-performing configuration (based on the decision criterion, see section 3.2.1). It is clear that NB is the worst performing model across all measures. SVM and BERT both show 10-30% better performance in every statistic. Comparing these last two using McNemar's test, we fail to reject the null hypothesis, indicating there is no significant difference in test performance. Disregarding the train set, a maximum disparity of 2% is found, with BERT showing slightly better results in most statistics. And SVM showing a minor improvement of 1% on the precision statistic. The best performing model is the voting ensemble, showing better results in every performance measure, although it is only ahead by a margin of 1-3%. Comparing it to SVM and BERT using McNemar's test, we are able to reject the null hypothesis (indicating a significant difference in performance). Recall section 3.5, where the hypothesis was made that cross-validation scores would improve upon validation and test statistics. As found in the results, this is the case, with a maximum performance gain of 6%. Performance measures for each individual cross-validation fold can be found in appendices C.2, C.6 and C.10.

		Precision	Recall	F1-score	Accuracy
NB	**Train**	0.57	0.40	0.41	0.61
	Validation	0.51	0.37	0.38	0.57
	Test	0.46	0.36	0.37	0.56
	10-Fold (μ/σ)	0.50/0.01	0.39/0.004	0.40/0.004	0.59/0.005
SVM	**Train**	0.76	0.68	0.69	0.74
	Validation	0.62	0.58	0.58	0.65
	Test	0.60	0.57	0.57	0.64
	10-Fold (μ/σ)	**0.65**/0.01	0.61/0.01	0.61/0.01	0.69/0.005
BERT	**Train**	**0.79**	**0.73**	**0.73**	**0.80**
	Validation	0.61	**0.59**	0.58	0.66
	Test	0.59	**0.59**	**0.58**	0.65
	10-Fold (μ/σ)	**0.65**/0.009	**0.62**/0.01	**0.62**/0.008	**0.70**/0.006
Voter	**Validation**	**0.64**	**0.59**	**0.59**	**0.67**
	Test	**0.61**	**0.59**	**0.58**	**0.66**

TABLE 4.1: Performance metrics of the different models (the best results are shown in bold).

Global statistics are generally important to optimize for research purposes, since they show the overall performance of a model, making it comparable to other research. However, for this research, it would suffice for only a subset of classes to have excellent performance. Therefore, now some class-specific performance measures are shown. In section 1.4.1 some key performance requirements were defined. Specifically, a precision of 90% and a recall of 70%. When taking the subset of classes that satisfy these requirements, we end up with the data shown in figure 4.6. In total, there are 26 industries that show sufficient performance in at least one model, from which: NB satisfies 9 industries, SVM and the voting ensemble satisfy 18 industries, and BERT satisfies 19 industries. For the client, a high number of satisfactory classes is desired, BERT manages to satisfy the most, notably, 11% of the 178 industries in total.

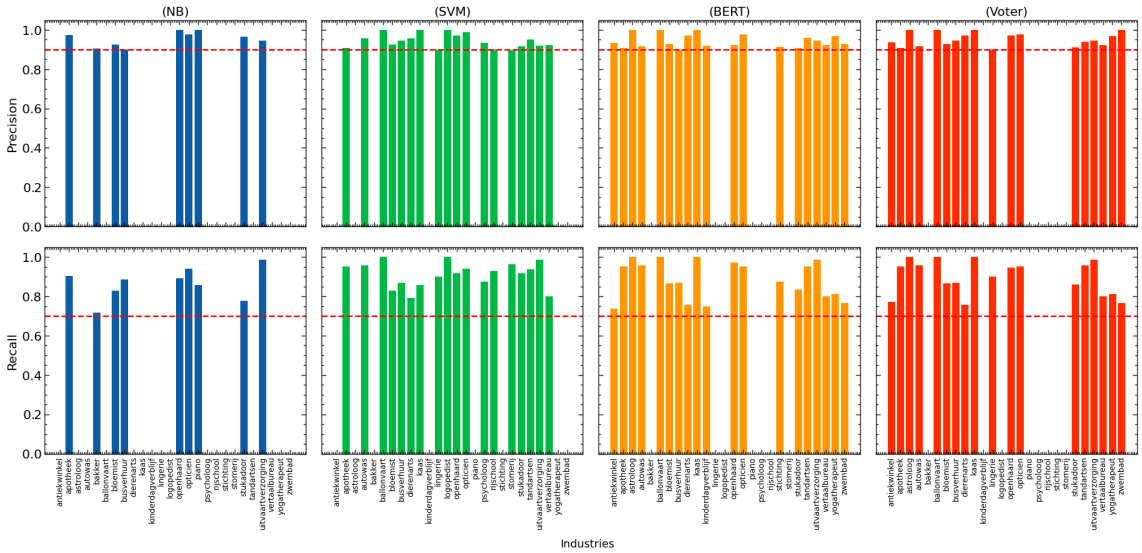

FIGURE 4.6: Industries satisfying the required performance measures based on test classifications, (required performance shown as horizontal line).

Detailed per-class performance measures can be found in appendices C.3, C.7, C.11 and C.14, where for each model the best performing industries are shown with their respective statistics. Appendix C also shows confusion matrices for each model. The key finding from these results is that NB only shows decent performance in industries with high support, whereas the other models also satisfy low support classes, such as astrology. The NB classifier has several classes that contain many FPs, identifiable by the horizontal lines in the confusion matrix, these are not as prevalent in the other models.

		Correct	Incorrect	
NB	**Unique**	1,505	7,646	
	Not Unique	43,958	22,991	
	Total	45,463	30,637	76,100
SVM	**Unique**	1,875	1,125	
	Not Unique	50,479	22,621	
	Total	52,354	23,746	76,100
BERT	**Unique**	3,416	1,902	
	Not Unique	49,702	21,080	
	Total	53,118	22,982	76,100
Common		40,931	17,700	

TABLE 4.2: Number of correct and incorrect classifications for each model based on 10-fold cross validation. Unique results are only classified respectively by that model, common results are classified respectively by each model.

When looking into the classifications acquired by running 10-fold cross validation, the results shown in table 4.2 are found. Combining the train and validation set yields 76,100 records, each of these resulting in a prediction for each model. Some records are correctly predicted by every model, others might only be correctly predicted by a single model. The table shows these different situations, for example: NB is the only model able to correctly classify 1,505 specific documents. Conversely, it incorrectly classifies 7,646 documents, for which at least one other model was able to correctly predict its label. 54% of all documents are correctly classified by every model, 23%, or 17,700 documents, are incorrectly predicted by each model. BERT manages to correctly classify the most documents, SVM also shows good results.

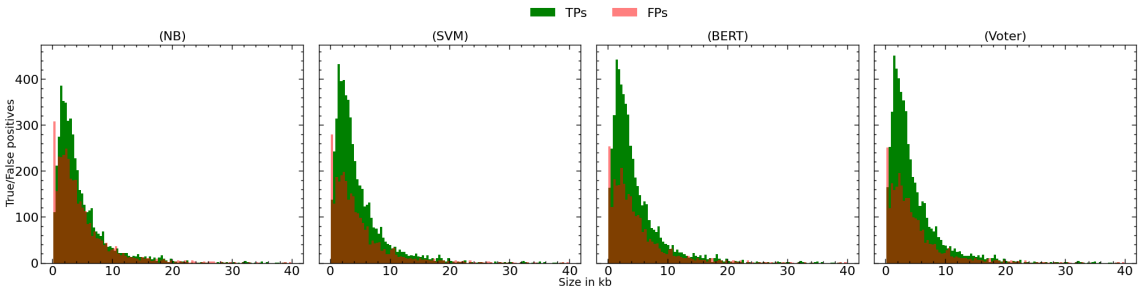

FIGURE 4.7: Number of TPs and FPs relative to document size for each model (truncated at 40kb).

4.2.1 Document Size

In figure 4.7 a histogram is shown, visualizing a possible correlation between TPs, FPs and the size of an input document. We observe a similar distribution for each model, albeit with a higher overall accuracy observed for SVM and BERT. From the visualization, a similar distribution between TPs and FPs can be observed, this shows that document size has no substantial impact on the classification accuracy. However, for documents in the smallest size range (<400 bytes), an increased amount of FPs are found. Although, documents with little content have been removed in preprocessing, still there are some that lack the necessary information to classify them. As for the voting classifier, it closely resembles the results from the BERT model. This demonstrates that this group of classifiers might not work well together in an ensemble, given their similarity in positive classifications.

4.2.2 Training Set Distribution

A closer look into the effect of training size on performance, yields the data shown in figure 4.8. The different industries are scattered onto a plane, where the horizontal axis shows the number of training samples, and the performance is shown vertically. Comparing NB with the other classifiers, there is a clear difference present. NB shows a larger correlation and is consequently more dependent on a large training set. This correlation is also present for SVM and BERT, although it is less noteworthy, since these classifiers also show high performance on some industries in the lower range. Appendix C.16 shows additional scatter plots that highlight the lower and upper range of training samples (<= 400 samples and >= 400 samples). The most notable finding is a negative correlation of the precision statistic on the NB classifier in the upper range. This indicates that large training sets add noise to the probabilistic function, causing FPs to increase. This also explains why the issue does not affect the recall statistic.

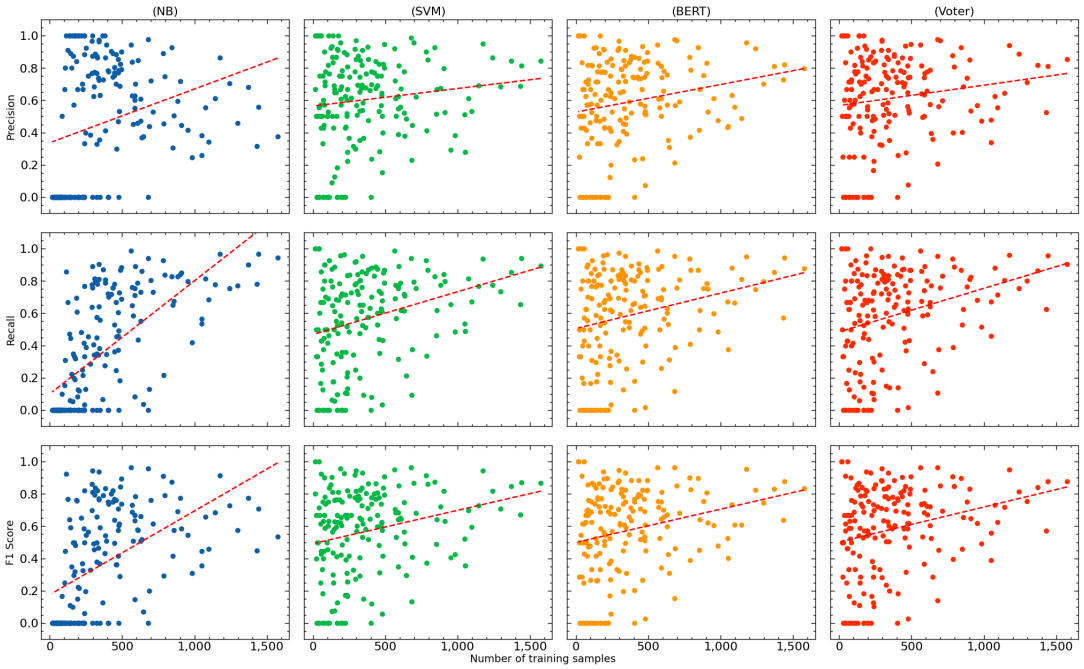

FIGURE 4.8: Relationship between test performance and the number of training samples for each model (each dot is an industry).

4.3 Dataset Refinement

In table 4.2, 17,700 documents were shown to be incorrectly classified by each model. Recall section 3.4, where the assumption is made that these records contain a high percentage of irregularities. Specifically, incorrect labeling or spam sites. A random sampling of 200 records has been taken from the common incorrect results to test this assumption. Intuitively, the original label can either be: correct, incorrect or the site is considered as spam. Additionally, a label predicted by a model might be considered correct. As an example, a company listed as a restaurant may also do takeaway, making a prediction as takeaway also valid. This results in seven different splits:

Type 1: Records that were correctly labeled, and additionally, no model prediction is considered correct.

Type 2: Records that were correctly labeled, and additionally, all model predictions are also considered correct.

Type 3: Records that were labeled incorrectly, and no model predicted the correct label.

Type 4: Records that were labeled incorrectly, and additionally, one model predicted the correct label.

Type 5: Records that were labeled incorrectly, and additionally, two models predicted the correct label.

Type 6: Records that were labeled incorrectly, and additionally, all three models predicted the correct label.

Type 7: Records that were considered as spam, either by the site not existing or having a label that is not present on the platform.

Table 4.3 shows the results from this test. From the 200 incorrect records, a total of 111 records should have been considered correct (type 2, 4, 5 and 6). 21 records (type 3) were found to be mislabeled, but no model was able to correctly predict its true label. Additionally, 40 records have been identified as spam, resulting in a total of 172 anomalies or an 86% fault rate in the sample, which is higher than the threshold of 25% set in section 3.4. This fault rate of 86% is also the precision metric of the anomaly detection method, which improves upon the suggested technique that achieved a precision of 70% [38].

Type	Description	Records
1	Actual correct, and all models incorrect	28
2	Actual correct, and all models correct	**16**
3	Actual incorrect, and all models incorrect	**21**
4	Actual incorrect, and one model correct	**15**
5	Actual incorrect, and two models correct	**21**
6	Actual incorrect, and all models correct	**59**
7	Spam	**40**

TABLE 4.3: Sample test results of common incorrect results, anomalies shown in bold.

We now define new datasets that are created by first omitting the 17,700 incorrectly classified records from the train and validation sets. This results in some industries being removed completely from the training set, consequently these are removed from the

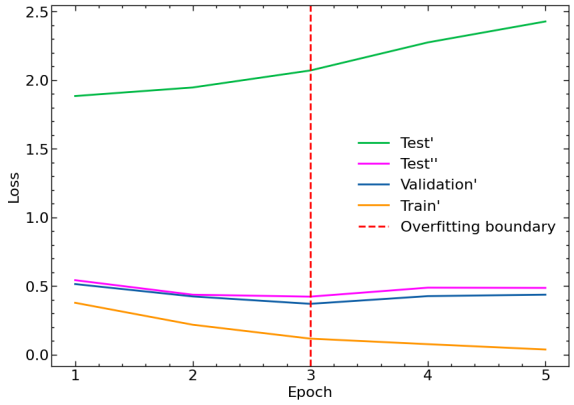

FIGURE 4.9: Validation', train', test' and test" loss of BERT model during retraining.

validation and test set to prevent noisy data. We refer to the resulting sets as *train', validation'* and *test'*. Additionally, we define another set *test"* that is a subset of *test'*, omitting the common incorrect results found in the original test data. Retraining and analyzing the models using these new datasets, the results in table 4.4 are found. Additionally, the training progress of the BERT classifier is shown in figure 4.9. The results reported in this document are based on the model at epoch three, after which overfitting was detected. The performance measures for the train' and validation' sets have improved compared to the original train and validation sets, with a maximum increase of 25%. BERT and SVM both show similar results in the statistics, except that the train' performance does not show the same improvement for BERT as it does for SVM. Using McNemar's test, SVM and BERT are found to improve significantly on NB. Furthermore, a significant difference is found between both test' sets of SVM and BERT, but no difference is found between the test" sets. Looking at the test' and test" sets, it becomes clear that merely cleaning the training set is not satisfactory to achieve a performance increase, however, cleaning the data on the test set, can increase performance by up to 25%.

		Precision (Δ)	**Recall (Δ)**	**F1-score (Δ)**	**Accuracy (Δ)**
NB	**Train'**	0.64 (+0.07)	0.48 (+0.08)	0.50 (+0.09)	0.78 (+0.17)
	Validation'	0.59 (+0.08)	0.47 (+0.10)	0.48 (+0.11)	0.76 (+0.11)
	Test'	0.48 (+0.02)	0.37 (+0.01)	0.37 (equal)	0.56 (equal)
	Test"	0.61 (+0.15)	0.48 (+0.12)	0.50 (+0.13)	0.77 (+0.11)
SVM	**Train'**	**0.93** (+0.17)	**0.89** (+0.21)	**0.90** (+0.21)	0.93 (+0.19)
	Validation'	**0.82** (+0.20)	0.80 (+0.22)	**0.80** (+0.22)	0.89 (+0.24)
	Test'	**0.62** (+0.02)	0.56 (-0.01)	0.57 (equal)	0.65 (+0.01)
	Test"	**0.81** (+0.21)	0.78 (+0.21)	**0.78** (+0.21)	**0.89** (+0.25)
BERT	**Train'**	0.91 (+0.12)	0.88 (+0.15)	0.88 (+0.15)	**0.96** (+0.16)
	Validation'	**0.82** (+0.21)	**0.81** (+0.22)	**0.80** (+0.22)	**0.90** (+0.24)
	Test'	**0.62** (+0.03)	**0.59** (equal)	**0.59** (+0.01)	**0.66** (+0.01)
	Test"	0.79 (+0.20)	**0.79** (+0.20)	**0.78** (+0.20)	**0.89** (+0.24)

TABLE 4.4: Performance metrics of retrained models (the best results are shown in bold, relative change Δ shown between ( ), ' and " denote the modified datasets).

With the base dataset, 26 industries showed satisfactory performance, this is 15% of the 178 industries. Now that new performance measures have been established, 98 industries

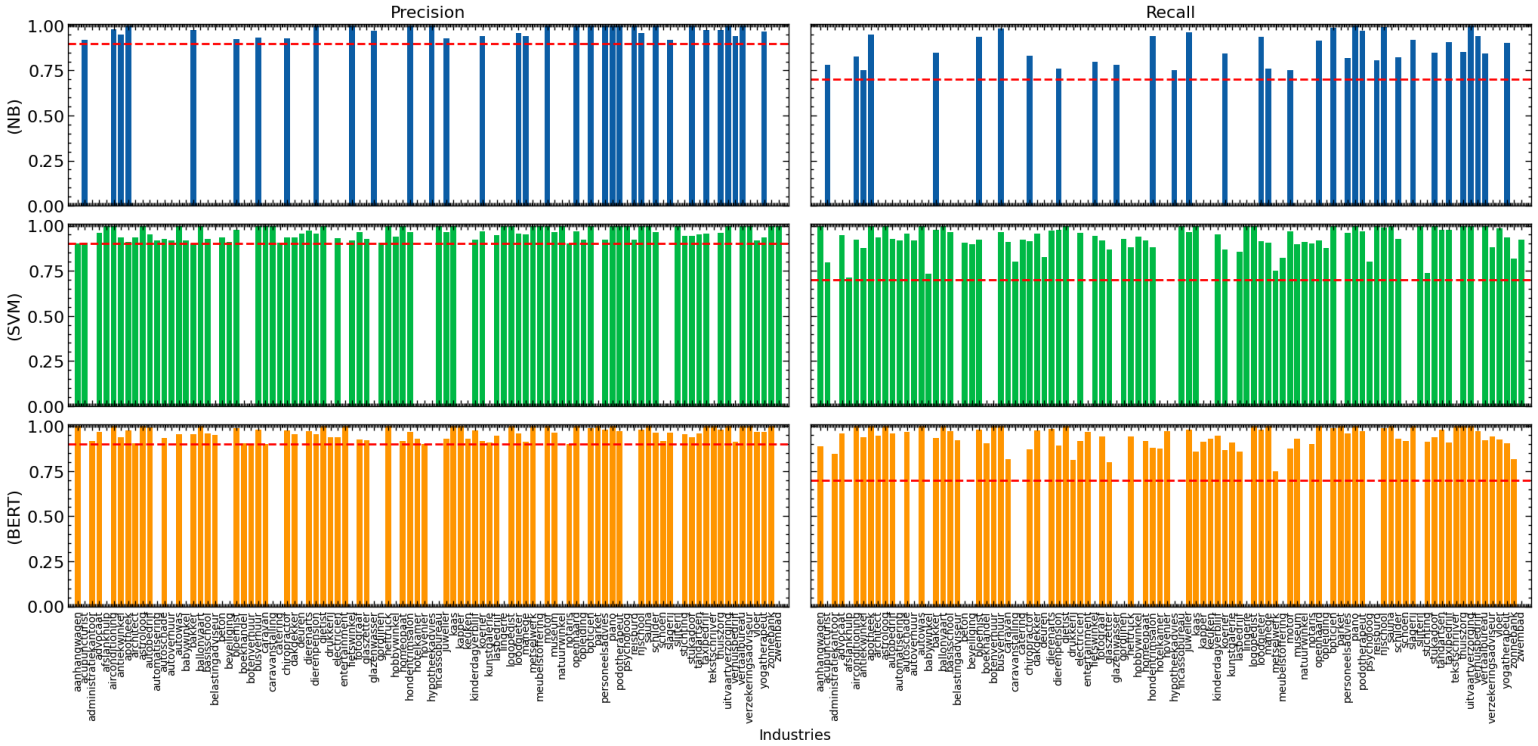

FIGURE 4.10: Industries that satisfy the required performance measures based on test" classifications for at least one classifier, the required performance is shown as a horizontal line. Empty bars indicate a classifier that does not satisfy that industry.

show satisfactory performance measures in at least one model, which is 55% of all industries. These satisfactory industries are shown in figure 4.10. SVM satisfies the most industries, notably 80 or 45%. NB and BERT satisfy 34 and 74 industries respectively.

Looking at industries that do not show the required performance, we find 19 industries that fail on both precision and recall on every classifier. Identifying the cause of these bad performances was done using the different analyses shown earlier. From the different experiments on document size, training set size and manual website scoring (looking at the site of the company and identifying causes of misclassification), we found one factor present in every bad performing industry. This factor was a small training set (<400 records). Which is exactly what was found in figure 4.8.

Chapter 5

Discussion

Before giving final conclusions, we now highlight the limitations of this research. These should be taken into account when using the implemented models in production or when continuing to research on this subject. As was identified in the results, the preprocessing steps have some limitations. Specifically, numerous records contain spam sites which have no correlation with the actual industry. Although removing these records from the dataset shows improved classification results, we hypothesize that multiple spam records still remain. A second limitation of this research is the assumption that all industries are present on Q-info.com, as our dataset did not contain any industries that the models were not trained on. In real-world scenarios, it is highly likely that there exists a company with an industry unsupported by Q-info.com, classifying this will lead to an FP, because there is no correct class for it. Consequently, precision will decrease. Additionally, FPs found in this research might actually be TPs in the real-world, as companies might fit within multiple industries on Q-info.com. As was noted in section 4.3, a restaurant might also do takeaway, making multiple industries correct. This was not taken into account in the performance analysis of this research. Section 3.4 explained a technique to detect mislabeled records in the dataset. Additionally, it was noted that unlike the proposed technique, cross validation was run once instead of 10 (or more) times [38]. This decision may have led to an increased amount of FPs in the mislabeling detection. And, although our technique has improved on precision compared to the proposed method, it should be noted that there is a stark difference between the datasets. A last limitation we note is the focus on text-based classification purely from the home-page of a website. Although, for this research, the scope was set specifically for this, gathering more data from images or subpages could improve performance.

Chapter 6

Conclusion

As a goal of this study, multiple research questions have been defined, as outlined in section 1.2. These questions will now be answered based on the results gathered from the carried out experiments. To do this, the five supporting questions will be answered individually, after which an answer is given to the main research question. To finish this document, some suggestions are made for further research on this subject.

6.1 Answers to Research Questions

RQ 1.1: What performance increase is achievable using an ensemble of classifiers compared to the individual models?

This research covered multiple aspects in terms of performance measures. The statistics: precision, recall, F1-score and accuracy have been documented macro-averaged across the models, as well as individual class results. The individual class results are considered important for the client, because confidence for a subset of industries will enable partial production usage (only for these classes). A minimum of 0.9 precision and 0.7 recall has been decided upon for an industry to be considered production-ready. Considering these statistics, BERT has shown the best individual model results on the base dataset. Outperforming NB in every statistic with a maximum margin of 23%. The difference between BERT and SVM has shown to be minimal, BERT generally showing marginally better results (1-2%). Although outperformed by SVM in terms of macro-precision (a difference of 1%), it managed to satisfy one more industry, notably 19, or 11% of all industries. Using an ensemble of the models NB, SVM and BERT has resulted in improvements across all macro-averaged statistics. Although only by a slight margin of 1-2%, the voting ensemble particularly showed a test performance of 61% on precision, 59% on recall, 58% on F1-Score and 66% on Accuracy. However, the voting ensemble did not manage to satisfy more industries than the individual BERT model, on the contrary, it manged to satisfy 18 industries, which is one less. Considering these results, and the fact that implementing a voting ensemble requires more computational resources, we conclude that implementing an ensemble of the models NB, SVM and BERT does not significantly improve classification performance for a text-based classification problem.

RQ 1.2: What is the correlation between document size and the number of TPs and FPs?

Because each webpage is different, both in structure and content type, the amount of data a scraping attempt collects changes depending on the type of website. This, combined with preprocessing, results in document sizes ranging from 100 bytes up to 40 KB (outliers excluded). Considering this fact, experiments have been carried out to find a possible correlation between document size and classification performance. It was found that, in general, document size does not influence the performance of a classifier. This holds for all sizes, except for small documents (<400 bytes). Using this information, we conclude that preprocessing steps should be slightly more strict in terms of document size, where documents should have a minimum size of 400 bytes. Additionally, there is no correlation found between document sizes and performance, such that a statement can be made about classification confidence based purely on input size.

RQ 1.3: What is the correlation between training set size and classification performance?

Because of the unbalanced nature of the dataset, this research looked into the correlation between training set size and classification performance. This correlation was analyzed for each statistic individually and across classifiers. It was found that, for larger training sets, performance is generally better than for smaller training sets. Specifically, for small training sets (<400 samples), performance is unpredictable, both showing good and bad results. With larger training sets ($\geq$400 samples) a clear improvement is noticed. This indicates that more improvement can be gained when the dataset is expanded, specifically for the industries with fewer records.

RQ 1.4: What is the approximated percentage of mislabeled and spam records in the dataset?

Using 10-fold cross validation on the different models, the common incorrect results were gathered. From the resulting set, a random sampling was taken to analyze how many mislabeled or spam records were present in the dataset. By manually labeling the random sampling, it was found that 20% of the sample was a spam site, and 66% of the sample was found to have an incorrect label. Additionally, we showed that using NB, SVM and BERT leads to a precision of 86% in detecting anomalies, which improves upon the original technique that achieved a precision of 70% [38]. Using the gathered statistics, we are able to conclude that approximately 20% of the base dataset is either mislabeled or can be considered spam.

RQ 1.5: How does performance of the classifiers change when removing mislabeled records, compared to the performance of classification with the original dataset?

After removing the common incorrect results, the performance has been analyzed again. From the results, it can be concluded that SVM and BERT show significant performance advances compared to the base dataset, with results on the new validation and test sets having minimum improvements of 20% in each statistic. It was found that SVM is more sensitive to mislabeled data, hence showing a larger improvement than BERT, resulting in

it being the best performing model in this research. This statement is strengthened when looking at the individual industries the retrained models are able to satisfy. SVM shows the highest percentage, satisfying 45% or 80 industries, which is an increase of 34% compared to the initial model. BERT satisfies 74 industries and NB 34. NB shows a maximum performance increase of 15%, showing a macro-precision of 61%. These statistics lead to the conclusion that removing mislabeled records results in a significant improvement of the classifier performance.

RQ 1: What macro-precision, recall and F1-score performance is achievable with NB, SVM and BERT classifiers, determining the industry of a company using the textual data from its website?

After answering the supporting questions, we now answer the main research question. Three models (NB, SVM and BERT) were implemented and additionally combined into a voting ensemble. Different experiments have been carried out to be able to answer the research question. From the analysis in this research, the following conclusions can be made about the achievable classifier performance: We found that using a voting ensemble with the established models, improves precision by 2% and accuracy by 1%. Nevertheless, it does not manage to satisfy more industries than using a single model. Improving classifier performance can be done using a larger training set for minority classes, as we showed that there is a correlation between performance and training set size. Additionally, the results show that input document size should be at least 400 bytes to be able to accurately classify it. The best performance measures in this research were found classifying a cleaned dataset, since filtering out mislabeled and spam records showed a performance increase of 20-25% and the models being able to satisfy 34% more industries. Based on these conclusions and the results from the experiments, two of the three models that were analyzed were found to show similar performance measures. SVM shows the highest precision score of 81% (with a 2% lead over BERT), and BERT is best on recall with 79% (with a 1% lead over SVM). They additionally tie for best in the other statistics (78% F1-score and 89% accuracy); both show a significant lead compared to NB (10-20% in every statistic). Important facts that were gathered are the number of industries that are satisfied by the classifiers. Based on the requirements set by the client, we managed to satisfy 80 industries using an SVM classifier, 74 industries using BERT and 34 with an NB classifier.

6.2 Further Research

During this research, we found several things that warrant future investigation. We now briefly explain these ideas individually.

Cleaning the dataset.

In section 4.3 it was found that cleaning the dataset can improve classification performance by 25%. However, this is not trivial, as manually cleaning large sets of data is not feasible and will defeat the purpose of classification. Therefore, we suggest research into automatic data cleaning to improve classification performance.

Expanding the dataset.

Section 4.2.2 showed that a correlation can be found between classification performance and training set size. Additionally, we showed that there are several industries with small training sets. It would be interesting to see how expansion of these training sets would influence performance.

Gather more data for small documents.

We found that generally there is no correlation between classification accuracy and input document size. Except for very small documents (<100 bytes) which show worse performance (see section 4.2.1). Therefore, an improvement is expected when these documents are either removed or their data is expanded. Because the input is a webpage, a solution might involve scanning the different subpages of the site for content.

Translating model input.

As was noted in section 1.4.1, Q-info.com necessitates growth for its future success. Currently, three markets are available, NL, BE and UK, from which only the UK market has been experimented on in this study. We identify expansion of Q-info.com not only on the UK market but also on the NL and BE markets. Furthermore, expanding into other markets, such as Germany, could give Q-info.com the necessary boost in growth. To do this with the current system, research has to be done into classifying translated input. By translating a foreign website into English, results might be achieved that satisfy expansion into other countries.

Bibliography

[1] BERT. URL: https://huggingface.co/docs/transformers/model_doc/bert.

[2] DMOZ. URL: https://dmoz-odp.org/.

[3] GitHub - urllib3/urllib3: urllib3 is a user-friendly HTTP client library for Python. URL: https://github.com/urllib3/urllib3/tree/main.

[4] sklearn.svm.LinearSVC. URL: https://scikit-learn/stable/modules/generated/sklearn.svm.LinearSVC.html.

[5] World Wide Knowledge Base (Web->KB) project. URL: https://www.cs.cmu.edu/~webkb/.

[6] LII. An essay towards solving a problem in the doctrine of chances. By the late Rev. Mr. Bayes, F. R. S. communicated by Mr. Price, in a letter to John Canton, A. M. F. R. S. *Philosophical Transactions of the Royal Society of London*, 53:370–418, December 1763. URL: https://royalsocietypublishing.org/doi/10.1098/rstl.1763.0053, doi:10.1098/rstl.1763.0053.

[7] Farah Alshanik, Amy Apon, Alexander Herzog, Ilya Safro, and Justin Sybrandt. Accelerating Text Mining Using Domain-Specific Stop Word Lists, November 2020. arXiv:2012.02294 [cs]. URL: http://arxiv.org/abs/2012.02294.

[8] Elshaimaa Amin, Yasmina M. Elgammal, M. A. Zahran, and Mohamed M. Abdelsalam. Alzheimer's disease: new insight in assessing of amyloid plaques morphologies using multifractal geometry based on Naive Bayes optimized by random forest algorithm. *Scientific Reports*, 13(1):18568, October 2023. Number: 1 Publisher: Nature Publishing Group. URL: https://www.nature.com/articles/s41598-023-45972-w, doi:10.1038/s41598-023-45972-w.

[9] Vimala Balakrishnan and Lloyd-Yemoh Ethel. Stemming and Lemmatization: A Comparison of Retrieval Performances. *Lecture Notes on Software Engineering*, 2(3):262–267, 2014. URL: http://www.lnse.org/show-34-165-1.html, doi:10.7763/LNSE.2014.V2.134.

[10] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit.* " O'Reilly Media, Inc.", 2009.

[11] Vincy Cherian. Heart Disease Prediction Using Naïve Bayes Algorithm and Laplace Smoothing Technique. 5(2), 2017.

[12] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised Cross-lingual Representation Learning at Scale, April 2020. arXiv:1911.02116 [cs]. URL: http://arxiv.org/abs/1911.02116.

[13] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, May 2019. arXiv:1810.04805 [cs]. URL: http://arxiv.org/abs/1810.04805, doi:10.48550/arXiv.1810.04805.

[14] E-Active. Bedrijf - alles over bedrijven. URL: https://www.klik-info.nl/.

[15] E-Active. Bedrijf - alles over bedrijven. URL: https://www.klik-info.be/.

[16] E-Active. Gespecialiseerd in webtechniek - e-Active. URL: https://www.e-active.nl/.

[17] E-Active. q-info.com. URL: https://www.q-info.com/.

[18] E-Active. Search for a company - everything about companies. URL: https://www.company-info.co.uk/.

[19] Allen L. Edwards. Note on the "correction for continuity" in testing the significance of the difference between correlated proportions. *Psychometrika*, 13(3):185–187, September 1948. doi:10.1007/BF02289261.

[20] Zakaria Elberrichi and Abdellatif Rahmoun. Experimenting N-Grams in Text Categorization. *International Arab Journal of Information Technology*, 4:377–387, October 2007.

[21] Tristan Fletcher. Support Vector Machines Explained.

[22] Eric J Glover and Kostas Tsioutsiouliklis. Using Web Structure for Classifying and Describing Web Pages.

[23] Sajad Fathi Hafshejani and Zahra Moberfard. A new trigonometric kernel function for support vector machine. *Iran Journal of Computer Science*, 6(2):137–145, June 2023. arXiv:2210.08585 [cs]. URL: http://arxiv.org/abs/2210.08585, doi:10.1007/s42044-022-00130-9.

[24] Mahdi Hashemi. Web page classification: a survey of perspectives, gaps, and future directions. *Multimedia Tools and Applications*, 79(17):11921–11945, May 2020. doi:10.1007/s11042-019-08373-8.

[25] Huan Liu and R. Setiono. Chi2: feature selection and discretization of numeric attributes. In *Proceedings of 7th IEEE International Conference on Tools with Artificial Intelligence*, pages 388–391, Herndon, VA, USA, 1995. IEEE Comput. Soc. Press. URL: http://ieeexplore.ieee.org/document/479783/, doi:10.1109/TAI.1995.479783.

[26] Sajjad Jalil, Muhammad Usman, and Alvis Fong. Highly accurate phishing URL detection based on machine learning. *Journal of Ambient Intelligence and Humanized Computing*, 14(7):9233–9251, July 2023. doi:10.1007/s12652-022-04426-3.

[27] Thorsten Joachims. Text categorization with Support Vector Machines: Learning with many relevant features. In Jaime G. Carbonell, Jörg Siekmann, G. Goos, J. Hartmanis, J. Van Leeuwen, Claire Nédellec, and Céline Rouveirol, editors, *Machine Learning: ECML-98*, volume 1398, pages 137–142. Springer Berlin Heidelberg, Berlin, Heidelberg, 1998. Series Title: Lecture Notes in Computer Science. URL: http://link.springer.com/10.1007/BFb0026683, doi:10.1007/BFb0026683.

[28] Jashanjot Kaur and Preetpal Buttar. A Systematic Review on Stopword Removal Algorithms. 4:207–210, April 2018.

[29] Aldin Kovačević, Zerina Mašetić, and Dino Kečo. Naive Website Categorization Based on Text Coverage. In Samir Avdaković, Ismar Volić, Aljo Mujčić, Tarik Uzunović, and Adnan Mujezinović, editors, *Advanced Technologies, Systems, and Applications V: Papers Selected by the Technical Sciences Division of the Bosnian-Herzegovinian American Academy of Arts and Sciences 2020*, Lecture Notes in Networks and Systems, pages 435–448. Springer International Publishing, Cham, 2021. doi:10.1007/978-3-030-54765-3_30.

[30] Li-Ping Jing, Hou-Kuan Huang, and Hong-Bo Shi. Improved feature selection approach TFIDF in text mining. In *Proceedings. International Conference on Machine Learning and Cybernetics*, volume 2, pages 944–946, Beijing, China, 2002. IEEE. URL: http://ieeexplore.ieee.org/document/1174522/, doi:10.1109/ICMLC.2002.1174522.

[31] Daniel López-Sánchez, Juan Corchado Rodríguez, and Angélica González. A CBR System for Image-Based Webpage Classification: Case Representation with Convolutional Neural Networks. May 2017.

[32] Quinn McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, June 1947. doi:10.1007/BF02295996.

[33] Sara Meshkizadeh and Dr. Amir Masoud Rahmani. Webpage Classification based on Compound of Using HTML Features & URL Features and Features of Sibling Pages. *International Journal of Advancements in Computing Technology*, 2(4):36–46, October 2010. URL: http://www.aicit.org/ijact/paper_detail.html?q=85, doi:10.4156/ijact.vol2.issue4.4.

[34] Ammar Mohammed and Rania Kora. An effective ensemble deep learning framework for text classification. *Journal of King Saud University - Computer and Information Sciences*, 34(10, Part A):8825–8837, November 2022. URL: https://www.sciencedirect.com/science/article/pii/S1319157821003013, doi:10.1016/j.jksuci.2021.11.001.

[35] Amit Kumar Nandanwar and Jaytrilok Choudhary. Contextual Embeddings-Based Web Page Categorization Using the Fine-Tune BERT Model. *Symmetry*, 15(2):395, February 2023. Number: 2 Publisher: Multidisciplinary Digital Publishing Institute. URL: https://www.mdpi.com/2073-8994/15/2/395, doi:10.3390/sym15020395.

[36] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai,

and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[37] Fabian Pedregosa, Gael Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, and David Cournapeau. Scikit-learn: Machine Learning in Python. *MACHINE LEARNING IN PYTHON*.

[38] Mannes Poel. Detecting Mislabeled Data Using Supervised Machine Learning Techniques. In Dylan D. Schmorrow and Cali M. Fidopiastis, editors, *Augmented Cognition. Neurocognition and Machine Learning*, volume 10284, pages 571–581. Springer International Publishing, Cham, 2017. Series Title: Lecture Notes in Computer Science. URL: https://link.springer.com/10.1007/978-3-319-58628-1_43, doi:10.1007/978-3-319-58628-1_43.

[39] Vikas C. Raykar and Amrita Saha. Data Split Strategiesfor Evolving Predictive Models. In Annalisa Appice, Pedro Pereira Rodrigues, Vítor Santos Costa, Carlos Soares, João Gama, and Alípio Jorge, editors, *Machine Learning and Knowledge Discovery in Databases*, Lecture Notes in Computer Science, pages 3–19, Cham, 2015. Springer International Publishing. doi:10.1007/978-3-319-23528-8_1.

[40] Leonard Richardson. Beautiful soup documentation. *April*, 2007.

[41] A. Selamat, N.Q. Do, and O. Krejcar. Detecting phishing URLs with word embedding and deep learning. In *Perspectives and Considerations on the Evolution of Smart Systems*, pages 296–319. 2023. doi:10.4018/978-1-6684-7684-0.ch011.

[42] Ali Selamat and Sigeru Omatu. Web page feature selection and classification using neural networks. *Information Sciences*, 158:69–88, January 2004. URL: https://www.sciencedirect.com/science/article/pii/S0020025503001944, doi:10.1016/j.ins.2003.03.003.

[43] Syeda Ayesha Siddiqha and M Islabudeen. Web-Page Content Classification on Entropy Classifiers using Machine Learning. In *2023 International Conference for Advancement in Technology (ICONAT)*, pages 1–5, Goa, India, January 2023. IEEE. URL: https://ieeexplore.ieee.org/document/10080462/, doi:10.1109/ICONAT57137.2023.10080462.

[44] Jasmeet Singh and Vishal Gupta. Text Stemming: Approaches, Applications, and Challenges. *ACM Computing Surveys*, 49(3):1–46, September 2017. URL: https://dl.acm.org/doi/10.1145/2975608, doi:10.1145/2975608.

[45] R.S. Wilkho, S. Chang, and N.G. Gharaibeh. FF-BERT: A BERT-based ensemble for automated classification of web-based text on flash flood events. *Advanced Engineering Informatics*, 59, 2024. doi:10.1016/j.aei.2023.102293.

[46] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M.

Rush. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: System demonstrations*, pages 38–45, Online, October 2020. Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/2020.emnlp-demos.6.

[47] Patrick Woogue, Gabriel Pineda, and Christian Maderazo. Automatic Web Page Categorization Using Machine Learning and Educational-Based Corpus. *International Journal of Computer Theory and Engineering*, 9:427–432, January 2017. doi:10.7763/IJCTE.2017.V9.1180.

[48] Fei Wu, Xiao-Yuan Jing, Pengfei Wei, Chao Lan, Yimu Ji, Guo-Ping Jiang, and Qinghua Huang. Semi-supervised multi-view graph convolutional networks with application to webpage classification. *Information Sciences*, 591:142–154, April 2022. URL: https://www.sciencedirect.com/science/article/pii/S0020025522000160, doi:10.1016/j.ins.2022.01.013.

[49] Yiming Yang, Seán Slattery, and Rayid Ghani. A Study of Approaches to Hypertext Categorization. *Journal of Intelligent Information Systems*, 18(2):219–241, March 2002. doi:10.1023/A:1013685612819.

[50] Jian Zhang, Rong Jin, Yiming Yang, and Alex G Hauptmann. Modified Logistic Regression: An Approximation to SVM and Its Applications in Large-Scale Text Categorization.

[51] Yongli Zhang. Support Vector Machine Classification Algorithm and Its Application. In Chunfeng Liu, Leizhen Wang, and Aimin Yang, editors, *Information Computing and Applications*, Communications in Computer and Information Science, pages 179–186, Berlin, Heidelberg, 2012. Springer. doi:10.1007/978-3-642-34041-3_27.

Appendix A

Dataset Hierarchy

A.1 Dataset Distribution

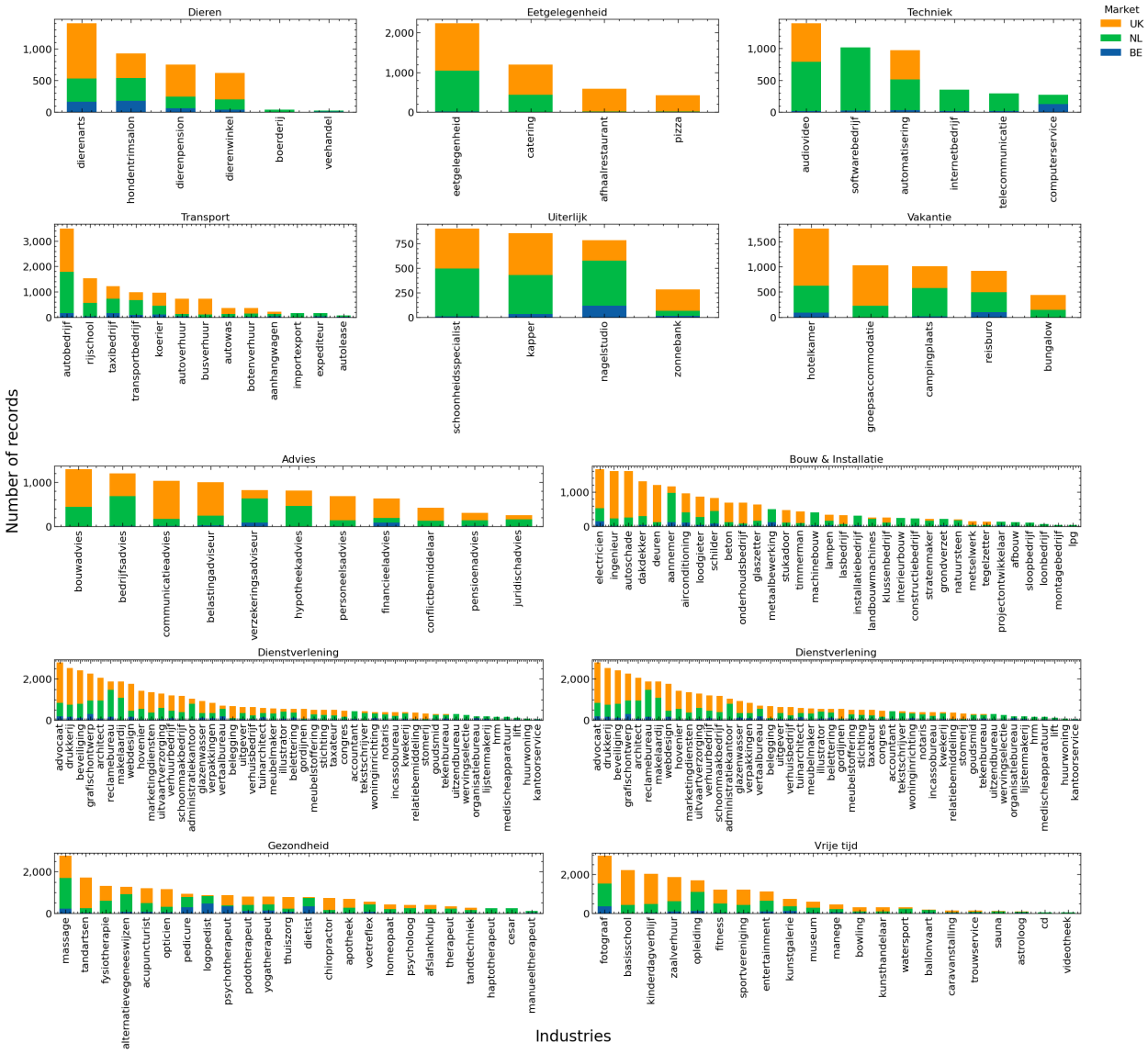

FIGURE A.1: Distribution of industries per top-level category, and across markets.

A.2 Dataset Distribution United Kingdom

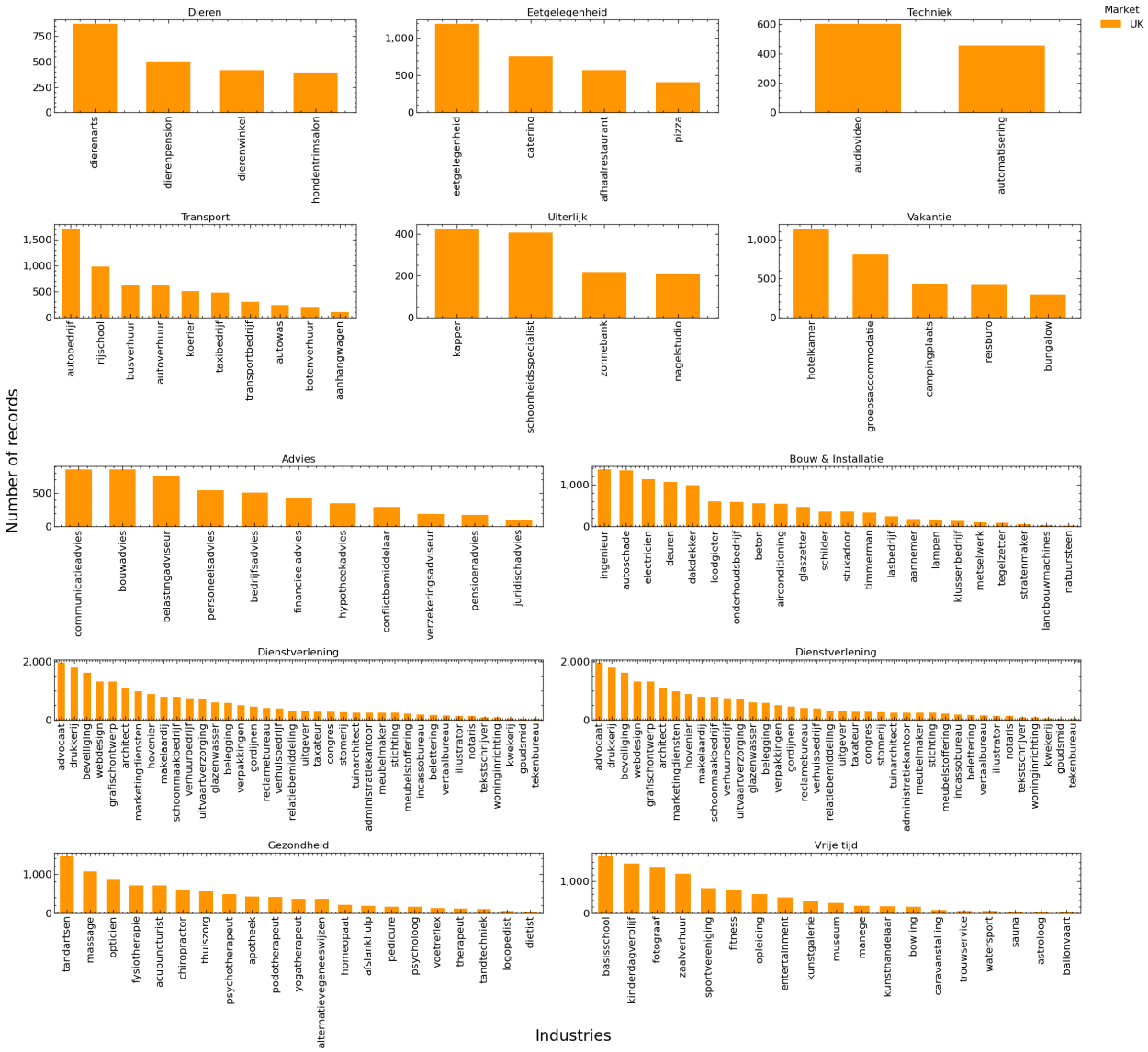

FIGURE A.2: Distribution of industries in the UK per top-level category.

A.3 Categories and Industries

Top-level category	Bottom-level industries
Advies	bedrijfsadvies, belastingadviseur, bouwadvies, communicatieadvies, conflictbemiddelaar, financieeladvies, hypotheekadvies, juridischadvies, pensioenadvies, personeelsadvies, verzekeringsadviseur
Bouw & Installatie	afhaalrestaurant, catering, eetgelegenheid, pizza
Dienstverlening	kapper, nagelstudio, schoonheidsspecialist, zonnebank
Dieren	aannemer, afbouw, airconditioning, autoschade, beton, constructiebedrijf, dakdekker, deuren, electricien, glaszetter, grondverzet, ingenieur, installatiebedrijf, interieurbouw, klussenbedrijf, lampen, landbouwmachines, lasbedrijf, loodgieter, loonbedrijf, lpg, machinebouw, metaalbewerking, metselwerk, montagebedrijf, natuursteen, onderhoudsbedrijf, projectontwikkelaar, schilder, sloopbedrijf, stratenmaker, stukadoor, tegelzetter, timmerman
Eetgelegenheid	acupuncturist, afslankhulp, alternatievegeneeswijzen, apotheek, cesar, chiropractor, dietist, fysiotherapie, haptotherapeut, homeopaat, logopedist, manueeltherapeut, massage, opticien, pedicure, podotherapeut, psycholoog, psychotherapeut, tandartsen, tandtechniek, therapeut, thuiszorg, voetreflex, yogatherapeut
Gezondheid	bungalow, campingplaats, groepsaccommodatie, hotelkamer, reisburo
Techniek	accountant, administratiekantoor, advocaat, architect, belegging, belettering, beveiliging, congres, drukkerij, glazenwasser, gordijnen, goudsmid, grafischontwerp, hovenier, hrm, huurwoning, illustrator, incassobureau, kantoorservice, kwekerij, lift, lijstenmakerij, makelaardij, marketingdiensten, medischeapparatuur, meubelmaker, meubelstoffering, notaris, organisatiebureau, reclamebureau, relatiebemiddeling, schoonmaakbedrijf, stichting, stomerij, taxateur, tekenbureau, tekstschrijver, tuinarchitect, uitgever, uitvaartverzorging, uitzendbureau, verhuisbedrijf, verhuurbedrijf, verpakkingen, vertaalbureau, webdesign, wervingselectie, woninginrichting
Transport	audiovideo, automatisering, computerservice, internetbedrijf, softwarebedrijf, telecommunicatie
Uiterlijk	astroloog, ballonvaart, basisschool, bowling, caravanstalling, cd, entertainment, fitness, fotograaf, kinderdagverblijf, kunstgalerie, kunsthandelaar, manege, museum, opleiding, sauna, sportvereniging, trouwservice, vereniging, videotheek, watersport, zaalverhuur
Vakantie	boerderij, dierenarts, dierenpension, dierenwinkel, hondentrimsalon, veehandel
Vrije tijd	aanhangwagen, autobedrijf, autolease, autoverhuur, autowas, botenverhuur, busverhuur, expediteur, importexport, koerier, rijschool, taxibedrijf, transportbedrijf
Winkel	antiekwinkel, babywinkel, bakker, bedden, bloemist, boekhandel, boten, cadeau, caravan, computerwinkel, doehetzelf, drogist, elektronica, fietswinkel, heftruck, hobbywinkel, juwelier, kaas, keuken, kleding, kozijnen, kunststoffen, lingerie, meubelwinkel, motor, natuurwinkel, openhaard, parket, piano, relatiegeschenken, sanitair, schoen, slagerij, slijterij, speelgoed, sportwinkel, tabak, tegels, textiel, tuinwinkel, tweedehands, vishandel, vloer, webshop, zonwering, zwembad

Table A.1: Industries belonging to top-level category.

Appendix B

Preprocessing Decisions

B.1 Duplicate Records Mapping

Industry a	**Industry** b	**Decision**
hovenier	**tuinarchitect**	hovenier
autoschade	**onderhoudsbedrijf**	autoschade
makelaardij	**taxateur**	makelaardij
sportvereniging	**vereniging**	sportvereniging
accountant	**belastingadviseur**	accountant
grafischontwerp	**webdesign**	grafischontwerp
hypotheekadvies	**verzekeringsadviseur**	hypotheekadvies
autobedrijf	**onderhoudsbedrijf**	autobedrijf
autoverhuur	**verhuurbedrijf**	autoverhuur
grafischontwerp	**reclamebureau**	reclamebureau
massage	**voetreflex**	voetreflex
drukkerij	**grafischontwerp**	drukkerij

TABLE B.1: Duplicate records with industry a and b resulting in a decision.

B.2 For Sale Detection

Matching string
deze domeinnaam is geactiveerd
to change this page, upload your website into the public_html
domein gereserveerd
domraider ico
onjuiste verwijzing domein
website is for sale
domeinnaam kan over een paar minuten van jou zijn
index of /
is beschikbaar voor de verkoop, vraag vandaag een vrijblijvende prijsopgave aan
is te koop voor maar
is te koop. mocht u interesse hebben in overname ervan
looks like this domain isn't connected to a website yet
is gereserveerd door een klant van transip
mooiedomeinnaam.nl is onderdeel van media village
you see this page because there is no web site at this address
helaas was iemand je al voor. check hieronder een andere domeinnaam
buy this domain
no website configured
domeinnaam geregistreerd
web server's default page
reserved domain
website is buiten gebruik
als je een domeinnaam aanschaft via dan.com
commercive beheert en onderhoudt een portfolio
suspended domain
domeinnaam is geparkeerd
your domain is active and is using
vip slots online casino review
de domeinnaam die u zoekt is geblokkeerd
is for sale
domeinnaam is geregistreerd voor
registered at namecheap.com
parking-crew.com
js/parking.2.103.

Table B.2: When a website contains a matching string, it is labeled as "for sale".

Appendix C

Detailed Results

C.1 NB Individual Experiments

Model configuration	Precision	Recall	F1
Stopword removal: Without, Lexical normalization: None, NGrams: Unigrams, TF-IDF min/max: 0/100, KBest: All	0.31	0.16	0.21
Stopword removal: With, Lexical normalization: None, NGrams: Unigrams, TF-IDF min/max: 0/100, KBest: All	0.37	0.21	0.27
Stopword removal: With, Lexical normalization: None, NGrams: Unigrams, TF-IDF min/max: 0/100, KBest: 2000	0.46	0.34	0.39
Stopword removal: With, Lexical normalization: None, NGrams: Unigrams, TF-IDF min/max: 0/100, KBest: 1500	0.48	0.34	0.40
Stopword removal: With, Lexical normalization: None, NGrams: Unigrams, TF-IDF min/max: 0/100, KBest: 1000	0.47	0.33	0.39
Stopword removal: With, Lexical normalization: None, NGrams: Unigrams, TF-IDF min/max: 0/100, KBest: 500	0.44	0.30	0.36
Stopword removal: With, Lexical normalization: None, NGrams: Unigrams, TF-IDF min/max: 0/100, KBest: 100	0.24	0.13	0.17
Stopword removal: With, Lexical normalization: Stemming, NGrams: Unigrams, TF-IDF min/max: 0/100, KBest: 1500	0.47	0.35	0.40
Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Unigrams, TF-IDF min/max: 0/100, KBest: 1500	0.46	0.35	0.40
Stopword removal: With, Lexical normalization: Stemming, NGrams: Unigrams, TF-IDF min/max: 5/100, KBest: 1500	0.50	0.36	0.42
Stopword removal: With, Lexical normalization: Stemming, NGrams: Unigrams, TF-IDF min/max: 10/100, KBest: 1500	0.50	0.37	**0.43**
Stopword removal: With, Lexical normalization: Stemming, NGrams: Unigrams, TF-IDF min/max: 20/100, KBest: 1500	**0.51**	0.37	**0.43**
Stopword removal: With, Lexical normalization: Stemming, NGrams: Unigrams, TF-IDF min/max: 30/100, KBest: 1500	**0.51**	0.37	**0.43**
Stopword removal: With, Lexical normalization: Stemming, NGrams: Unigrams, TF-IDF min/max: 20/90, KBest: 1500	**0.51**	0.37	**0.43**
Stopword removal: With, Lexical normalization: Stemming, NGrams: Unigrams, TF-IDF min/max: 20/80, KBest: 1500	**0.51**	0.37	**0.43**
Stopword removal: With, Lexical normalization: Stemming, NGrams: Unigrams, TF-IDF min/max: 20/10, KBest: 1500	0.50	**0.38**	**0.43**
Stopword removal: With, Lexical normalization: Stemming, NGrams: Both, TF-IDF min/max: 20/100, KBest: 1500	0.48	0.34	0.40
Stopword removal: With, Lexical normalization: Stemming, NGrams: Bigrams, TF-IDF min/max: 20/100, KBest: 1500	0.41	0.20	0.27

TABLE C.1: NB model configurations with corresponding validation macro-averaged precision and recall performance, with resulting F1-score. The best results and the chosen best configuration are shown in bold.

C.2 NB 10-Fold Cross Validation Results

Fold	Precision	Recall	F1-Score	Accuracy
1	0.500804	0.392078	0.398127	0.595806
2	0.502751	0.394621	0.400284	0.587102
3	0.492464	0.392719	0.400049	0.595615
4	0.508032	0.395133	0.401746	0.595973
5	0.497555	0.394434	0.401166	0.588047
6	0.482196	0.383942	0.391326	0.591101
7	0.497085	0.395128	0.402950	0.590700
8	0.511973	0.399176	0.408109	0.604993
9	0.522688	0.396846	0.406459	0.596104
10	0.503302	0.393139	0.400873	0.596854
Mean	0.50	0.39	0.40	0.59
Std	0.01	0.004	0.004	0.005

TABLE C.2: NB cross validation results per fold, mean and standard deviation (std) shown.

C.3 NB Per-class Performance

Label	Support		TP		FP		Precision		Recall		F1-Score	
	Val	Test	Val	Test	Val	Test	Val	Test	Val	Test	Val	Test
uitvaartverzorging	71	70	70	69	1	4	0.99	0.95	0.99	0.99	0.99	0.97
opticien	86	85	86	80	1	2	0.99	0.98	1.0	0.94	0.99	0.96
openhaard	38	37	32	33	2	0	0.94	1.0	0.84	0.89	0.89	0.94
apotheek	44	42	37	38	2	1	0.95	0.97	0.84	0.9	0.89	0.94
piano	15	14	13	12	0	0	1.0	1.0	0.87	0.86	0.93	0.92
tandartsen	148	146	147	141	29	22	0.84	0.87	0.99	0.97	0.91	0.91
rijschool	99	98	93	91	21	11	0.82	0.89	0.94	0.93	0.87	0.91
busverhuur	63	61	55	54	5	6	0.92	0.9	0.87	0.89	0.89	0.89
bloemist	106	105	97	87	18	7	0.84	0.93	0.92	0.83	0.88	0.87
stukadoor	37	36	25	28	2	1	0.93	0.97	0.68	0.78	0.78	0.86
juwelier	62	60	50	52	16	11	0.76	0.83	0.81	0.87	0.78	0.85
taxibedrijf	49	47	32	38	11	6	0.74	0.86	0.65	0.81	0.7	0.84
hondentrimsalon	40	39	32	31	9	4	0.78	0.89	0.8	0.79	0.79	0.84
podotherapeut	42	41	37	32	3	5	0.92	0.86	0.88	0.78	0.9	0.82
airconditioning	55	54	47	41	8	5	0.85	0.89	0.85	0.76	0.85	0.82
slagerij	30	30	19	23	0	3	1.0	0.88	0.63	0.77	0.78	0.82
yogatherapeut	37	37	28	28	2	4	0.93	0.88	0.76	0.76	0.84	0.81
bakker	54	53	17	38	4	4	0.81	0.9	0.31	0.72	0.45	0.8
verhuisbedrijf	40	38	35	31	19	9	0.65	0.78	0.88	0.82	0.74	0.79
dakdekker	100	99	76	83	29	28	0.72	0.75	0.76	0.84	0.74	0.79
vloer	74	74	66	66	39	28	0.63	0.7	0.89	0.89	0.74	0.79
gordijnen	46	46	36	40	18	15	0.67	0.73	0.78	0.87	0.72	0.79
autobedrijf	172	171	142	154	84	72	0.63	0.68	0.83	0.9	0.71	0.78
thuiszorg	57	55	45	43	3	14	0.94	0.75	0.79	0.78	0.86	0.77
reisburo	43	42	23	27	2	1	0.92	0.96	0.53	0.64	0.68	0.77

TABLE C.3: NB per-class performance metrics of best 25 industries based on test F1-score (satisfactory industries shown in bold).

C.4 NB Confusion Matrix

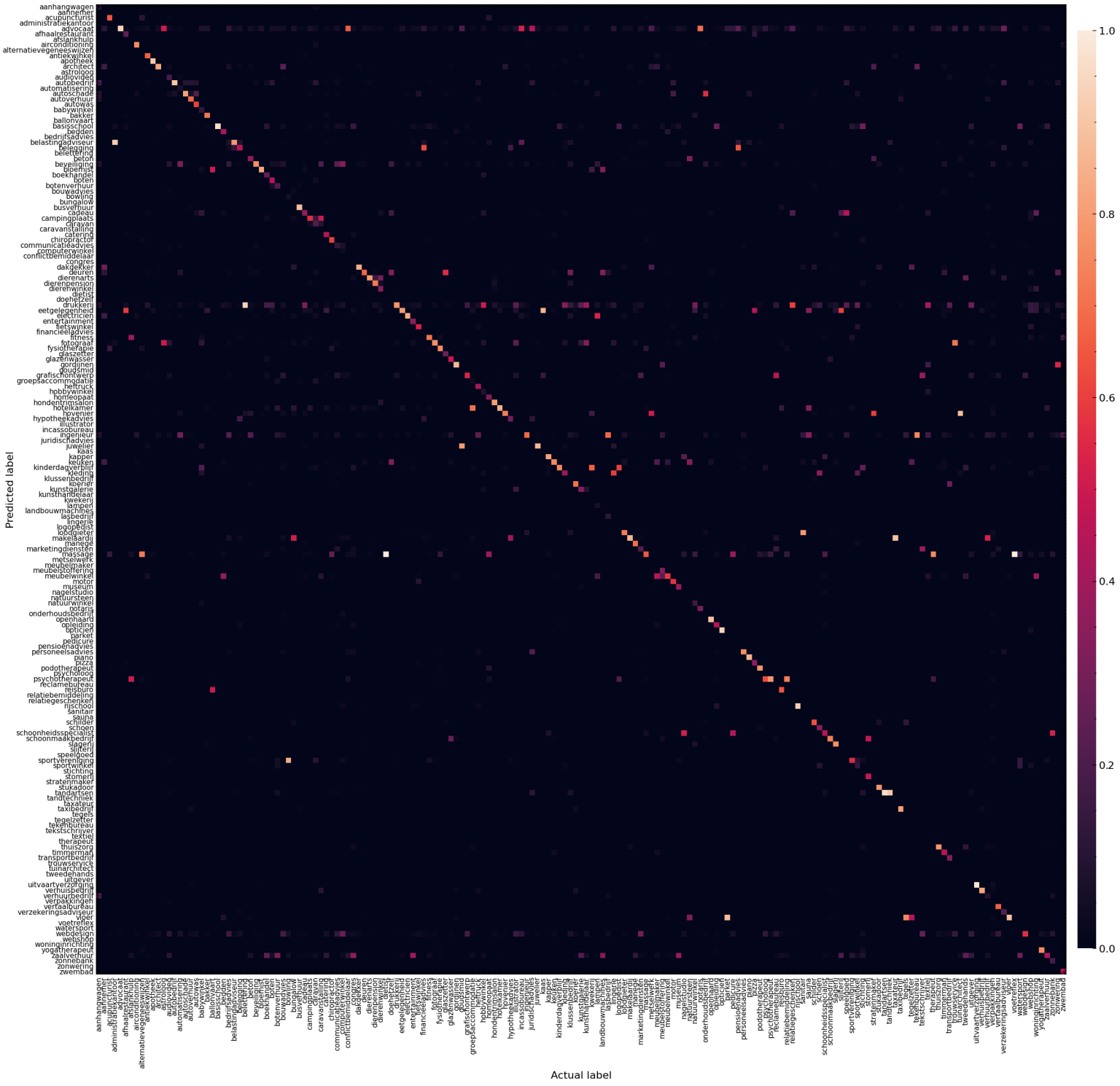

FIGURE C.1: Confusion matrix of Naive Bayes classifier on test data.

C.5 SVM Individual Experiments

Model configuration	Precision	Recall	F1
Stopword removal: Without, Lexical normalization: None, NGrams: Unigrams, TF-IDF min/max: 0/100, KBest: All, C: 1, Loss: squared_hinge, Penalty: l2, Dual: True, Tol: 1e-4, Class weight: None, Degree: Na, Kernel: Linear	0.60	0.57	0.58
Stopword removal: With, Lexical normalization: None, NGrams: Unigrams, TF-IDF min/max: 0/100, KBest: All, C: 1, Loss: squared_hinge, Penalty: l2, Dual: True, Tol: 1e-4, Class weight: None, Degree: Na, Kernel: Linear	0.60	0.58	0.59
Stopword removal: With, Lexical normalization: Stemming, NGrams: Unigrams, TF-IDF min/max: 0/100, KBest: All, C: 1, Loss: squared_hinge, Penalty: l2, Dual: True, Tol: 1e-4, Class weight: None, Degree: Na, Kernel: Linear	0.60	0.57	0.58
Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Unigrams, TF-IDF min/max: 0/100, KBest: All, C: 1, Loss: squared_hinge, Penalty: l2, Dual: True, Tol: 1e-4, Class weight: None, Degree: Na, Kernel: Linear	0.61	0.57	0.59
Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Unigrams, TF-IDF min/max: 0/100, KBest: 2500, C: 1, Loss: squared_hinge, Penalty: l2, Dual: True, Tol: 1e-4, Class weight: None, Degree: Na, Kernel: Linear	0.61	0.57	0.59
Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Unigrams, TF-IDF min/max: 0/100, KBest: 2000, C: 1, Loss: squared_hinge, Penalty: l2, Dual: True, Tol: 1e-4, Class weight: None, Degree: Na, Kernel: Linear	0.60	0.57	0.58
Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Unigrams, TF-IDF min/max: 0/100, KBest: 1500, C: 1, Loss: squared_hinge, Penalty: l2, Dual: True, Tol: 1e-4, Class weight: None, Degree: Na, Kernel: Linear	0.61	0.57	0.59
Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Unigrams, TF-IDF min/max: 0/100, KBest: 1000, C: 1, Loss: squared_hinge, Penalty: l2, Dual: True, Tol: 1e-4, Class weight: None, Degree: Na, Kernel: Linear	0.60	0.56	0.58
Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Unigrams, TF-IDF min/max: 0/100, KBest: 500, C: 1, Loss: squared_hinge, Penalty: l2, Dual: True, Tol: 1e-4, Class weight: None, Degree: Na, Kernel: Linear	0.55	0.53	0.54
Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Unigrams, TF-IDF min/max: 0/100, KBest: 100, C: 1, Loss: squared_hinge, Penalty: l2, Dual: True, Tol: 1e-4, Class weight: None, Degree: Na, Kernel: Linear	0.27	0.34	0.30
Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Unigrams, TF-IDF min/max: 5/100, KBest: 2500, C: 1, Loss: squared_hinge, Penalty: l2, Dual: True, Tol: 1e-4, Class weight: None, Degree: Na, Kernel: Linear	0.62	0.58	**0.60**
Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Unigrams, TF-IDF min/max: 10/100, KBest: 2500, C: 1, Loss: squared_hinge, Penalty: l2, Dual: True, Tol: 1e-4, Class weight: None, Degree: Na, Kernel: Linear	0.62	0.58	**0.60**
Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Unigrams, TF-IDF min/max: 20/100, KBest: 2500, C: 1, Loss: squared_hinge, Penalty: l2, Dual: True, Tol: 1e-4, Class weight: None, Degree: Na, Kernel: Linear	0.62	0.58	**0.60**
Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Unigrams, TF-IDF min/max: 5/90, KBest: 2500, C: 1, Loss: squared_hinge, Penalty: l2, Dual: True, Tol: 1e-4, Class weight: None, Degree: Na, Kernel: Linear	0.40	0.26	0.32
Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Bigrams, TF-IDF min/max: 5/100, KBest: 2500, C: 1, Loss: squared_hinge, Penalty: l2, Dual: True, Tol: 1e-4, Class weight: None, Degree: Na, Kernel: Linear	0.54	0.45	0.49

Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Both, TF-IDF min/max: 5/100, KBest: 2500, C: 1, Loss: squared_hinge, Penalty: l2, Dual: True, Tol: 1e-4, Class weight: None, Degree: Na, Kernel: Linear	0.61	0.57	0.59
Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Unigrams, TF-IDF min/max: 5/100, KBest: 2500, C: 0.1, Loss: squared_hinge, Penalty: l2, Dual: True, Tol: 1e-4, Class weight: None, Degree: Na, Kernel: Linear	0.59	0.55	0.57
Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Unigrams, TF-IDF min/max: 5/100, KBest: 2500, C: 10, Loss: squared_hinge, Penalty: l2, Dual: True, Tol: 1e-4, Class weight: None, Degree: Na, Kernel: Linear	0.60	0.56	0.58
Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Unigrams, TF-IDF min/max: 5/100, KBest: 2500, C: 1, Loss: hinge, Penalty: l2, Dual: True, Tol: 1e-4, Class weight: None, Degree: Na, Kernel: Linear	0.57	0.56	0.56
Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Unigrams, TF-IDF min/max: 5/100, KBest: 2500, C: 1, Loss: squared_hinge, Penalty: l2, Dual: False, Tol: 1e-4, Class weight: None, Degree: Na, Kernel: Linear	0.62	0.58	**0.60**
Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Unigrams, TF-IDF min/max: 5/100, KBest: 2500, C: 1, Loss: squared_hinge, Penalty: l2, Dual: True, Tol: 1e-3, Class weight: None, Degree: Na, Kernel: Linear	0.62	0.58	**0.60**
Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Unigrams, TF-IDF min/max: 5/100, KBest: 2500, C: 1, Loss: squared_hinge, Penalty: l2, Dual: True, Tol: 1e-2, Class weight: None, Degree: Na, Kernel: Linear	0.62	0.58	**0.60**
Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Unigrams, TF-IDF min/max: 5/100, KBest: 2500, C: 1, Loss: hinge, Penalty: l2, Dual: True, Tol: 1e-4, Class weight: Balanced, Degree: Na, Kernel: Linear	0.55	**0.60**	0.57
Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Unigrams, TF-IDF min/max: 5/100, KBest: 2500, C: 1, Loss: Na, Penalty: Na, Dual: Na, Tol: 1e-4, Class weight: None, Degree: 3, Kernel: Polynomial	**0.64**	0.45	0.53
Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Unigrams, TF-IDF min/max: 5/100, KBest: 2500, C: 1, Loss: Na, Penalty: Na, Dual: Na, Tol: 1e-4, Class weight: None, Degree: 4, Kernel: Polynomial	**0.64**	0.40	0.49
Stopword removal: With, Lexical normalization: Lemmatization, NGrams: Unigrams, TF-IDF min/max: 5/100, KBest: 2500, C: 1, Loss: Na, Penalty: Na, Dual: Na, Tol: 1e-4, Class weight: None, Degree: Na, Kernel: RBF	0.63	0.55	0.59

TABLE C.4: SVM model configurations with corresponding validation macro-averaged precision and recall performance, with resulting F1-score. The best results and the chosen best configuration are shown in bold. Unsupported and computationally infeasible configurations left out (>= 10min training time), non-applicable parameters marked with "Na".

C.6 SVM 10-Fold Cross Validation Results

Fold	Precision	Recall	F1-Score	Accuracy
1	0.653433	0.621006	0.622542	0.685235
2	0.640966	0.613759	0.612514	0.687492
3	0.657859	0.625060	0.622720	0.685585
4	0.641330	0.596768	0.598403	0.677827
5	0.650343	0.608123	0.609335	0.688259
6	0.659714	0.623303	0.622740	0.699262
7	0.667993	0.631455	0.633892	0.693806
8	0.625128	0.597736	0.595391	0.685249
9	0.646738	0.611985	0.610634	0.688757
10	0.670148	0.611976	0.620316	0.687920
Mean	0.65	0.61	0.61	0.69
Std	0.013	0.011	0.011	0.005

TABLE C.5: SVM cross validation results per fold, mean and standard deviation (std) shown.

C.7 SVM Per-class Performance

Label	Support		TP		FP		Precision		Recall		F1-Score	
	Val	Test	Val	Test	Val	Test	Val	Test	Val	Test	Val	Test
ballonvaart	3	2	3	2	0	0	1.0	1.0	1.0	1.0	1.0	1.0
logopedist	6	5	6	5	3	0	0.67	1.0	1.0	1.0	0.8	1.0
opticien	86	85	86	80	0	1	1.0	0.99	1.0	0.94	1.0	0.96
autowas	25	23	24	22	0	1	1.0	0.96	0.96	0.96	0.98	0.96
uitvaartverzorging	71	70	70	69	0	6	1.0	0.92	0.99	0.99	0.99	0.95
openhaard	38	37	33	34	5	1	0.87	0.97	0.87	0.92	0.87	0.94
tandartsen	148	146	144	137	13	7	0.92	0.95	0.97	0.94	0.94	0.94
stomerij	27	27	21	26	1	3	0.95	0.9	0.78	0.96	0.86	0.93
apotheek	44	42	42	40	5	4	0.89	0.91	0.95	0.95	0.92	0.93
kaas	8	7	7	6	2	0	0.78	1.0	0.88	0.86	0.82	0.92
stukadoor	37	36	28	33	2	3	0.93	0.92	0.76	0.92	0.84	0.92
rijschool	99	98	91	91	11	10	0.89	0.9	0.92	0.93	0.91	0.91
busverhuur	63	61	57	53	4	3	0.93	0.95	0.9	0.87	0.92	0.91
lingerie	11	10	8	9	3	1	0.73	0.9	0.73	0.9	0.73	0.9
psycholoog	17	16	10	14	8	1	0.56	0.93	0.59	0.88	0.57	0.9
advocaat	198	196	185	175	38	32	0.83	0.85	0.93	0.89	0.88	0.87
bloemist	106	105	96	87	10	7	0.91	0.93	0.91	0.83	0.91	0.87
dierenarts	88	87	77	69	4	3	0.95	0.96	0.88	0.79	0.91	0.87
basisschool	181	180	166	169	47	39	0.78	0.81	0.92	0.94	0.84	0.87
airconditioning	55	54	50	48	12	9	0.81	0.84	0.91	0.89	0.85	0.86
vertaalbureau	17	15	15	12	2	1	0.88	0.92	0.88	0.8	0.88	0.86
taxibedrijf	49	47	31	41	11	7	0.74	0.85	0.63	0.87	0.68	0.86
dakdekker	100	99	83	86	19	15	0.81	0.85	0.83	0.87	0.82	0.86
autobedrijf	172	171	138	146	38	27	0.78	0.84	0.8	0.85	0.79	0.85
vloer	74	74	66	66	22	15	0.75	0.81	0.89	0.89	0.81	0.85

TABLE C.6: SVM per-class performance metrics of best 25 industries based on test F1-score (satisfactory industries shown in bold).

C.8 SVM Confusion Matrix

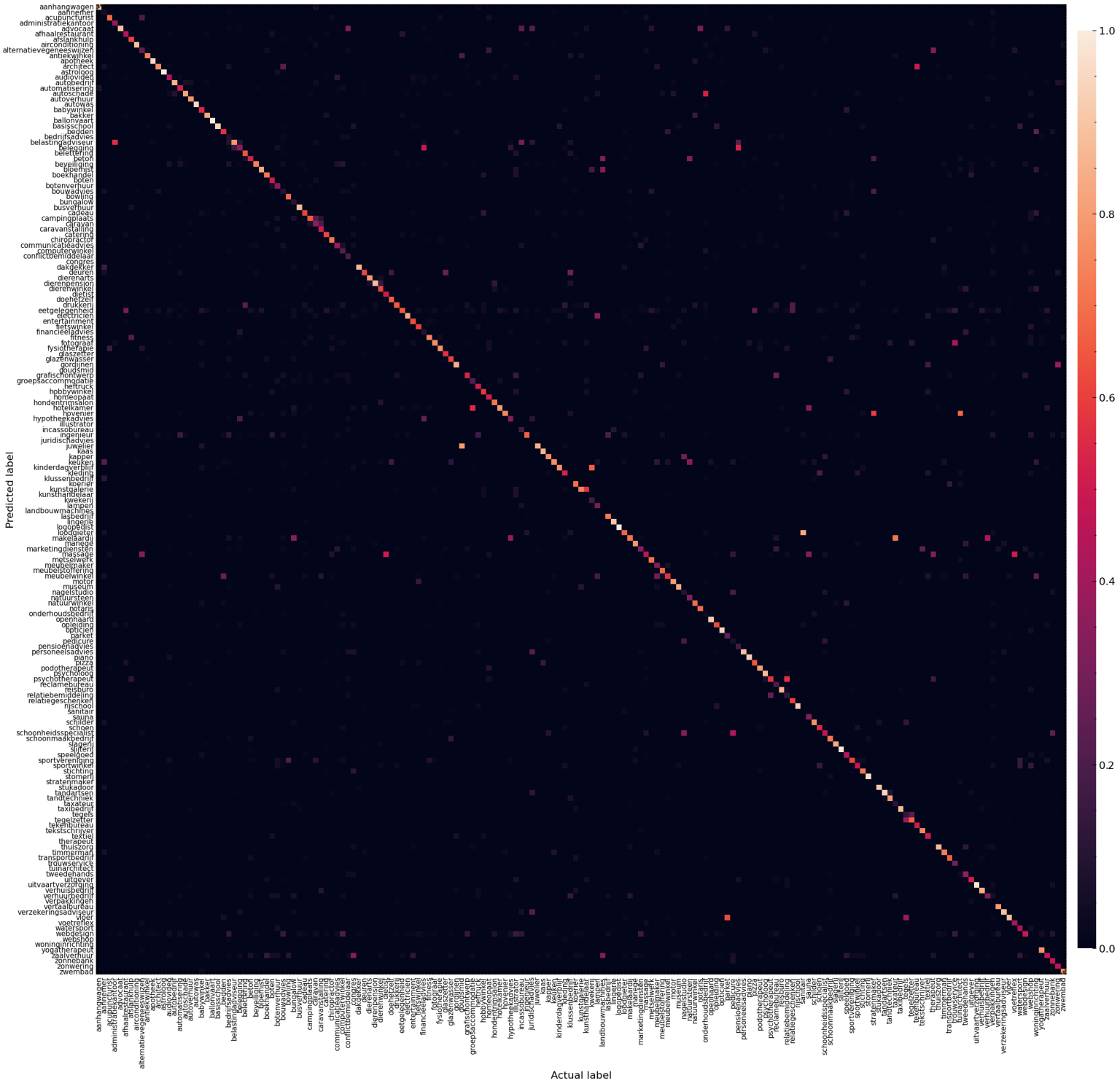

FIGURE C.2: Confusion matrix of SVM classifier on test data.

C.9 BERT Individual Experiments

Model configuration	Precision	Recall	F1	TTPE
Max sequence length: 100, lr: 5e-5, warmup: 0.1, model: $BERT_{BASE}$	0.59	0.57	0.58	**15min**
Max sequence length: 100, lr: 3e-5, warmup: 0.1, model: $BERT_{BASE}$	0.58	0.55	0.56	**15min**
Max sequence length: 100, lr: 2e-5, warmup: 0.1, model: $BERT_{BASE}$	0.57	0.55	0.56	**15min**
Max sequence length: 200, lr: 5e-5, warmup: 0.1, model: $BERT_{BASE}$	0.60	0.57	0.58	25min
Max sequence length: 512, lr: 5e-5, warmup: 0.1, model: $BERT_{BASE}$	0.60	**0.59**	0.59	65min
Max sequence length: 512, lr: 5e-5, warmup: 0.1, model: XLM-R	0.60	**0.59**	0.59	135min
Max sequence length: 200, lr: 5e-5, warmup: 0.1, model: $BERT_{LARGE}$	0.59	0.57	0.58	80min
Max sequence length: 512, lr: 5e-5, warmup: 0.05, model: $BERT_{BASE}$	**0.61**	**0.59**	**0.60**	65min

TABLE C.7: BERT model configurations with corresponding validation macro-averaged precision and recall performance, with resulting F1-score; and average Training Time Per Epoch (TTPE). The best results and the chosen best configuration are shown in bold. (TTPE rounded to closest 5 minutes)

C.10 BERT 10-Fold Cross Validation Results

Fold	Precision	Recall	F1-Score	Accuracy
1	0.646310	0.624631	0.618994	0.693586
2	0.648767	0.607565	0.606857	0.697161
3	0.655072	0.634676	0.632225	0.706391
4	0.642961	0.618077	0.617768	0.697384
5	0.644979	0.617630	0.615973	0.689210
6	0.648512	0.626422	0.620254	0.696733
7	0.654281	0.616886	0.616655	0.699174
8	0.660159	0.641181	0.631501	0.708262
9	0.643192	0.614154	0.611421	0.690247
10	0.671661	0.635006	0.630563	0.701899
Mean	0.65	0.62	0.62	0.70
Std	0.009	0.01	0.008	0.006

TABLE C.8: BERT cross validation results per fold, mean and standard deviation (std) shown.

C.11 BERT Per-class Performance

Label	Support		TP		FP		Precision		Recall		F1-Score	
	Val	Test	Val	Test	Val	Test	Val	Test	Val	Test	Val	Test
astroloog	3	2	1	2	0	0	1.0	1.0	0.33	1.0	0.5	1.0
kaas	8	7	8	7	1	0	0.89	1.0	1.0	1.0	0.94	1.0
ballonvaart	3	2	3	2	2	0	0.6	1.0	1.0	1.0	0.75	1.0
uitvaartverzorging	71	70	70	69	1	4	0.99	0.95	0.99	0.99	0.99	0.97
opticien	86	85	86	81	0	2	1.0	0.98	1.0	0.95	1.0	0.96
tandartsen	148	146	144	139	13	6	0.92	0.96	0.97	0.95	0.94	0.96
openhaard	38	37	35	36	5	3	0.88	0.92	0.92	0.97	0.9	0.95
autowas	25	23	24	22	2	2	0.92	0.92	0.96	0.96	0.94	0.94
apotheek	44	42	41	40	8	4	0.84	0.91	0.93	0.95	0.88	0.93
logopedist	6	5	6	5	2	1	0.75	0.83	1.0	1.0	0.86	0.91
stomerij	27	27	21	26	1	5	0.95	0.84	0.78	0.96	0.86	0.9
bloemist	106	105	100	91	6	7	0.94	0.93	0.94	0.87	0.94	0.9
rijschool	99	98	92	92	13	14	0.88	0.87	0.93	0.94	0.9	0.9
stichting	25	24	17	21	6	2	0.74	0.91	0.68	0.88	0.71	0.89
basisschool	181	180	173	170	29	37	0.86	0.82	0.96	0.94	0.9	0.88
airconditioning	55	54	51	49	14	8	0.78	0.86	0.93	0.91	0.85	0.88
busverhuur	63	61	55	53	3	6	0.95	0.9	0.87	0.87	0.91	0.88
taxibedrijf	49	47	32	42	12	6	0.73	0.88	0.65	0.89	0.69	0.88
yogatherapeut	37	37	31	30	1	1	0.97	0.97	0.84	0.81	0.9	0.88
piano	15	14	15	13	3	3	0.83	0.81	1.0	0.93	0.91	0.87
stukadoor	37	36	25	30	1	3	0.96	0.91	0.68	0.83	0.79	0.87
vertaalbureau	17	15	14	12	2	1	0.88	0.92	0.82	0.8	0.85	0.86
bowling	21	19	18	16	1	2	0.95	0.89	0.86	0.84	0.9	0.86
lingerie	11	10	10	9	2	2	0.83	0.82	0.91	0.9	0.87	0.86
dierenarts	88	87	73	66	2	2	0.97	0.97	0.83	0.76	0.9	0.85
museum	32	31	23	27	1	6	0.96	0.82	0.72	0.87	0.82	0.84
zwembad	18	17	14	13	6	1	0.7	0.93	0.78	0.76	0.74	0.84
manege	24	23	21	21	5	6	0.81	0.78	0.88	0.91	0.84	0.84
kinderdagverblijf	156	155	136	116	11	10	0.93	0.92	0.87	0.75	0.9	0.83
dakdekker	100	99	81	86	31	23	0.72	0.79	0.81	0.87	0.76	0.83
hondentrimsalon	40	39	37	34	14	9	0.73	0.79	0.92	0.87	0.81	0.83
slagerij	30	30	24	24	5	4	0.83	0.86	0.8	0.8	0.81	0.83
autobedrijf	172	171	136	146	28	34	0.83	0.81	0.79	0.85	0.81	0.83
advocaat	198	196	181	172	52	44	0.78	0.8	0.91	0.88	0.84	0.83
autoverhuur	63	61	44	49	16	9	0.73	0.84	0.7	0.8	0.72	0.82
juwelier	62	60	49	53	14	16	0.78	0.77	0.79	0.88	0.78	0.82
antiekwinkel	58	57	49	42	30	3	0.62	0.93	0.84	0.74	0.72	0.82

TABLE C.9: BERT per-class performance metrics of best 37 industries based on test F1-score (satisfactory industries shown in bold).

C.12 BERT Confusion Matrix

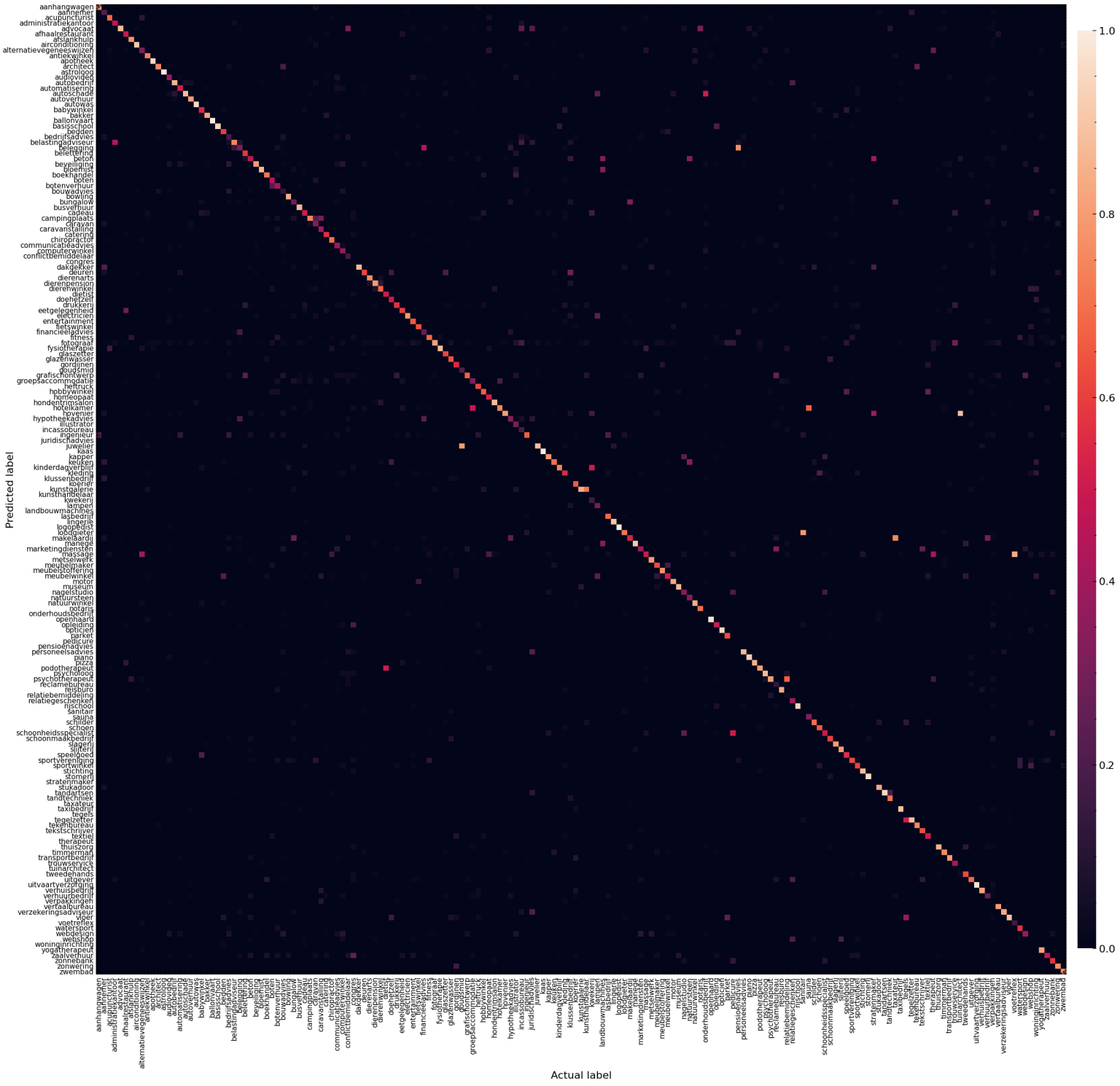

FIGURE C.3: Confusion matrix of BERT classifier on test data.

C.13 Voter Individual Experiments

Voting strategy	**Precision**	**Recall**	**F1**
Hard	0.63	0.55	0.59
Intermediate	**0.64**	0.57	0.60
Soft	**0.64**	**0.59**	**0.61**

TABLE C.10: Voting strategies with corresponding validation macro-averaged precision and recall performance, with resulting F1-score. The best results and the chosen best configuration are shown in bold.

C.14 Voter Per-class Performance

Label	Support		TP		FP		Precision		Recall		F1-Score	
	Val	Test	Val	Test	Val	Test	Val	Test	Val	Test	Val	Test
astroloog	3	2	1	2	0	0	1.0	1.0	0.33	1.0	0.5	1.0
kaas	8	7	8	7	1	0	0.89	1.0	1.0	1.0	0.94	1.0
ballonvaart	3	2	3	2	2	0	0.6	1.0	1.0	1.0	0.75	1.0
uitvaartverzorging	71	70	70	69	1	4	0.99	0.95	0.99	0.99	0.99	0.97
opticien	86	85	86	81	0	2	1.0	0.98	1.0	0.95	1.0	0.96
openhaard	38	37	34	35	5	1	0.87	0.97	0.89	0.95	0.88	0.96
tandartsen	148	146	144	140	12	9	0.92	0.94	0.97	0.96	0.95	0.95
autowas	25	23	24	22	2	2	0.92	0.92	0.96	0.96	0.94	0.94
stomerij	27	27	21	27	1	4	0.95	0.87	0.78	1.0	0.86	0.93
apotheek	44	42	41	40	8	4	0.84	0.91	0.93	0.95	0.88	0.93
busverhuur	63	61	55	53	3	3	0.95	0.95	0.87	0.87	0.91	0.91
rijschool	99	98	92	92	13	13	0.88	0.88	0.93	0.94	0.9	0.91
logopedist	6	5	6	5	2	1	0.75	0.83	1.0	1.0	0.86	0.91
lingerie	11	10	10	9	2	1	0.83	0.9	0.91	0.9	0.87	0.9
airconditioning	55	54	51	51	11	8	0.82	0.86	0.93	0.94	0.87	0.9
bloemist	106	105	100	91	6	7	0.94	0.93	0.94	0.87	0.94	0.9
stukadoor	37	36	27	31	2	3	0.93	0.91	0.73	0.86	0.82	0.89
advocaat	198	196	185	177	40	30	0.82	0.86	0.93	0.9	0.87	0.88
taxibedrijf	49	47	32	42	12	6	0.73	0.88	0.65	0.89	0.69	0.88
basisschool	181	180	173	172	34	40	0.84	0.81	0.96	0.96	0.89	0.88
yogatherapeut	37	37	31	30	2	1	0.94	0.97	0.84	0.81	0.89	0.88
zwembad	18	17	14	13	5	0	0.74	1.0	0.78	0.76	0.76	0.87
museum	32	31	23	27	0	5	1.0	0.84	0.72	0.87	0.84	0.86
vertaalbureau	17	15	15	12	2	1	0.88	0.92	0.88	0.8	0.88	0.86
antiekwinkel	58	57	50	44	34	3	0.6	0.94	0.86	0.77	0.7	0.85
dakdekker	100	99	82	88	26	21	0.76	0.81	0.82	0.89	0.79	0.85
dierenarts	88	87	74	66	2	2	0.97	0.97	0.84	0.76	0.9	0.85

TABLE C.11: Voter per-class performance metrics of best 27 industries based on test F1-score (satisfactory industries shown in bold).

C.15 Voter Confusion Matrix

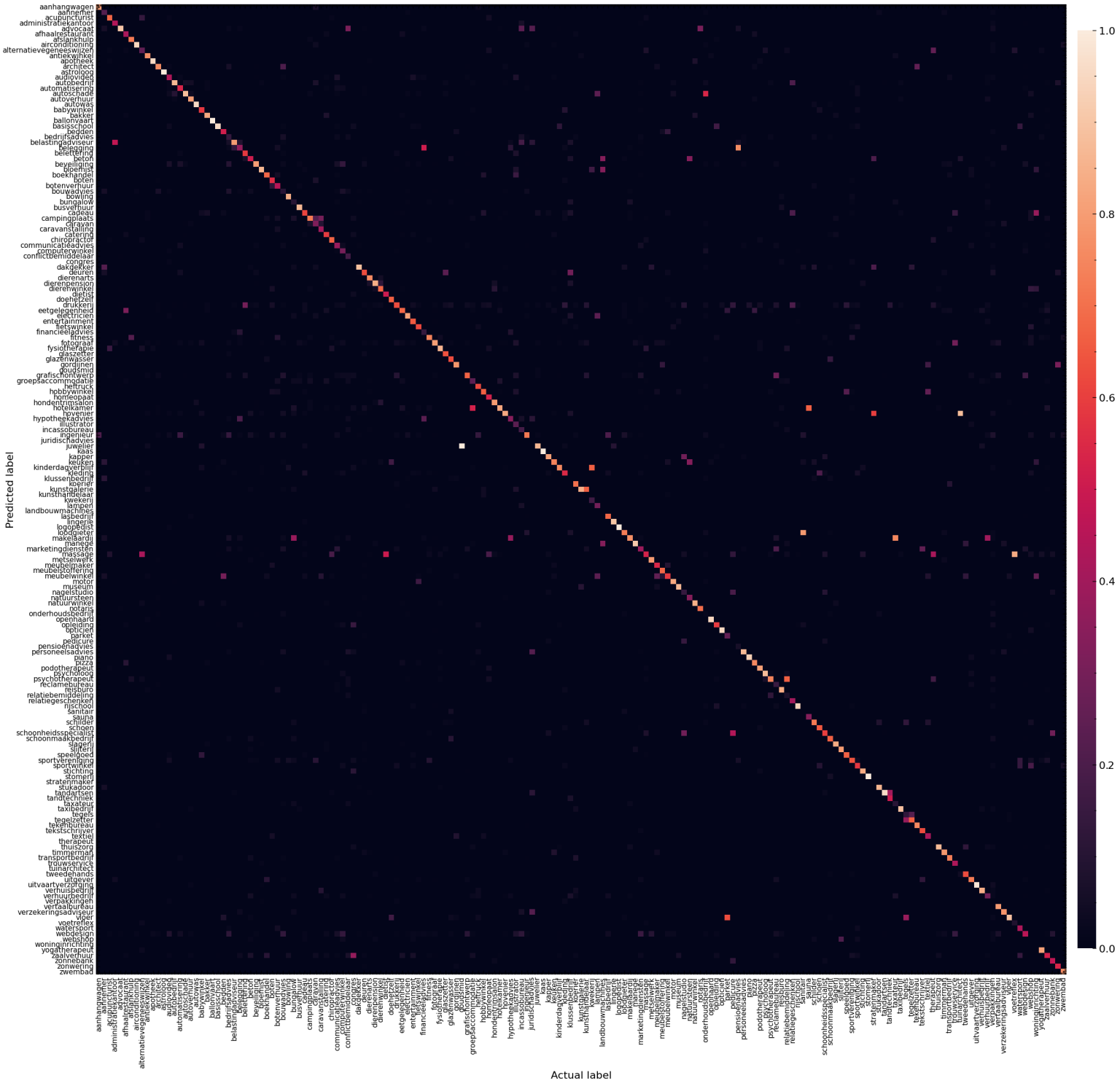

FIGURE C.4: Confusion matrix of Voting classifier on test data.

C.16 Training Size Correlation with Performance

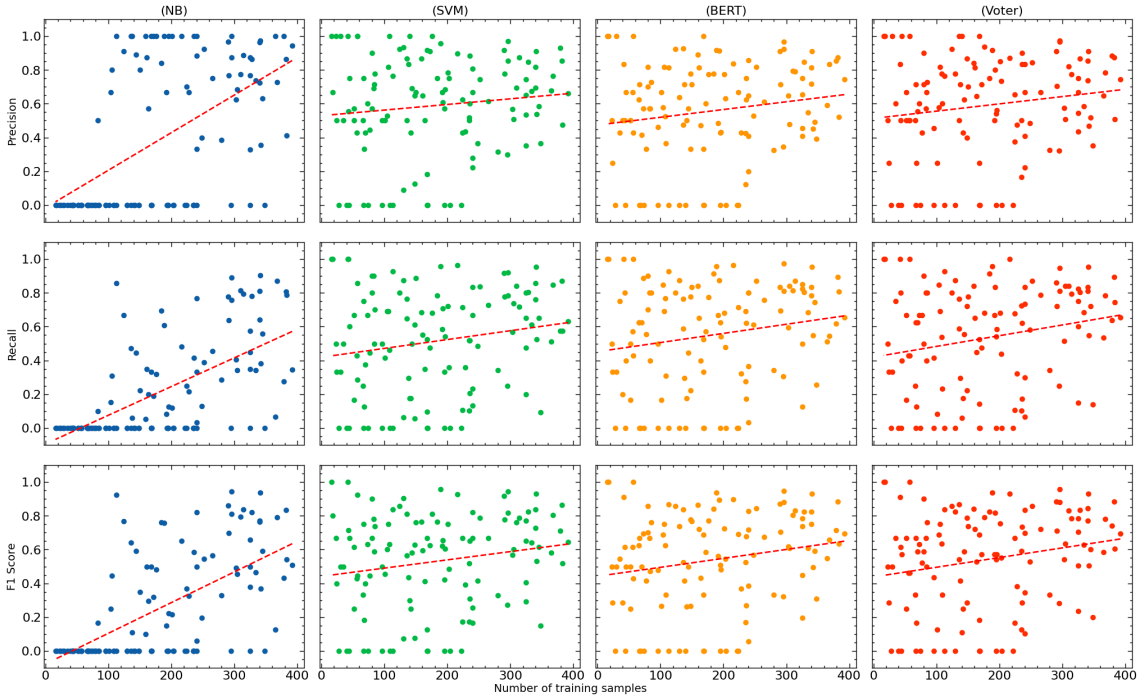

FIGURE C.5: Relationship between test performance and the number of training samples for each model, where train size <= 400 (each dot is an industry).

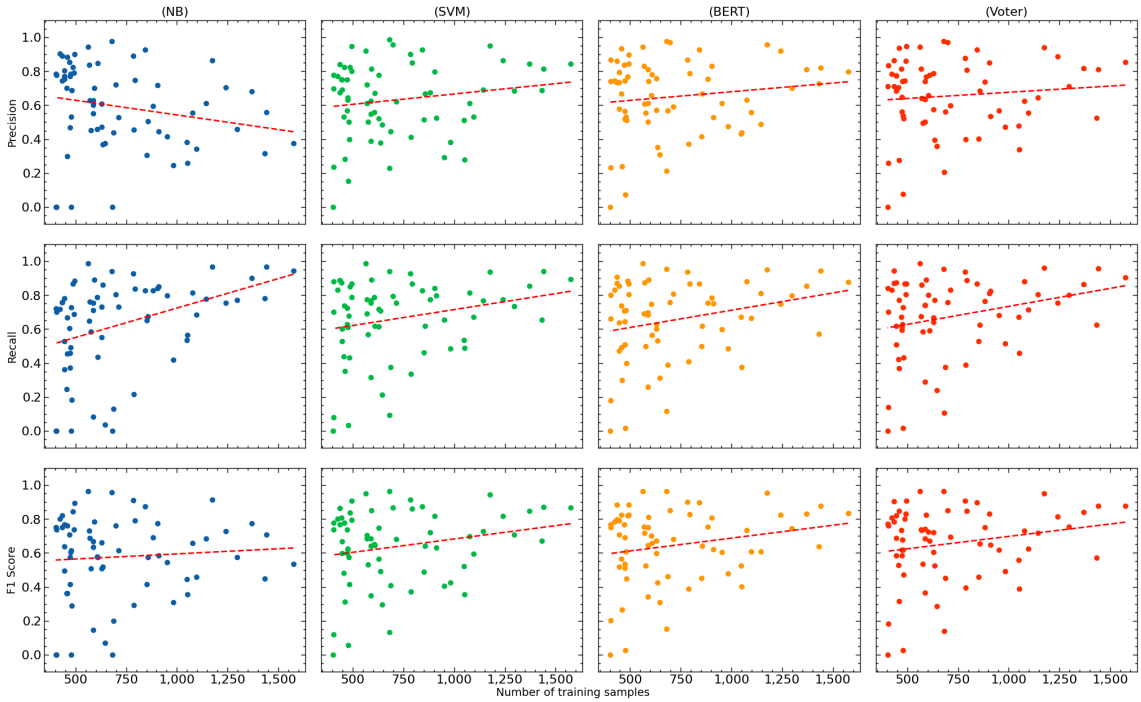

FIGURE C.6: Relationship between test performance and the number of training samples for each model, where train size >= 400 (each dot is an industry).