Detecting the online romance scam: Recognising images used in fraudulent dating profiles

Koen de Jong[1]

[1]Department of EEMCS, University of Twente,

November 21, 2019

Abstract

The online romance scam is a scam with a high financial impact as well as a high emotional impact on the victims. Neither law nor awareness campaigns have proven to be effective against this scam, so a technical solution might be needed. This paper looks at a technique which can be used in such a technical solution. We present a classifier which is trained to recognise images used in the romance scam by using the occurrence of these images on the web. Besides this a dataset is constructed consisting out of images used in the romance scam complemented with normal images of people. We achieve an accuracy of 92.4% combined with a false negative rate of 19.7% using a random forest classifier. Although the results are promising, further research is needed to amongst others lower the false negative rate before this technique can be implemented in an end-user tool.

Keywords: Online Romance scam, Reverse Image Search, Text Classification, Machine Learning

1 Introduction

The rise of the internet has changed a lot in the world of dating by opening up the possibilities to get in touch with way more people than before. To that extent it is no surprise that online dating became a booming business. In 2013 already 13% of the people in the Netherlands met their lover online [1]. This was even before the rise of apps such as Tinder. In 2015 15% of U.S. adults had used online dating websites or apps [2]. It is reported that nowadays one in three people connected to the internet has used or is using online dating websites or apps, although people do not necessarily use them for finding new relationships, but also use them just for fun [3]. With the growing popularity of online dating a new scam became apparent: the online romance scam. This scam is introduced in the next subsection.

1.1 The online romance scam

The online romance scam is a relatively new scam, which takes place online. The scam works in the following way: Scammers contact potential victims on dating sites or other social media platforms using a fake online profile using a photo found on the internet. During the following months they develop what the victim believes to be a true romantic relationship. Eventually the scammer will ask for money using reasons such as the wish to visit the victim. The scammer often requests the money in untraceable ways and of course never shows up [4].
This romance scam in the online world is simply referred to as the online romance scam. It became apparent around 2008 [5] and is closely related to catfishing, although catfishing does not necessarily involve money and the goal and motivation of a catfisher does not need to be financial gain.

The financial impact of the online romance scam is high. Losses in the USA exceed $ 200 million on a yearly basis [6, 7]. The Dutch Fraudehelpdesk received 134 reports from victims in 2017, together losing around €1,5 million in that year [8]. Although the financial loss of victims is high, it is not the most upsetting part of the scam. Victims experience the loss of the relationship once they find out that they are being scammed more traumatising. They experience emotions such as shame, anger and stress or even feel suicidal once they find out that their relationship is not real [9].
As the impact of the romance scam is high, there is a need for effective countermeasures. As scammers

often operate from abroad, making it hard to find and arrest them, law has not proven itself to be an effective solution [10, 11]. Neither do awareness campaigns help from stopping this scam, as awareness of the online romance scam does not necessarily prevent individuals from becoming victimised [12].

1.2 The current approach

The Fraudehelpdesk is a Dutch organisation that has the task of collecting all reports of fraud within the Netherlands. Besides this they try to prevent Dutch citizens and organisations from becoming victimised by fraud and they support victims of fraud by getting them in touch with the right organisations [13]. When people reach out to the Fraudehelpdesk because they are worried they might be victimised by an online romance scam, the help desk staff uses multiple techniques to check if this is actually the case. One of these techniques is the use of reverse image search engines such as Google reverse image search. For this they use the image of the person with which the potential victim is dating. If the search result shows that the image is used on multiple dating sites, for example with different names, it is likely to be a scam.
One of the drawbacks is that people usually reach out to the Fraudehelpdesk when it is already too late. That is why the Fraudehelpdesk also advises people that are dating online to use reverse image search themselves [14]. However, people will probably only do a reverse image search themselves after recognising "red flags" regarding the people they are dating online. The problem is that potential victims easily overlook these "red flags" [11]. A software tool or agent that raises the "red flags" and cannot be overlooked or ignored, is suggested as a suitable solution [11]. However, such a technical solution does to our knowledge not yet exist. That is why this study aims at exploring techniques that can be implemented in such a tool.

1.3 Research objectives

The aim of this thesis is to explore techniques that can help people from becoming victimised in an online romance scam. In this research we present a machine learning based classifier which uses the output of reverse image search engines to recognise scammers. Techniques from the fields of natural language processing, text classification and web page classification are used to extract useful features from the from the reverse image search results and train the classifiers.
Considering the above, the research objective can be formulated in the following way:

RO: *Design and evaluate a classifier that recognises images used in online romance scams, by looking at their occurrence on the internet.*

The key contributions of this research are:

- Generating a dataset usable for classifying romance scams by collecting images used by scammers in romance scams as well as normal images.
- Exploring the use of reverse image search and the results of the queries for recognition of the romance scam.
- Design of different machine learning based classifiers for the recognition of images used in the romance scams.
- Comparison of performance of the different machine learning based classifiers in recognising images used in the romance scam.

This thesis is organised in the following way: In section 2 the current status of the related fields is presented. Section 3 gives a more elaborate description of the proposed research. Section 4 presents the result achieved by the classifiers. These results are discussed in section 5. The conclusions can be found in section 6.

2 Related Work

The focus of this thesis is the development of a technical countermeasure against the online romance scam. For this, techniques from the fields of reverse image search, natural language processing, text categorisation and web page classification will be used. The state of the art regarding these fields will be discussed below.

2.1 The online romance scam

The online romance scam is a relatively new scam, which came apparent around 2008 [5]. Beals et al. [4] describe the online romance scam in the following way:
"*A type of 'Relationship and Trust Fraud'. In these scams, victims are contacted in-person or online by someone who appears interested in them. In many cases, the fraudster sets up a fake online profile using a photo found on the internet ('catfishing'). Over the course of weeks or months, they develop what the victim believes to be a true romantic relationship. Eventually, the perpetrator will ask for money for a variety of reasons, which may include wanting to visit the victim but being unable to afford the flight, needing to clear a debt, or wanting to help a dear relative. The money is often requested in un-traceable ways, like a money order or a prepaid card.*"
Other papers, such as [5, 9, 10, 11, 15, 16] use a similar description of the online romance scam.

Although that the online romance scam has been widely studied, most studies focus on the victim, what in particular makes the victims vulnerable for the scam and the impact of the scam on the victim [5, 9, 10, 12, 15, 16]. They recognise that the romance scam is particularly traumatising due to a double hit effect: victims not only lose money, they also experience the loss of a significant romantic relationship once they find out that they are being scammed.

Besides the impact on the victims, the financial impact is properly registered as well. Reports from Internet Crime Complaint Center of the FBI report losses of over $ 200 million a year in the USA [6, 7]. The UK Fraud Cost Measurement Committee reported the damage due to mass-marketing frauds, including the online romance scam was £ 4.5 billion in 2016 [17]. The Dutch Fraudehelpdesk received 313 notifications about the romance scam in 2017 [8]. Of these 313 notifications, 134 people reported to be victimised, together losing around € 1.5 million. As the impact is high, effective countermeasures are needed. Suggested and researched countermeasures are discussed below.

2.1.1 Countermeasures

Research has made clear that both the emotional and financial impact of the romance scam is high, and scammers are well aware of how to lure victims into the scam. Unfortunately, law has not proven to be an effective solution against this scam [10, 11]. One of the issues is that it is hard to find and arrest the scammers as they are often operating from abroad. Awareness campaigns are not effective solutions either, as awareness of mass-marketing fraud doesn't necessarily prevent individuals from becoming victims. This is why other types of intervention are needed [12].
Norta et al. [11] suggests that a technical solution such as a software tool that raises the "red flags" and cannot be overlooked or ignored might be needed. They also suggest that Google image search might provide to be a useful tool.

There are a number of studies which look into the use of (machine learning based) classifiers for the detection of malicious activities such as the classification of spam and phishing websites [18, 19, 20, 21, 22, 23, 24, 25, 26]. The drawback of these studies is that they often use more complex features such as the number of JavaScript elements. Although these elements are useful in the recognition of not legitimate websites, these elements are not relevant in the field of the romance scam as we deal with fraudulent profiles on legitimate websites.

It was only until recently that there was no research done on technical solutions specific for the online romance scam. To our knowledge Suarez-Tangil et al. [27] are the first to look into a technical solution specific for the online romance scam. They create a dataset by extracting publicly available legitimate dating profiles from a dating website and extracting fraudulent profiles from a forum related to this dating website. The data of a profile consists out of demographic information, one or multiple images and a profile description which a user can write him- or herself. For each of these three parts, they create and extract features differently and train a classifier. In the end a weighted vote over the three classifiers is used to come to a final classification. They obtain an accuracy of 97 %.
A drawback of this study is that the dataset consists out of profiles from one single dating site. Besides that, the contact between two people dating does not necessarily needs to take place at a dating site. Both in the case that contact takes place at another dating site as well as in the case that contact takes place outside of a dating site, demographic information and a profile description might not be available or not available in the right format. This is why in this research we will only look at the image and

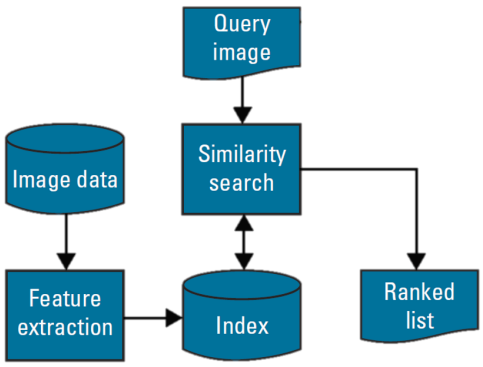


Figure 1: Main components of a typical CBIR system [30].

its recurrence online. For looking at the recurrence online, we use reverse image search engines. The relevant techniques for this topic are discussed in the next subsection.

2.2 Reverse image search

Reverse image search is the technique which focuses on making a query with an image as input. The research field related to this problem is often referred to as Content-Based Image Retrieval (CBIR). The field became apparent in the 1990s and grew quickly in the years after [28]. Content Based Image Retrieval is defined as an automated technique that takes an image as query and returns a set of images similar to the query [29]. How a CBIR system usually works is visualised in figure 1 and can be explained in the following way:

- Visual features are extracted from the images' pixel data. This extraction uses specialised image processing algorithms that encode the visual content into a feature vector.
- The extracted features are stored in an index
- A query image is uploaded and indexed. A similarity search is done against the previously stored index.

The first commercial CBIR system was QBIC which was released in the 1990's by IBM [31]. Although more CBIR systems were developed through the years, it was only in 2008 that TinEye released the first reverse image search engine freely available to the public on the web [32]. Three years later, Google started offering reverse image search in Google Images. Nowadays, most major search engines such as Bing, Yandex (Russia's most used search engine) and Baidu (China's most used search engine) offer reverse image search as well [32].

Although there has been a lot of research done regarding techniques used in CBIR, there is only little written about the performance of commercial reverse image search engines such as Google reverse image search. The studies that do exist have as a drawback that they only use a small dataset, with usually less than 100 images. Terras et al [33] and Kelly [34] both compare the performance of Google reverse image search and TinEye. They both conclude that Google reverse image search performs best. This is also what Nieuwenhuysen [35] concludes after comparison of Google, Yandex and TinEye. In this research Google gives the most results, followed by Yandex. This is in line with the number of images that the search engines have indexed.

2.3 Text classification

Although there is only one paper which uses machine learning for the recognition of romance scams, there are other well studied fields which will be used in this thesis. One of these fields is text classification and in particular web page classification. This field focuses on labelling texts by topic. For this, techniques from the natural language processing (NLP) field are often used alongside machine learning techniques. Web page classification also tries to label by topic. Only in this particular fields, it is possible to use the structure of the HTML-documents as well as extra content such as links, HTML-tags and JavaScript tags.

Jindal et al. [36] explore this field by reviewing 132 studies. They find that 83% of the studies use a bag-of-words (BoW) or a comparable vector space model as features to represent the data in documents. 86% of the studies use a machine learning method for classification. SVM is the most used algorithms, followed by k-Nearest Neighbours (kNN), Naive Bayes (NB) and Artificial Neural Networks (NN).

To get an overview of the current state of the art in the field of text and web page classification, a structured literature review was conducted. A total of 34 papers were reviewed. An overview of the reviewed papers, the used features, used machine classifiers and obtained performance can be found in appendix A. An overview of the most important conclusions is presented below:

2.3.1 Feature selection

First of all, we can conclude that a good choice of features is important. The most basic feature set that can be used is bag-of-words, which is used in more than half of the papers. Good performance can be obtained with bag-of-words. However, improvement can be made by using n-grams. A bigger step can be made by using Term-Frequency - Inverse Document Frequency (TF-IDF). The drawback of TF-IDF is that is more time complex than bag-of-words and n-grams.

2.3.2 Machine learning

We see that a lot of classifiers have been studied in the field of text and web page classification. Most of the used methods use machine learning for classification. The most used methods are Naive Bayes, Support Vector Machines, Decision Trees and Random Forests. These algorithms and their performance will be discussed below:

Naive Bayes Of the 34 reviewed papers, 21 used Naive Bayes. Naive Bayes is so often used as it is a relatively simple classifier. It uses Bayes' theorem, which assumes independence between features. Although this is often not the case, Naive Bayes classifiers work reasonably well in most cases. However, their performance is usually not the best either and the obtained accuracy by naive Bayes classifiers in the field of text classification usually does not come above 90%.

Support Vector Machine The other classifier that is used in more than half of the papers is Support Vector Machines, which is used in 56% of the reviewed papers. It is a supervised learning method to classify data in two categories. In the simplest form it is a linear classifier, but by applying the kernel trick, it can also be used for non-linear classification. Compared to Naive Bayes, SVM performs better in most cases and is also more consistent in the performance that can be achieved. In most cases, the accuracy of SVM classifiers in text categorisation is above 90%.

Decision Tree Although decision trees are not used as much as the methods above, they are discussed in over a quarter of the reviewed papers. The advantage of decision trees is that they are considered to be white box classifiers, which makes it easier to explain how a classification is made. A disadvantage on the other hand is that they are known to be vulnerable to over fitting. In the field of web page classification accuracies over 95% are achieved. However, it should be kept in mind that these studies can use more structured features than text only.

Random Forest To overcome the problem of over fitting an ensemble learning method can be used, a model which combines multiple classifiers into one classifier. Random forest is such a method, which consists out of multiple decision trees. Although it is only used in 4 of the reviewed papers, the results are generally good. Onan et al. [37] concludes that it outperforms Naive Bayes and SVM. In combination with bagging they achieve an accuracy of 94% for text categorisation.

Looking at the field of Text Classification and in particular the field of Web Page classification we can conclude that some of the most used machine learning algorithms include SVM, NB DT and RF. Not only the used algorithm is important, the used features are just as important. In the field of text classification and web page classification bag-of-words, n-grams and TF-IDF are most often used as features.

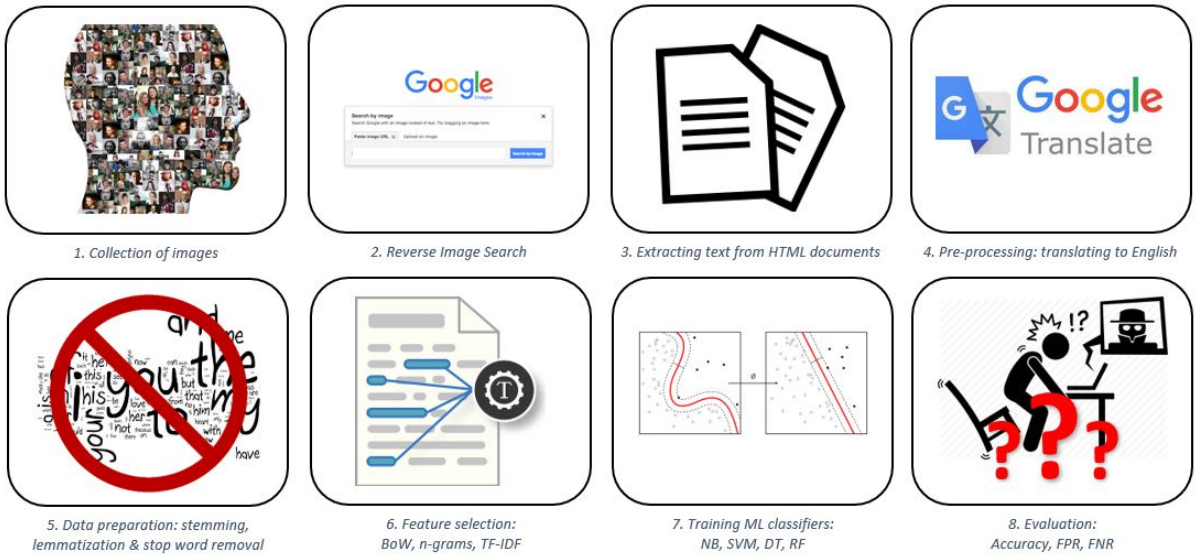


Figure 2: Pipeline of the research

There is still a lot of research happening in the field of text and web page classification. Most papers focus on adjusting one of the mentioned feature selection techniques or machine learning algorithms and try to adjust these to improve the performance. Although most papers are able to improve performance in specific cases, there are no solutions that seem to increase the performance for all datasets discussed in these papers. On top of that these solutions often make the models more complex.
Considering this complexity while looking at the scope of this thesis, we will use the standard methods of the machine learning algorithms.

3 Methodology

The objective of this research is to design, evaluate and analyse a classifier that recognises images used in online romance scams, by looking at their occurrence on the internet. In this section we will elaborate on how that is done.
The first part of this research consists out of gathering data by the collection of images. After that reverse image search will be used and the text from the web pages where these images occur need to be extracted. How this is all done, is discussed in subsection 3.1. In subsection 3.2 some of the methods to further prepare the data for machine learning are discussed. Subsection 3.3 presents how the features are extracted from the data. These features can then be used for training a machine learning algorithm. The used algorithms and tested parameter settings are presented in subsection 3.4. Last the model evaluation measures are discussed in subsection 3.5. A visualisation of the pipeline of this research can be found in figure 2

3.1 Data Collection

One of the challenges of this research was, that there was no dataset available which was suitable for the purposes of this research. This meant that a dataset had to be constructed. How this was done is discussed below. Some ethical and legal issues related to the collection of this data are discussed in section 3.1.5

3.1.1 Collection of images

The data in this research consists out of images. We decided to use images instead of a complete dating profile as it is likely that most people will have an image of the person they are dating with online. Although a complete dating profile might contain extra information about demographics or a short description of the person, the availability of this information can be platform specific. On the other hand, when using an image, there is no limitation to the service or platform used, whether this is e-mail, a chat service or a dating site or app.

(a) (b) (c) (d)

Figure 3: Examples of images used by scammers (a,b) and normal images (c,d)

One of the drawbacks of using images, was that there was no standard dataset available. Luckily, there are many websites collecting information which warns for the online romance scam and which display images used in the scam. However, for training also a dataset with negative examples is needed, so two datasets were generated.
The first consists out of images known to be used in the online romance scam. These images were extracted from online forums warning for scammers. The second dataset consists of general images of people. These were selected from websites and online available datasets offering images which are free to use (in some cases only for personal or research purposes). The images in this second dataset were selected in such a way that they are visual comparable to the first dataset, meaning that they could be used as a profile picture for a social media account.
Due to ethical and juridical reasons, as further discussed in section 3.1.5, only the URLs linking to the images were saved for both datasets. A list of websites from which the images were collected can be found in appendix B. In case the websites presented the images in a structured way, the URLs of the images were extracted in an automated way using ScraPy[1].
In total 2447 imaged used by scammers and 2449 normal images were selected for further processing. Examples of the selected images can be found in figure 3. More examples can be found in appendix C.

3.1.2 Reverse Image Search

After that collection of the images took place, the images were searched using the reverse image search. To decide on which reverse image search engines to use, we had a look at the literature (section 2.2) and the potential capabilities of the search engines. Looking at literature, it can be concluded that Google and Yandex gave most relevant result, followed by TinEye. On top of that, Google and Yandex both offer the possibility to embed the URL of an image into the URL of the search engine to launch a query. This is why Google and Yandex were chosen as reverse image search engines.

The reverse image searches were done automatically using Selenium[2] in a Firefox browser. The results of these queries show recurrences or similar looking images occurring on the web. The URLs linking to the pages where the similar looking images occur were extracted and saved. For each query a maximum of only the first 5 results per search engine were saved.

Besides the use of reverse image search, we decided not to extract features from images itself in this research. As most of our images consist out of images that could be used as profile pictures, we don't expect to find any relevant differences in features that can be extracted from images between the two datasets. On top of that, we don't expect scammers to have such an advanced way of choosing which images to use in a scam.

3.1.3 Extracting text

For each original image we now have multiple URLs referring to websites in which the image or a similar looking image occurs. Of each website we extract the text that is visible to the user in a browser. This is done in an automated way using ScraPy and BeautifulSoup[3]. One of the implications of this approach

[1] `https://scrapy.org/`
[2] `https://www.seleniumhq.org/`
[3] `https://pypi.org/project/beautifulsoup4/`

is that content loaded via APIs is not contained in the text data extracted from a website. Besides this, all the extra information that is contained within a HTML document is lost.

3.1.4 Adjustment of the text

One of the problems with the extracted text is that these texts are not necessarily in the same language. Although most pages are written in English, the data also contained text in other languages. This would become a problem in training the classifier, as it might cause the classifier to recognise the language. To solve this problem all texts were translated to English using Google Translate. This was again done in an automated way using Selenium and Firefox.

During the extraction of the text data from the web pages and the interaction of Selenium and Google Translate in some cases problems occurred. These problems could be that the website no longer existed or could not be reached within the time-out time, or that Google Translate was not able to translate the text. As it was infeasible to solve this by hand, some of the data had to be disregarded. In the end text files belonging to the results extracted from 4154 out of the original 4996 images could be used for training and testing.

3.1.5 Ethical and legal reflection

As the research deals with images containing persons, these should be considered as personal identifiable information. Although these images are used by scammers, they usually steel these images from other people, meaning that the person in the image is usually not the scammer itself. By the use of the images the people in the image to some extent become a subject in the research. This raises some ethical and juridical issues. Before the start of this research these issues were considered. The full work that has been done regarding the ethical and juridical issues of this research can be found in appendix G. We will shortly mention the most important considerations below.

Whenever human beings are involved in research it is common to get informed consent. However, we do not know and are not interested in the identity of the subjects. To get informed consent we would actively try to recover the identity of the subjects. As the violation of privacy when using the images compared to the violation of privacy for retrieving the identity of the subjects is minimal, we argued it was better not to get informed consent. Besides the collection of Personal Identifiable Information was kept to a minimum. Considering that, it is reasonable to say that the benefits of this research, helping people from becoming victimised in the online romance scam, out weights the loss of privacy of the subjects.
As the research involves human beings, a proposal has been submitted to the ethical committee of the EEMCS faculty of the University of Twente[4], which has been approved under reference number *RP 2019-09*.

As we work with personal data the GDPR should be considered as well. The GDPR considers images containing people as a special category of personal data, for which extra strict requirements are needed to justify the processing. We are able to use an exception for scientific research. On top of that we suggested that a data protection impact assessment[5] (DPIA) should be considered. However, the responsible Privacy Contact Person of the University of Twente decided, that this was not needed.
The GDPR also demands that subjects are informed about the processing of personal data. However, this would require the collection of extra personal information. As this is not desirable, the GDPR also offers an exception. Besides there is an extra exception for scientific research.
Overall, the research can be considered legally justifiable. However, to get this justification some exceptions which are specifically for scientific research are used. This means, that if the results would be used for the development of a (commercial) tool, the justification of lawfulness as for this research does no longer apply and the legal justification regarding the GDPR should be considered again.
Before the start of this research, the processing of data has been reported with the DPO team of the University of Twente[6].

[4] `https://www.utwente.nl/en/eemcs/research/ethics/`
[5] `https://www.utwente.nl/en/cyber-safety/privacy/pre_dpia_form/`
[6] `https://www.utwente.nl/en/cyber-safety/privacy/`

3.2 Data preparation

At this point our dataset consists out of text-files. However, these cannot directly be used for training machine learning algorithms. Text normalisation is a technique that is known to improve performance of text classification in some cases. To see if text normalisation could improve the performance both stemming and lemmatization have been applied to the data. These techniques are explained below:

Stemming Stemming is the technique where each word is reduced to its stem (root) by removing prefixes and suffixes. The result of this does not necessarily needs to be a word itself. Stemming was applied to the data using Natural Language Tool Kit (NLTK)[7], a python library for natural language processing. It was applied using a Snowball Stemmer with the language set to English.

Lemmatization Lemmatization is a more complex technique, where a word is brought back to its first form variant, always creating an actual word. Where stemming can just use standard rules to create the stem, for lemmatization more knowledge about the meaning of the word in the sentence needs to be known. Lemmatization was applied using the implementation in the SpaCy[8] library.

Besides text normalisation, it can also useful to remove words that don't add information about the content. Stop word removal is a common technique, which does this.

Stop word removal Stop word removal is a commonly used technique in the field of natural language processing. Although there is not a single fixed list of stop words, they are considered to be the most common words in a language. As they occur so often, they are easily selected as features. However, these words do not add a big semantic meaning within a sentence, which often means that they are not useful for classification. Stop word removal tools remove the stop words from the data using a given list of stop words. Stop words from the NLTK package were used for stop word removal.

After pre-processing the data, the features can be extracted and selected from the data. How this is done, is explained in the next subsection.

3.3 Feature Selection

After normalising the text and removing stop words, it is time to select features from the data. This is done by bag-of-words, n-grams and TF-IDF. These techniques are shortly explained below.

Bag-of-words Bag-of-words, which is also referred to as uni-grams, is a commonly used method for document classification. All words in the data are represented in a bag, showing how many occurrences each word has in the data. In this way each document can be transformed to a vector, for which each index is a representation of the number of occurrences of a certain word in the document. Bag-of-words ignores more complex properties such as word order. Features were selected using the most occurring words in the training data. For the choice of the size of the bag of words we considered both the time complexity as well as the size of the dataset. The following sizes of bag-of-words were used:

- bag of words: [10, 50, 100, 250, 500, 750, 1000, all words]

N-grams N-grams are quite similar to bag-of-words. Where bag-of-words only looks at one word at a time, n-grams look at the occurrence of word combinations with a length of n words. Features were selected using the most occurring n-grams in the training data. The following n-grams with the following sizes were used:

- bi-grams: [10, 50, 100, 250, 500]
- tri-grams: [10, 50, 100, 250, 500]

[7] `https://www.nltk.org/`
[8] `https://spacy.io/`

TF-IDF Term Frequency - Inverse Document Frequency is (TF-IDF) a statistic that shows how important in a text document is compared to the occurrence in the whole corpus of all documents. This is done by calculating the term frequency of a word in the document and correcting this with the inverse document frequency, which says how often a word occurs in the whole corpus. When features are selected using TF-IDF, those with the highest TF-IDF score are selected. The following sizes were used for TF-IDF:

- TF-IDF: [10, 50, 100, 250, 500, 750, 1000]

Using these features, the machine learning algorithms were trained. Before training the features of Bag-of-Words and n-grams were first normalised using a standardised scaler. Bag-of-Words, n-grams, TF-IDF and data scaling have been applied using the Scikit Learn package. Which machine learning algorithms were used for classification and with which settings, is discussed in the next section.

3.4 Machine Learning

To find the best model, multiple machine learning algorithms were tested. These models were chosen based on the literature review conducted in section 2. All methods were implemented using the Scikit Learn package[9].

Naive Bayes Naive Bayes classifiers are relatively simple probabilistic models, which use Bayes' theorem, which assumes independence between features. The model can be varied by choosing the probability function used in Bayes' theorem. The following parameter settings were varied for this algorithm:

- Probability function: Gaussian distribution, the multinomial distribution and the Bernoulli distribution

Support Vector Machine SVM is a supervised learning method to classify data in two categories. In the simplest form it is a linear classifier, but by applying the kernel trick it can also be used for non-linear classification. The following parameter setting were varied for SVM:

- Used kernel: radial basic function, sigmoid and polynomial with degree 1 to 5.

Decision Tree Decision trees are trees consisting out of decision nodes. At each node a split is made based on some of the features, until at the leaves a classification is made. The following parameter settings have been tested:

- splitting criterion: Gini and entropy
- splitter: best and random
- max. depth: 10 to 100 with step size 10
- min. # items needed for a split: 2 to 10 with step size 2

Random Forest Random Forest is a supervised ensemble learning method. The model consists out of many decision trees, which makes it less vulnerable for over-fitting. As the model consists out of decision trees, many of the parameters that can be varied are the same. In this case the varied parameters are:

- number of trees: 1 to 200 with step size 10
- splitting criterion: Gini and entropy

[9] `https://scikit-learn.org/`

		Predicted class	
		P'	***N'***
Actual class	***P***	*TP*	*FN*
	N	*FP*	*TN*

(a) The confusion matrix

Measure	**Formula**
Accuracy	$\frac{TP+TN}{P+N}$
False Positive Rate (FPR)	$\frac{FP}{N}$
False Negative Rate (FNR)	$\frac{FN}{P}$

(b) Used performance measures and formulas

Table 1: Used performance measures

3.5 Model Training and Evaluation

To evaluate the performance of the different machine learning algorithms, the data has been split into training and test data, in an 80-20% split. The above presented machine learning algorithms have been trained with different feature sets and different parameter setting. 3-fold cross validation was used to find the best parameter settings for each combination of machine learning algorithm and feature set. The best parameter setting was chosen using accuracy.

For evaluation of the performance of the different feature sets and parameter setting on the test set accuracy (Acc), False Positive Rate (FPR) and False Negative Rate (FNR) will be used. These evaluation measures and their formulas are given in table 1. For evaluation we label images used by scammers as positive (P) and normal images as negative (N). This means that the FPR indicates how often it happens that a genuine person is considered as a scammer by the model. On the other hand, the FNR would indicate how often a scammer is not recognised as such by the model. Both of these cases could have a violent effect on the user of our model.
In the next section we will present the obtained performance of the models using different feature sets and different parameter settings.

4 Evaluation

In the previous section we described how the dataset was constructed and which models are used in this research. In this section we will present the obtained results of the classifiers and discuss the most important findings. We ran all combinations of selected features: we used the different n-grams and TF-IDF feature vectors, with lemmatization, stemming or neither of these two and with and without stop word removal. The Bag-of-Words in which all words were included for training, could not be used as feature vector as it was too big to be handled by Scikit Learn.

A complete overview of the results can be found in appendix F. In this section we present a selection of the results. An overview of the achieved accuracy for all different classifiers can be found in table 2. In tables 3, 4, 5 and 6 a more elaborate overview per classifier can be found. These were selected based on the highest accuracy for that size of feature set. Results presented in bold are those which gave the best performance for that size of the feature set. Below we will discuss the findings per machine learning algorithm. Finally, we will also present the overall findings.

4.1 Naive Bayes

The first classifier we trained was a Naive Bayes classifier. The results of this classifier can be found in table 3. These were selected from tables, 16, 17, 18 and 19 in appendix F.
Looking at the results we see that the naive Bayes classifier performs well compared to the SVM and decision tree classifiers. In most cases the accuracy is more or less the same and for the bigger features sets even better.

Looking at the performance of the classifier while using uni-grams, we see that the accuracy keeps increasing when choosing bigger feature sets. Although no use of lemmatization or stemming gives the best results in the training phase independent of the feature set size, we do not see this back in the results for the test set. Neither do we see a clear structure in whether stop word removal works for uni-grams.

The use of bi-grams does not improve the accuracy of the model. On top of that, the false positive rate increases a lot. When looking at the tri-grams we see the same.

Table 2: Highest achieved accuracy on test-set

feature set [uni,bi,tri-grams]	NB	SVM	DT	RF
[10,0,0]	0.794	0.807	**0.832**	0.824
[50,0,0]	0.866	0.853	0.881	**0.897**
[100,0,0]	0.872	0.869	0.883	**0.903**
[250,0,0]	0.881	0.883	0.881	**0.922**
[500,0,0]	0.902	0.884	0.894	**0.919**
[750,0,0]	0.908	0.881	0.888	**0.919**
[1000,0,0]	0.911	0.888	0.897	**0.918**
[0,10,0]	**0.826**	0.813	0.824	0.813
[0,50,0]	0.826	0.818	0.832	**0.867**
[0,100,0]	0.850	0.834	0.821	**0.880**
[0,250,0]	0.850	0.840	0.854	**0.892**
[0,500,0]	0.870	0.845	0.858	**0.892**
[0,0,10]	**0.786**	0.777	**0.786**	0.778
[0,0,50]	0.802	0.807	0.804	**0.821**
[0,0,100]	0.804	0.807	0.805	**0.816**
[0,0,250]	0.794	0.815	0.810	**0.826**
[0,0,500]	0.796	0.820	0.815	**0.850**
TF-IDF[10]	0.812	0.788	0.823	**0.840**
TF-IDF[50]	0.864	0.862	0.884	**0.908**
TF-IDF[100]	0.875	0.850	0.897	**0.913**
TF-IDF[250]	0.900	0.813	0.880	**0.918**
TF-IDF[500]	0.905	0.741	0.892	**0.922**
TF-IDF[750]	0.913	-	0.892	**0.924**
TF-IDF[1000]	0.911	-	0.892	**0.924**

Table 3: Performance of the naive Bayes classifier on the test set (selection from tables 16, 17, 18 and 19, appendix F)

feature set [uni,bi,tri-grams]	Lemmatization Stemming	Stop word removal	tn	fp	fn	tp	acc	fpr	fnr
[10, 0, 0]	Lem	Yes	381	23	107	121	0.794	**0.057**	0.469
[50, 0, 0]	Stem	Yes	395	9	76	152	0.866	**0.022**	**0.333**
[100, 0, 0]	Stem	No	395	9	72	156	0.872	0.022	**0.316**
[250, 0, 0]	None	Yes	395	9	66	162	0.881	0.022	**0.289**
[500, 0, 0]	Lem	Yes	401	3	59	169	0.902	**0.007**	**0.259**
[750, 0, 0]	Lem	Yes	399	5	53	175	0.908	0.012	0.232
[1000, 0, 0]	None	No	400	4	52	176	0.911	**0.010**	0.228
[0, 10, 0]	None	Yes	395	9	101	127	0.826	0.022	**0.443**
[0, 50, 0]	None	Yes	400	4	106	122	0.826	0.010	**0.465**
[0, 100, 0]	Lem	No	401	3	92	136	0.850	0.007	**0.404**
[0, 250, 0]	Stem	No	400	4	91	137	0.850	0.010	0.399
[0, 500, 0]	Stem	No	393	11	71	157	0.870	0.027	0.311
[0, 0, 10]	Lem	No	402	2	133	95	0.786	0.005	**0.583**
[0, 0, 50]	Lem	Yes	400	4	121	107	0.802	0.010	**0.531**
[0, 0, 100]	Lem	No	403	1	123	105	0.804	0.002	**0.539**
[0, 0, 250]	Lem	No	404	0	130	98	0.794	**0.000**	0.570
[0, 0, 500]	Lem	No	398	6	123	105	0.796	0.015	0.539
TF-IDF[10]	Lem	Yes	387	17	102	126	0.812	0.042	0.447
TF-IDF[50]	Stem	Yes	395	9	77	151	0.864	0.022	**0.338**
TF-IDF[100]	None	Yes	396	8	71	157	0.875	0.020	**0.311**
TF-IDF[250]	None	Yes	400	4	59	169	0.900	**0.010**	**0.259**
TF-IDF[500]	None	Yes	395	9	51	177	0.905	0.022	0.224
TF-IDF[750]	Stem	Yes	393	11	44	184	0.913	0.027	**0.193**
TF-IDF[1000]	None	No	400	4	52	176	0.911	**0.010**	0.228

Table 4: Performance of the SVM classifier on the test set (selection from tables 20, 21, 22 and 23, appendix F)

feature set [uni,bi,tri-grams]	Lemmatization Stemming	Stop word removal	tn	fp	fn	tp	acc	fpr	fnr
[10, 0, 0]	Lem	Yes	404	0	122	106	0.807	**0.000**	**0.535**
[50, 0, 0]	Lem	Yes	403	1	92	136	0.853	**0.002**	**0.404**
[100, 0, 0]	Lem	Yes	402	2	81	147	0.869	0.005	**0.355**
[250, 0, 0]	Lem	Yes	401	3	71	157	0.883	0.007	**0.311**
[500, 0, 0]	None	No	402	2	71	157	0.884	0.005	**0.311**
[750, 0, 0]	None	Yes	403	1	74	154	0.881	**0.002**	**0.325**
[1000, 0, 0]	None	No	403	1	70	158	0.888	**0.002**	**0.307**
[0, 10, 0]	Lem	Yes	398	6	112	116	0.813	0.015	0.491
[0, 50, 0]	Lem	Yes	396	8	107	121	0.818	0.020	**0.469**
[0, 100, 0]	Lem	Yes	401	3	102	126	0.834	0.007	**0.447**
[0, 250, 0]	Lem	No	402	2	99	129	0.840	0.005	**0.434**
[0, 500, 0]	Lem	No	403	1	97	131	0.845	0.002	**0.425**
[0, 0, 10]	Lem	No	400	4	137	91	0.777	0.010	**0.601**
[0, 0, 50]	Lem	Yes	400	4	118	110	0.807	0.010	**0.518**
[0, 0, 100]	Lem	No	398	6	116	112	0.807	0.015	**0.509**
[0, 0, 250]	Lem	No	403	1	116	112	0.815	0.002	**0.509**
[0, 0, 500]	Lem	No	402	2	112	116	0.820	0.005	**0.491**
TF-IDF[10]	Lem	Yes	401	3	131	97	0.788	**0.007**	0.575
TF-IDF[50]	Stem	Yes	404	0	87	141	0.862	**0.000**	0.382
TF-IDF[100]	None	Yes	404	0	95	133	0.850	**0.000**	**0.417**
TF-IDF[250]	Stem	Yes	404	0	118	110	0.813	**0.000**	**0.518**
TF-IDF[500]	Stem	No	404	0	164	64	0.741	0.000	**0.719**
TF-IDF[500]	None	No	404	0	164	64	0.741	0.000	**0.719**

The use of TF-IDF has a small positive effect on the achieved accuracy compared to uni-grams. Stop word removal has in general a positive effect on TF-IDF. Although that there is no strategy that guarantees the best results, the combination of stemming and stop word removal works generally well.

If we look at the best performing feature set, the maximum achieved accuracy is 0.913, with a false positive rate of 0.027 and a false negative rate of 0.193. This result can be obtained using TF-IDF of size 750, with stemming and stop word removal. This also gives us the lowest possible false negative rate for naive Bayes.

4.2 SVM

The results of the SVM classifier can be found in table 4. The complete tables, 20, 21, 22 and 23, can be found in appendix F.
Looking at the results, we see that the SVM classifier is in most cases the poorest performing classifier, although the results are often not far worse than the naive Bayes and decision tree classifiers.

When we look at the obtained results using uni-grams as features, we see that there is a minimal difference in accuracy between the feature sets with sizes between 250 and 1000. On top of that the False Negative Rate hardly declines. Lemmatization and stop word removal seem to have a positive effect on the performance for small feature sets. However, if the size of the feature sets becomes 500 or larger, it is better not to use lemmatization or stemming.

Looking at the bi-grams and tri-grams, we can see that increasing the size of the feature set has a positive effect. The use of lemmatization has a positive effect on the achieved accuracy as well. However, we can conclude that the use of bi-grams and tri-grams does not increase accuracy, and the False Negative Rate rises significant.

Table 5: Performance of the decision tree classifier on the test set (selection from tables 24, 25, 26 and 27, appendix F)

feature set [uni,bi,tri-grams]	Lemmatization Stemming	Stop word removal	tn	fp	fn	tp	acc	fpr	fnr
[10, 0, 0]	Lem	Yes	381	23	83	145	0.832	**0.057**	**0.364**
[50, 0, 0]	Lem	Yes	394	10	65	163	0.881	**0.025**	**0.285**
[100, 0, 0]	Lem	No	390	14	60	168	0.883	0.035	0.263
[100, 0, 0]	Lem	Yes	394	10	64	164	0.883	**0.025**	0.281
[250, 0, 0]	Stem	Yes	368	36	39	189	0.881	0.089	**0.171**
[500, 0, 0]	None	No	394	10	57	171	0.894	0.025	0.250
[750, 0, 0]	None	Yes	397	7	64	164	0.888	**0.017**	0.281
[1000, 0, 0]	Lem	No	389	15	50	178	0.897	**0.037**	0.219
[0, 10, 0]	Lem	Yes	402	2	109	119	0.824	0.005	**0.478**
[0, 50, 0]	Lem	Yes	394	10	96	132	0.832	0.025	0.421
[0, 100, 0]	Lem	Yes	385	19	94	134	0.821	0.047	0.412
[0, 250, 0]	Stem	Yes	399	5	87	141	0.854	**0.012**	0.382
[0, 250, 0]	None	Yes	395	9	83	145	0.854	0.022	0.364
[0, 500, 0]	Lem	No	378	26	64	164	0.858	0.064	**0.281**
[0, 0, 10]	Lem	No	401	3	132	96	0.786	0.007	**0.579**
[0, 0, 50]	Lem	Yes	397	7	117	111	0.804	0.017	0.513
[0, 0, 100]	Lem	Yes	396	8	115	113	0.805	0.020	0.504
[0, 0, 250]	Lem	No	376	28	92	136	0.810	0.069	**0.404**
[0, 0, 500]	Lem	No	398	6	111	117	0.815	0.015	**0.487**
TF-IDF[10]	Lem	Yes	387	17	95	133	0.823	0.042	0.417
TF-IDF[50]	Lem	Yes	398	6	67	161	0.884	**0.015**	0.294
TF-IDF[100]	Stem	Yes	388	16	49	179	0.897	0.040	**0.215**
TF-IDF[250]	Stem	Yes	380	24	52	176	0.880	0.059	**0.228**
TF-IDF[500]	Lem	Yes	395	9	59	169	0.892	0.022	0.259
TF-IDF[750]	None	Yes	386	18	50	178	0.892	0.045	**0.219**
TF-IDF[1000]	Lem	No	396	8	60	168	0.892	**0.020**	0.263

The use of TF-IDF does not have a positive effect on the SVM classifier. It suffers from over-fitting and training took so long that it caused a time-out. For TF-IDF of size 750 and 1000 it even predicts all items to be negative, so we excluded these from table 4

The best performing feature set is the one with size 1000, which does not use lemmatization or stemming and has no stop word removal applied. The achieved accuracy is 0.888, the false positive rate is 0.005 and the false negative rate is 0.307. Although it is possible to find feature sets with a lower false positive rate, this would have a negative impact on both the accuracy and the false negative rate.

4.3 Decision Tree

In table 5 the results for the decision tree classifier are shown. These were selected from tables, 24, 25, 26 and 27, which can be found in appendix F.

Looking at the results achieved by using uni-grams, we see that decision tree classifier perform reasonably well when using smaller feature sets. An accuracy of 0.881 can be achieved with only 50 features. There is no consistent combination of lemmatization or stemming with stop word removal which gives the best performance. However, here again we see that lemmatization works generally well for smaller feature sets and no lemmatization or stemming at all generally works better for larger feature sets.

If we analyse the results for bi-grams and tri-grams, we see that they do not have a positive impact on the achieved accuracy but do decrease the false negative rate.

Table 6: Performance of the random forest classifier on the test set (selection from tables 28, 29, 30 and 31, appendix F)

feature set [uni,bi,tri-grams]	Lemmatization Stemming	Stop word removal	tn	fp	fn	tp	acc	fpr	fnr
[10, 0, 0]	Lem	Yes	367	37	74	154	0.824	0.092	**0.325**
[50, 0, 0]	Stem	Yes	401	3	62	166	0.897	**0.007**	**0.272**
[100, 0, 0]	Lem	Yes	401	3	58	170	0.903	**0.007**	0.254
[100, 0, 0]	Stem	Yes	400	4	57	171	0.903	0.010	**0.250**
[250, 0, 0]	None	Yes	403	1	48	180	0.922	**0.002**	**0.211**
[500, 0, 0]	None	No	402	2	49	179	0.919	**0.005**	**0.215**
[500, 0, 0]	None	Yes	402	2	49	179	0.919	**0.005**	**0.215**
[750, 0, 0]	Lem	Yes	403	1	50	178	0.919	**0.002**	0.219
[750, 0, 0]	None	No	401	3	48	180	0.919	0.007	**0.211**
[1000, 0, 0]	Lem	Yes	401	3	49	179	0.918	0.007	0.215
[1000, 0, 0]	None	Yes	400	4	48	180	0.918	0.010	**0.211**
[0, 10, 0]	Lem	No	375	29	78	150	0.831	0.072	**0.342**
[0, 50, 0]	Lem	Yes	392	12	72	156	0.867	0.030	**0.316**
[0, 100, 0]	Lem	No	397	7	69	159	0.880	0.017	**0.303**
[0, 250, 0]	Lem	Yes	397	7	61	167	0.892	0.017	**0.268**
[0, 500, 0]	Lem	Yes	400	4	64	164	0.892	0.010	**0.281**
[0, 0, 10]	Lem	Yes	395	9	131	97	0.778	0.022	0.575
[0, 0, 10]	None	No	404	0	140	88	0.778	**0.000**	0.614
[0, 0, 50]	Lem	No	384	20	93	135	0.821	0.050	**0.408**
[0, 0, 100]	Lem	No	377	27	89	139	0.816	0.067	**0.390**
[0, 0, 250]	Lem	No	380	24	86	142	0.826	0.059	**0.377**
[0, 0, 250]	Lem	Yes	383	21	89	139	0.826	0.052	0.390
[0, 0, 500]	Lem	No	387	17	78	150	0.850	0.042	**0.342**
TF-IDF[10]	Lem	Yes	380	24	77	151	0.840	**0.059**	**0.338**
TF-IDF[50]	Stem	Yes	400	4	54	174	0.908	**0.010**	**0.237**
TF-IDF[100]	Stem	Yes	402	2	53	175	0.913	**0.005**	**0.232**
TF-IDF[100]	Lem	Yes	402	2	53	175	0.913	**0.005**	**0.232**
TF-IDF[250]	None	Yes	403	1	51	177	0.918	**0.002**	0.224
TF-IDF[500]	None	No	402	2	47	181	0.922	**0.005**	**0.206**
TF-IDF[750]	None	No	401	3	45	183	0.924	0.007	**0.197**
TF-IDF[1000]	None	No	401	3	45	183	0.924	0.007	**0.197**

The achieved performance of TF-IDF is comparable with the performance of the uni-grams. Here again we see no strict pattern for whether lemmatization or stemming works best. However, in general stop word removal has a positive effect. Surprisingly there is no difference in achieved performance between the feature sets of size 500, 7500 and 1000.
If we look at which feature set performs best, we find that the highest accuracy can be achieved with a feature set of size 1000 with lemmatization and no stop word removal. This gives us an accuracy of 0.897, a false positive rate of 0.037 and a false negative rate of 0.219. The lowest false negative rate can be achieved by choosing a feature set of size 250 with stemming and stop word removal. This gives a false negative rate of 0.171, an accuracy of 0.881 and a false positive rate of 0.089.

4.4 Random Forest

The last tested classifier is the random forest classifier. The results of this classifier can be found in table 6. The complete overview of results can be found in tables, 28, 29, 30 and 31, in appendix F.

Looking at the achieved accuracy for uni-grams, we see that stop word removal works generally well for this classifier. On top of that lemmatization also improves the achieved accuracy in most cases. The results of the classifiers using uni-grams outperform all other classifiers.

If we look at the results for bi-grams and tri-grams we see that lemmatization still helps to improve the performance, and that stop words removal also has a positive effect on bi-grams. On the other hand, this is not the case for most tri-grams. Although the performance of the bi-grams and tri-grams is less good than the performance of the uni-grams, the performance of the bi-grams is still better than the maximum performance of most other classifiers using uni-grams as feature sets.

When using bigger feature sets for TF-IDF, we see that it is better not to use lemmatization or stemming. Stop word removal does not help either. In general TF-IDF achieves a slightly higher accuracy than uni-grams. On top of that the false negative rate is also lower.

The best results using a random forest classifier can be obtained using a TF-IDF feature set of size 750, without lemmatization or stemming and stop word removal applied. This gives an accuracy of 0.924, a false positive rate of 0.007 and a false negative rate of 0.197. No improvement of the false negative rate can be made by choosing another feature set.

4.5 General evaluation

If we look to the general results, we can see that there is no consistency for which text normalisation method works best. Stop word removal however seems to have a positive effect on the achieved accuracy, especially for the smaller feature set up to a size of 250. Besides this we see that for none of the used classifiers the use of bi-grams or tri-grams has a positive effect on the achieved performance. On the other hand, we see that, except for SVM, TF-IDF gives comparable results to uni-grams and can increase the accuracy as well as lower the false negative rate of the classifier.

In general, the random forest classifier outperforms the other methods. We were able to achieve a maximum accuracy of 0.924, combined with a false positive rate of 0.007 and a false negative rate of 0.197. Which was also the minimum false negative rate for random forest classifiers.

To further lower the false negative rate, we considered using other performance measures for cross-validation, namely recall and the f1-score, and only using positive data for creating the Bag-of-Words and n-grams. Although this did have a minimal positive effect on the achieved false negative rate in some cases, the accuracy decreased significantly, so we left those results out of the evaluation and discussion.

In the next section we will discuss the results that were achieved and give some recommendations for further work.

5 Discussion

In this thesis we constructed a dataset of images used in the romance scam and developed a classifier for the recognition of images used by scammers in the romance scam, by using the recurrence of these images in the online environment. In this section the findings are discussed and compared to related works. The limitations of this research are also discussed. Finally, some suggestions for future work will be given. This includes some consideration that should be made before incorporating this classifier into a technical solution against the romance scam, such as an anti-fraud software application.

5.1 Findings

In this section we will discuss the obtained results as presented in section 4. First, we will shortly discuss the used features and the used classifiers. After that we will discuss the achieved performance in section 5.1.3.

5.1.1 Used features

We considered the use of multiple feature sets with uni-grams from size 10 to 1000, bi- and tri-grams of size 10 to 500 and TF-IDF of size 10 to 1000. We also looked at the effects of the use of stemming or lemmatization and stop word removal.

In general, we can conclude that the use of bi-grams or tri-grams does not have a positive effect on

the achieved accuracy. On top of that the false negative rate increases. This was not what we expected as generally bi-grams perform better than uni-grams and tri-grams perform better than bi-grams. Probably bi-and tri-grams alone are not distinctive enough to make a good classification. However, premature results which are not included in this work, show that using a combination of uni-grams, bi-grams and tri-grams can actually improve performance.
On the other hand, we can see that TF-IDF achieves comparable results to uni-grams and does have a positive effect on the achieved performance for the naive Bayes and random forest classifiers.

Usually, choosing a bigger feature set will increase the performance of the classifier. However, this effect stagnates when using larger feature sets for both decision trees and random forest.
There is no general solution for the use of either stemming or lemmatization or the use of stop word removal. However, the use of lemmatization or stemming does not have a positive effect on the performance for feature sets of size 500 and larger in most cases in this research. Whether stop word removal had a positive effect, was dependent on the size of the feature set as well as the use of text normalisation.

5.1.2 Machine learning

We considered 4 machine learning algorithms as classifiers, namely Naive Bayes, Support Vector Machines, Decision trees and Random Forest. It was surprising to see that Naive Bayes classifier outperformed the SVM classifier in most cases. as in other studies this is often the other way around. In general, the Naive Bayes classifier performed reasonably well. On top of that the training time of the Naive Bayes classifier is extremely low, on average around 1 second, compared to multiple minutes for the other classifiers, especially when a bigger feature set was used.

As expected, the random forest classifier achieved the highest accuracy. On top of that, we could notice that increasing of the number of features above 250 did not further improved the accuracy and false negative rate.

5.1.3 Achieved performance

We achieved a maximum accuracy of 92.4% while using a random forest classifier. This means that approximately 1 in every 13 images gets misclassified. The false positive rate is low with a score of 0.7%. However, this comes at the expense of a high false negative rate. The achieved false negative rate is 19.7%, which means that approximately 1 out of every 5 images used by a scammer is not recognised as such.

If we are willing to allow for a classifier with a lower accuracy, a decision tree, might also provide a reasonable solution. This model gives an accuracy of 88.1%, a false positive rate of 8.9% and a false negative rate of 17.1%. In this case 1 in every 8 images gets misclassified and 1 out of every 6 images used by scammers is not recognised as such. Besides these models there were models found with a lower achieved false negative rate. However, these are two of the best while still maintaining a reasonable accuracy.

Although the achieved accuracy is comparable to other studies, the false negative rate is relatively high. In the field of text classification, this might not have a big impact. For the recognition of the romance scam on the other hand, this impact is bigger, as this could become a problem for end-users when this classifier is implemented into a tool. This means that besides improving the accuracy there is also a need for lowering the false negative rate. A more elaborate discussion on the possible effects of this can be found in section 5.4.3. Of course, it should be kept in mind that there will always be a trade-off between the false positive rate and false negative rate.

5.2 Comparison with other works

To our knowledge there is only one other research that focused on developing a technical measure in the form of a classifier for the recognition of the romance scam. In this recent work by Suarez-Tangil et al. [27] full dating profiles were used to train a classifier. By using the full dating profile, they are able to use a more diverse set of features. They use features based on the images of the profile as well as demographic features and text features extracted from the text in the profile. By using a more realistic dataset and more features, they are able to achieve an accuracy of 0,970 with a false negative rate of

0,071, which means that only 1 out of 14 images is not recognised as such. This is already a great improvement compared to the achieved accuracy of 0.924 in this study. However, it should be kept in mind that they use features extracted from a dating profile that are specific for the website from which they obtained the data. This makes their solution less generic if it should be implemented into a tool.

Although there are no other studies that focus on technical measures for the recognition of the romance scam, this study applies techniques from the field of text and web-page categorisation. We already discussed the work done in this field in section 2.3. An overview of the reviewed literature can be found in table 7 in appendix A. We find that this work achieves similar results compared to other text classification studies, but that the results are not as good the results of the web-page classification studies. This can be explained by the extra features extracted from HTML-documents that are usually used in web-page classification.

5.3 Limitations

The biggest limitation in this research were the used normal images, representing these not used by scammers. Although we were able to find a good dataset of images used by scammers, it was hard to find a representative set of images which we assumed not to be used by scammers. As it is ethically undesirable to use images from random social media accounts, we used images which can be found on the internet and are free to use without license for lack of better alternatives. We paid close attention to finding images that look visually similar to the images in the first dataset, meaning that they looked like they could be used as profile pictures. However, we cannot guarantee that these images generate the same data as images used by genuine persons use on their dating profile. In fact, we should consider that the images used in this study are already widely spread throughout the world wide web, and might occur on different types of web sites, not only limited to dating sites. Actual images used by daters might be less widely spread and we would expect to find these only on dating sites and social media platforms.
Some suggestion on how to tackle the problem of the used data are given in section 5.4.

Another limitation of this study is the use of Selenium for the reverse image search and translation of text. If this solution will be implemented into an actual tool, this might become a problem, as it might not be possible to implement Selenium into the tool. Google offers APIs for most of their services, however these are not free in use. As we did not had budget for this, we automated our process of reverse image search with Selenium. One of the drawbacks is that Selenium is relatively slow compared to other methods, such as APIs, which also limited us in the collection of data.
A last limitation of the use of Selenium is that we had to disregard some of the data, as we could not correctly retrieve it. Reasons for this included that websites no longer existed, could not be reached within the time-out time or that Google Translate was not able to translate the text. This also caused a disbalance between positive and negative data, which might be related to the low false positive rate and high false negative rate.

5.4 Future work

The ultimate objective of the achieved work in this thesis is that it can be implemented into a tool, such as an anti-fraud application which can be used by end-users at home. Before such a tool will be implemented, we would suggest to further explore the possibilities and limitations, such as the used dataset, first. In this section we will give some suggestions on future work for further developing the model as well as some advice for implementation into a tool.

5.4.1 Data

One of the limitations of this research is the used dataset. For future work we suggest using other datasets including more representative images for the normal image dataset. For example, the dataset used by Suarez-Tangil et al. [27] would be a good starting point.
To improve the performance of the classifier, the use of extra features could be considered as well. An example of this is the information which can be extracted from the HTML-documents by using techniques from the field of web page classification.

5.4.2 Features and classifiers

Besides improvement of the used data, further research could also consider other features such as a combination of uni-, bi- and tri-grams and word embedding as well as other machine learning algorithms such as ConvNets.
In this research we only looked at uni-, bi- and tri-grams separately. An exploratory experiment showed that combining these can improve the performance of the classifier. Besides using n-grams and TF-IDF, further research can also focus on word embedding. Word embedding techniques such as word2vec have shown promising results in the field of text classification, although we are not sure if they will work on such a specific topic as the romance scam.
When we consider other classifiers, ConvNets are being used more often recently and show promising results in the field of text classification. One of the drawbacks of ConvNets however is the time complexity of training the models.
At last the machine learning algorithms can be combined with a blacklist containing forums and sites which collect images used by scammers, so that if the reverse image search shows a result from a forum or website warning for scammers, a negative advice will be given immediately.
If the model is further improved and has shown to be robust enough, it can be implemented into a tool. The things that should be kept in mind before doing so are discussed in the next subsection.

5.4.3 Implementation in a tool

After further testing and improving the classifier, the model can be used for a tool which an end-user at home can use to see is in contact with a scammer or a genuine person. Such a tool could be part of the PISA project [38], a project which aims at developing a Personal Information Security Assistant. The tool might for example be a browser plug-in which takes an image as input and gives an advice to the end-user. However, when developing such a tool there are some ethical aspects that should be kept in mind, which are stated below.

Before launching such a tool, we should consider the implications for the end-user. The end-user might become less careful when the tool draws the conclusion that the image is not used by a scammer. In the case of a false negative, in other words, the tool says the image is not used by a scammer, but in reality it is used by a scammer, the end-user might become less careful and miss "red flags" as it trusts the tool. Considering the big impact of the scam on victims, both financially as emotional, this risk should be kept as low as possible.
On the other hand, it the case of a false positive, when the tool says the image is used by a scammer, but in reality it is a genuine person, the end-user might end a true relationship. This might not only be a traumatising experience for the end-user who will most likely end the relationship as he or she thinks to be scammed, but also for the person who he or she is dating with. This person not only loses a relationship but can also be accused of being a scammer by the end-user.

It should be kept in mind that there is always a trade-off between the false positive rate and false negative rate. Considering that most users on a dating site are legitimate, a low false positive rate and a higher false negative rate will mean that the end-user will get a correct advice from the tool most of the times, increasing the willingness of people to use the tool. However, if the user blindly trusts the tool, the impact on the user of a wrong advice of the tool will be higher.
A higher false positive rate combined with a low false negative rate will solve this problem. However, if people will get a wrong advice more often, it is more likely that they will ignore the advice of the tool or just not use it at all.

We would advise to conduct a study on the effect of the advice of the classifier on the behaviour of the end-user before implementation in a tool. This should in particular focus on the effect of end-users either recognising or ignoring the "red flags" with or without an either positive or negative advice from the classifier.

The above should be considered when developing a tool which can be used by end-users to see if they are being scammed. One of the solutions to minimise the effects of false positive and false negatives, might be to not strictly answer with a "yes" or a "no", but include "maybe" and "probably" as possible options, or to give a chance instead. On top of that some advice on how to recognise scammers can be given to the end-user.

6 Conclusion

The online romance scam is a scam with both a high emotional and financial impact. As prevention campaigns nor law have been proven as effective countermeasures, there is a need for other solutions, such as a tool which can be used by end-user at home to check if he or she is in contact with a scammer. In this thesis we made a first step by developing a framework for the recognition of images used by scammers in the romance scam using the recurrence of these images in the online environment.

The first contribution of this research is the dataset that was constructed. This dataset consists out of a total of 4154 images. This dataset has both images used by scammers in the online romance scam and normal images of people.
These images were used as input for reverse image search. We showed how reverse image search can be used to give a good representation of the recurrence of images online and that the output of reverse image search engines is suitable to train classifiers.

In this work we compared the performance of naive Bayes, SVM, decision tree and random forest classifiers while using bag-of-words, bi-grams, tri-grams and TF-IDF as features.
The best performance was achieved using TF-IDF and a random forest classifier. We achieved an accuracy of 92.4% in combination with a false positive rate of 0.7% and a false negative rate of 19.7%. A low false positive rate is positive as only a little amount of legitimate people dating online will be labelled as a scammer. On the other hand, the false negative rate is high which might in practice increase the risk of users becoming scammed if they blindly trust this advice.

The objective of this research was to design and evaluate a classifier that recognises images used in online romance scams by looking at their occurrence online. We generated a dataset of images used in the romance scam and showed that the results of reverse image search can be used to train a classifier. The achieved performance is promising and keeps up to obtained results in related fields. However, further research is needed before this technique can be incorporated in end-user tooling, such as an anti-fraud software application. We expect that the performance of the classifier can be increased and in particular the false negative rate can be lowered by amongst others the use of a better dataset of images not used in the online romance scam.

References

[1] Steeds vaker relatie via internet. https://www.cbs.nl/nl-nl/nieuws/2014/25/steeds-vaker-relatie-via-internet. Last accessed: 01-04-2019.

[2] 5 facts about online dating. https://www.pewresearch.org/fact-tank/2016/02/29/5-facts-about-online-dating/. Last accessed: 01-04-2019.

[3] Online dating research: Statistics, scams, pros and cons. https://www.kaspersky.com/blog/online-dating-report/. Last accessed: 01-04-2019.

[4] Michaela Beals, Marguerite DeLiema, and Martha Deevy. Framework for a taxonomy of fraud. *Washington DC: Stanford Longevity Center/FINRA Financial Investor Education Foundation/Fraud Research Center. Retrieved August*, 25:2016, 2015.

[5] Monica T Whitty and Tom Buchanan. The online romance scam: A serious cybercrime. *CyberPsychology, Behavior, and Social Networking*, 15(3):181–183, 2012.

[6] Romance scams: Online imposters break hearts and bank accounts. https://www.fbi.gov/news/stories/romance-scams. Last accessed: 21-01-2019.

[7] Internet Crime Complaintcenter. Internet crime report 2017. https://pdf.ic3.gov/2017_IC3Report.pdf. Last accessed: 21-01-2019.

[8] Karlijn Vernooij. 134 mensen slachtoffer van datingfraude in 2017. https://demonitor.kro-ncrv.nl/artikelen/134-mensen-slachtoffer-van-datingfraude-in-2017, 2018. Last accessed: 21-01-2019.

[9] Monica T Whitty and Tom Buchanan. The online dating romance scam: The psychological impact on victims–both financial and non-financial. *Criminology & Criminal Justice*, 16(2):176–194, 2016.

[10] Aunshul Rege. What's love got to do with it? exploring online dating scams and identity fraud. *International Journal of Cyber Criminology*, 3(2), 2009.

[11] A. Norta, K. Nyman-Metcalf, A. B. Othman, and A. Rull. *"My agent will not let me talk to the general": Software agents as a tool against internet scams*, pages 11–44. The Future of Law and eTechnologies. 2016.

[12] Monica T Whitty. Mass-marketing fraud: a growing concern. *IEEE Security & Privacy*, 13(4):84–87, 2015.

[13] Fraudehelpdesk - over ons. https://www.fraudehelpdesk.nl/over-ons/. Last accessed: 10-10-2019.

[14] Datingfraude: als daten eindigt in een drama. https://www.fraudehelpdesk.nl/thema/datingfraude-als-daten-eindigt-in-een-drama/. Last accessed: 19-11-2019.

[15] Tom Buchanan and Monica T Whitty. The online dating romance scam: causes and consequences of victimhood. *Psychology, Crime & Law*, 20(3):261–283, 2014.

[16] L. A. Pizzato, J. Akehurst, C. Silvestrini, K. Yacef, I. Koprinska, and J. Kay. *The effect of suspicious profiles on people recommenders*, volume 7379 LNCS of *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. 2012. Cited By :3.

[17] UK Fraud Costs Measurement Committee. Annual fraud indicator 2017: Identifying the costs of fraud to the uk. https://www.experian.co.uk/assets/identity-and-fraud/annual-fraud-indicator-report-2017.pdf. Last accessed: 21-01-2019.

[18] Yung-Tsung Hou, Yimeng Chang, Tsuhan Chen, Chi-Sung Laih, and Chia-Mei Chen. Malicious web content detection by machine learning. *Expert Systems with Applications*, 37(1):55–60, 2010.

[19] Alexandros Ntoulas, Marc Najork, Mark Manasse, and Dennis Fetterly. Detecting spam web pages through content analysis. In *Proceedings of the 15th international conference on World Wide Web*, pages 83–92. ACM, 2006.

[20] Yuancheng Li, Rui Xiao, Jingang Feng, and Liujun Zhao. A semi-supervised learning approach for detection of phishing webpages. *Optik*, 124(23):6027–6033, 2013.

[21] Young Sup Hwang, Jin Baek Kwon, Jae Chan Moon, and Seong Je Cho. Classifying malicious web pages by using an adaptive support vector machine. *Journal of Information Processing Systems*, 9(3):395–404, 2013.

[22] Santosh Kumar, Xiaoying Gao, Ian Welch, and Masood Mansoori. A machine learning based web spam filtering approach. In *2016 IEEE 30th International Conference on Advanced Information Networking and Applications (AINA)*, pages 973–980. IEEE, 2016.

[23] Rami M Mohammad, Fadi Thabtah, and Lee McCluskey. Intelligent rule-based phishing websites classification. *IET Information Security*, 8(3):153–160, 2014.

[24] Weiwei Zhuang, Yanfang Ye, Yong Chen, and Tao Li. Ensemble clustering for internet security applications. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(6):1784–1796, 2012.

[25] Ankit Kumar Jain and Brij B Gupta. A novel approach to protect against phishing attacks at client side using auto-updated white-list. *EURASIP Journal on Information Security*, 2016(1):9, 2016.

[26] Brij B Gupta, Aakanksha Tewari, Ankit Kumar Jain, and Dharma P Agrawal. Fighting against phishing attacks: state of the art and future challenges. *Neural Computing and Applications*, 28(12):3629–3654, 2017.

[27] G. Suarez-Tangil, M. Edwards, C. Peersman, G. Stringhini, A. Rashid, and M. Whitty. Automatically dismantling online dating fraud. *IEEE Transactions on Information Forensics and Security*, 2019. cited By 0.

[28] Arnold WM Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta, and Ramesh Jain. Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (12):1349–1380, 2000.

[29] Vipin Tyagi. *Content-Based Image Retrieval: Ideas, Influences, and Current Trends*. Springer, 2018.

[30] Oge Marques. Visual information retrieval: the state of the art. *IT Professional*, 18(4):7–9, 2016.

[31] Myron Flickner, Harpreet Sawhney, Wayne Niblack, Jonathan Ashley, Qian Huang, Byron Dom, Monika Gorkani, Jim Hafner, Denis Lee, Dragutin Petkovic, et al. Query by image and video content: The qbic system. *computer*, 28(9):23–32, 1995.

[32] Best reverse image search engines, apps and its uses. `https://beebom.com/reverse-image-search-engines-apps-uses/`. Last accessed: 01-02-2019.

[33] MM Terras and I Kirton. Where do images of art go once they go online? a reverse image lookup study to assess the dissemination of digitized cultural heritage. *Museums and the Web*, 2013.

[34] Elizabeth Joan Kelly. Reverse image lookup of a small academic library digital collection. *Codex: the Journal of the Louisiana Chapter of the ACRL*, 3(2):80–92, 2015.

[35] Paul Nieuwenhuysen. Image search process in the web using image copy. *Journal of Multimedia Processing and Technologies*, 9(4):124–133, 2018.

[36] Rajni Jindal, Ruchika Malhotra, and Abha Jain. Techniques for text classification: Literature review and current trends. *webology*, 12(2), 2015.

[37] Aytuğ Onan, Serdar Korukoğlu, and Hasan Bulut. Ensemble of keyword extraction methods and classifiers in text classification. *Expert Systems with Applications*, 57:232–247, 2016.

[38] Roeland HP Kegel and Roel J Wieringa. Behavior change support systems for privacy and security. In *BCSS@ PERSUASIVE*, pages 51–55, 2015.

[39] Hwanjo Yu, Jiawei Han, and Kevin Chen-Chuan Chang. Pebl: Web page classification without negative examples. *IEEE Transactions on Knowledge & Data Engineering*, (1):70–81, 2004.

[40] Win Thanda Aung, Yangon Myanmar, and Khin Hay Mar Saw Hla. Random forest classifier for multi-category classification of web pages. In *2009 IEEE Asia-Pacific Services Computing Conference (APSCC)*, pages 372–376. IEEE, 2009.

[41] Xiaoguang Qi and Brian D Davison. Web page classification: Features and algorithms. *ACM computing surveys (CSUR)*, 41(2):12, 2009.

[42] Myungsook Klassen and Nikhila Paturi. Web document classification by keywords using random forests. In *International Conference on Networked Digital Technologies*, pages 256–261. Springer, 2010.

[43] Lam Hong Lee, Chin Heng Wan, Rajprasad Rajkumar, and Dino Isa. An enhanced support vector machine classification framework by using euclidean distance function for text document categorization. *Applied Intelligence*, 37(1):80–99, 2012.

[44] He Youquan, Xie Jianfang, and Xu Cheng. An improved naive bayesian algorithm for web page text classification. In *2011 Eighth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, volume 3, pages 1765–1768. IEEE, 2011.

[45] Eda Baykan, Monika Henzinger, Ludmila Marian, and Ingmar Weber. A comprehensive study of features and algorithms for url-based topic classification. *ACM Transactions on the Web (TWEB)*, 5(3):15, 2011.

[46] Ram B Basnet, Andrew H Sung, and Quingzhong Liu. Feature selection for improved phishing detection. In *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, pages 252–261. Springer, 2012.

[47] Charu C Aggarwal and ChengXiang Zhai. A survey of text classification algorithms. In *Mining text data*, pages 163–222. Springer, 2012.

[48] Anuj Sharma and Shubhamoy Dey. A comparative study of feature selection and machine learning techniques for sentiment analysis. In *Proceedings of the 2012 ACM research in applied computation symposium*, pages 1–7. ACM, 2012.

[49] David Martens and Foster Provost. Explaining data-driven document classifications. 2013.

[50] Rodrigo Moraes, JoãO Francisco Valiati, and Wilson P GaviãO Neto. Document-level sentiment classification: An empirical comparison between svm and ann. *Expert Systems with Applications*, 40(2):621–633, 2013.

[51] Liangxiao Jiang, Zhihua Cai, Harry Zhang, and Dianhong Wang. Naive bayes text classifiers: a locally weighted learning approach. *Journal of Experimental & Theoretical Artificial Intelligence*, 25(2):273–286, 2013.

[52] R Rajalakshmi and Chandrabose Aravindan. Web page classification using n-gram based url features. In *2013 fifth international conference on advanced computing (ICoAC)*, pages 15–21. IEEE, 2013.

[53] Grigori Sidorov, Francisco Velasquez, Efstathios Stamatatos, Alexander Gelbukh, and Liliana Chanona-Hernández. Syntactic n-grams as machine learning features for natural language processing. *Expert Systems with Applications*, 41(3):853–860, 2014.

[54] Vishwanath Bijalwan, Vinay Kumar, Pinki Kumari, and Jordan Pascual. Knn based machine learning approach for text and document mining. *International Journal of Database Theory and Application*, 7(1):61–70, 2014.

[55] Michael Crawford, Taghi M Khoshgoftaar, Joseph D Prusa, Aaron N Richter, and Hamzah Al Najada. Survey of review spam detection using machine learning techniques. *Journal of Big Data*, 2(1):23, 2015.

[56] Joseph Lilleberg, Yun Zhu, and Yanqing Zhang. Support vector machines and word2vec for text classification with semantic features. In *2015 IEEE 14th International Conference on Cognitive Informatics & Cognitive Computing (ICCI* CC)*, pages 136–140. IEEE, 2015.

[57] Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, pages 649–657, 2015.

[58] Guozhong Feng, Jianhua Guo, Bing-Yi Jing, and Tieli Sun. Feature subset selection using naive bayes for text classification. *Pattern Recognition Letters*, 65:109–115, 2015.

[59] Shasha Wang, Liangxiao Jiang, and Chaoqun Li. Adapting naive bayes tree for text classification. *Knowledge and Information Systems*, 44(1):77–89, 2015.

[60] Siwei Lai, Liheng Xu, Kang Liu, and Jun Zhao. Recurrent convolutional neural networks for text classification. In *Twenty-ninth AAAI conference on artificial intelligence*, 2015.

[61] Liangxiao Jiang, Chaoqun Li, Shasha Wang, and Lungan Zhang. Deep feature weighting for naive bayes and its application to text classification. *Engineering Applications of Artificial Intelligence*, 52:26–39, 2016.

[62] Bo Tang, Haibo He, Paul M Baggenstoss, and Steven Kay. A bayesian classification approach using class-specific features for text categorization. *IEEE Transactions on Knowledge and Data Engineering*, 28(6):1602–1606, 2016.

[63] Lungan Zhang, Liangxiao Jiang, Chaoqun Li, and Ganggang Kong. Two feature weighting approaches for naive bayes text classifiers. *Knowledge-Based Systems*, 100:137–144, 2016.

[64] Aytuğ Onan. Classifier and feature set ensembles for web page classification. *Journal of Information Science*, 42(2):150–165, 2016.

[65] Alexis Conneau, Holger Schwenk, Loïc Barrault, and Yann Lecun. Very deep convolutional networks for text classification. *arXiv preprint arXiv:1606.01781*, 2016.

Appendices

A Text classification

Table 7: Overview of reviewed text and web page classification literature of section 2.3

Title	**Year**	**Used Features**	**Used Classifier(s)**	**max. performance**
PEBL: Web page classification without negative examples [39]	2004		PEBL, SVM	-
Detecting spam web pages through content analysis [19]	2006	n-grams, average word length, #words	SVM, C4.5, NN	Acc: 97%
Random Forest Classifier for Multi-category classification of web pages [40]	2009	BoW	RF	Acc: 99%
Web page classification: Features and algorithms [41]	2009	BoW, n-grams, html tags	SVM, kNN, NB	-
Malicious web content detection by machine learning [18]	2010	BoW, length of document, average word length, html tags	NB, SVM, DT, boosted DT	Acc: 96%
Web Document Classification by Keywords Using Random Forest [42]	2010	BoW	RF	Acc: 83%
An enhanced SVM classification framework by using Euclidean distance function for text document categorization [43]	2011	TD-IDF	SVM	Acc: 94%
An improved Naive Bayesian algorithm for Web page text classification [44]	2011	BoW, html tags	NB	-
A Comprehensive study of features and algorithms for URL-based topic classification [45]	2011	token based n-grams	NB, SVM	F1: 84%
Feature selection for improved phishing detection [46]	2012		NB, RF, LR	
A survey of text classification algorithms [47]	2012	BoW	DT, SVM, NB, NN	
A Comparative Study of Feature Selection and Machine Learning Techniques for Sentiment Analysis [48]	2012	BoW	kNN, NB, DT, SVM	Acc: 91%
A semi-supervised learning approach for detection of phishing webpages [20]	2013	Image features, link features	SVM	Acc: 96%
Explaining Data-Driven Document Classifications [49]	2013	BoW	NB, DT, RF, kNN, SVM, NN	
Classifying Malicious Web Pages by Using an Adaptive SVM [21]	2013		SVM, NN	Acc: 94%
Document-level sentiment classification: An empirical comparison between SVM and ANN [50]	2013	BoW, TF-IDF	NB, SVM, NN	
Naive Bayes text classifiers: a locally weighted learning approach [51]	2013		NB	Acc: 87%
Web page classification using n-gram based URL Features [52]	2013	URL based n-grams, TF-IDF	SVM, ME	f1: 78%
Syntactic N-grams as machine learning features for natural language processing [53]	2014	n-grams	SVM, NB, DT	Acc: 100%

Table 7: Overview of reviewed text and web page classification literature of section 2.3

Title	**Year**	**Used Features**	**Used Classifier(s)**	**max. performance**
Techniques for text classification: Literature review and current Trends [36]	2015	BoW	DT, NB, NN, kNN, SVM	
KNN based Machine Learning Approach for Text and Document mining [54]	2015	BoW, TF-IDF	kNN, NB	Acc: 99%
Survey of review spam detection using machine learning techniques [55]	2015	BoW, TF	SVM, NB	
Support vector machines and Word2vec for text classification with semantic features [56]	2015	word2vec, TD-IDF	SVM	Acc: 89%
Character-level convolutional networks for text classification [57]	2015	Characters, BoW, n-grams, TD-IDF	ConvNet	Acc: 98%
Feature subset selection using naive Bayes for text classification [58]	2015		NB	
Adapting naive Bayes tree for text classification [59]	2015	BoW	NB Tree	Acc: 96%
Recurrent Convolutional Neural Networks for Text Classification [60]	2015		ConvNet	Acc: 96%
A Machine Learning Based Web Spam Filtering Approach [22]	2016		SVM	acc: 96%
Deep feature weighting for naive Bayes and its application to text classification [61]	2016	BoW	NB	
Ensemble of keyword extraction methods and classifiers in text classification [37]	2016	BoW, n-grams, TF-IDF	NB, kNN, SVM, RF	94%
A Bayesian classification approach using class-specific features for text categorization [62]	2016	BoW	NB	
Two feature weighting approaches for naive Bayes text classifiers [63]	2016	BoW	NB	
Classifier and feature set ensembles for web page classification [64]	2016		NB, kNN, DT	
Very Deep Convolutional Networks for Text Classification [65]	2017	Characters	ConvNet	

B Image dataset

The following pages contain images that have been used by scammers as profile pictures in the online romance scam:
https://www.scamsurvivors.com/forum/viewforum.php?f=11&sid=48cee71b2e2979331239dbdf8e605dff
http://scamdigger.com/
https://www.male-scammers.com/browse-all-scams-and-frauds.asp
https://www.stop-scammers.com/
http://scamhatersutd.blogspot.com/
https://1sc.org/scam-on-the-net/romance-scam/photos-used-by-scammers/
http://www.delphifaq.com/outside_the_cube_dating_scams_full.htm
https://www.ripandscam.com/
https://www.datingscams.cc/

The following pages contain normal images:
https://diverseui.com/
https://randomuser.me/photos
https://uifaces.co/
https://morguefile.com/photos/morguefile/1/portrait%20people/pop
http://vis-www.cs.umass.edu/lfw/
These images are complemented with stock photos with properly chosen queries such as “portrait man”, “portrait woman” or “profile picture”. Websites from which such photos were extracted are:
https://pixabay.com/
https://www.pexels.com/

C Examples of images

In figure 4, 18 images used in the online romance scam are presented. They were selected from the sites above. In figure 5, 18 images which were selected from the sites containing normal images are presented. They were used as the set for images not used in the online romance scam in the test case.

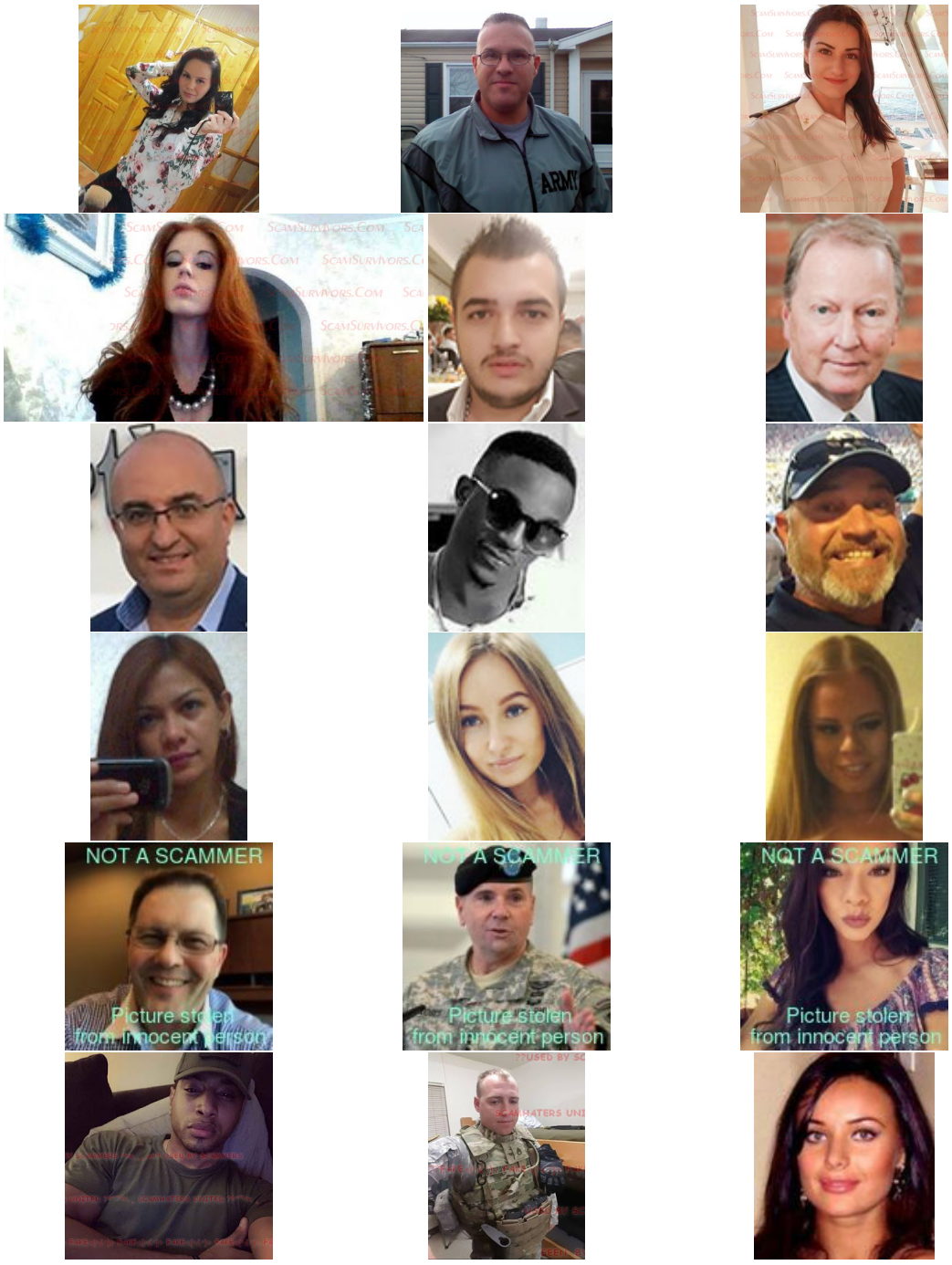


Figure 4: Images used by scammers used in the test case

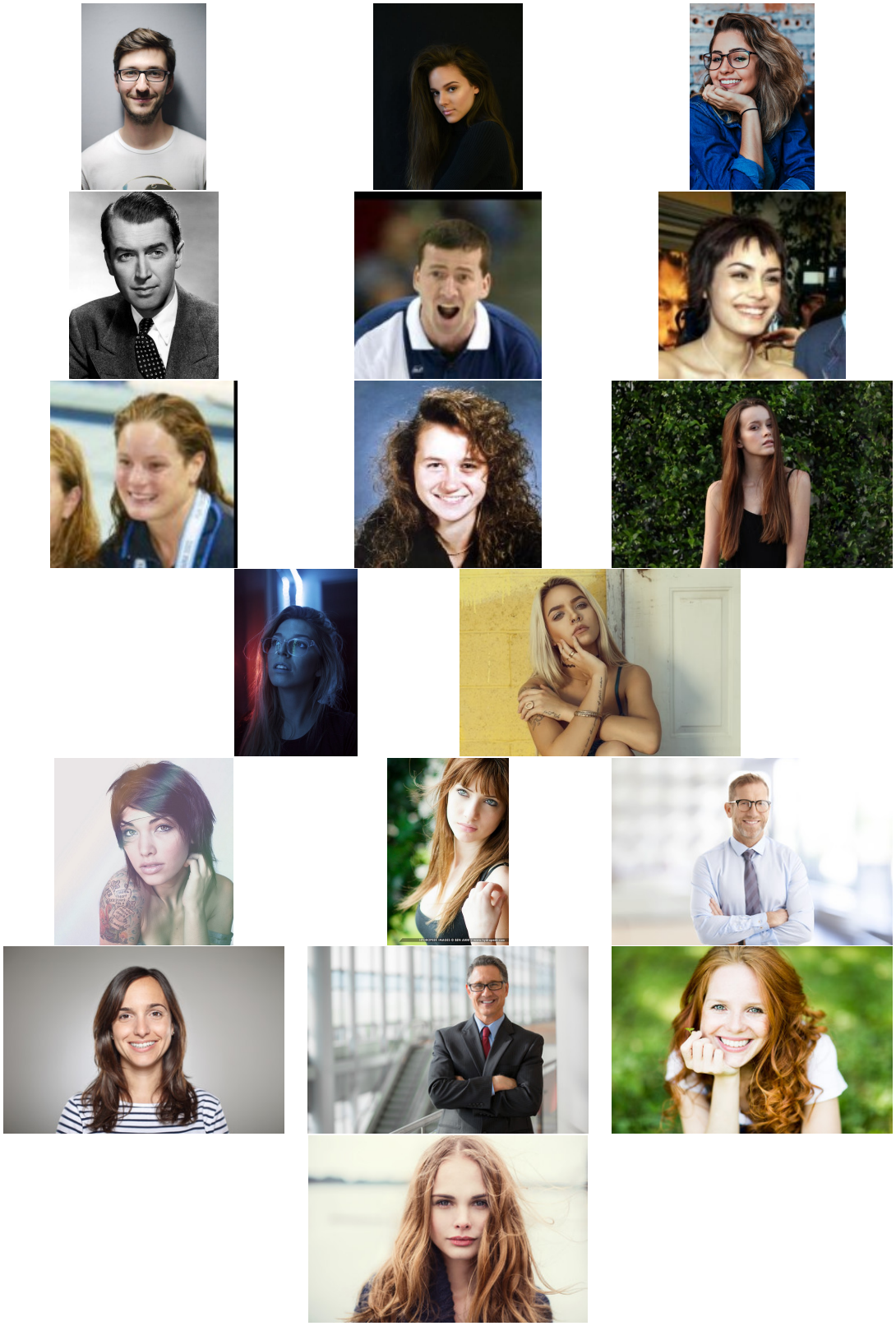
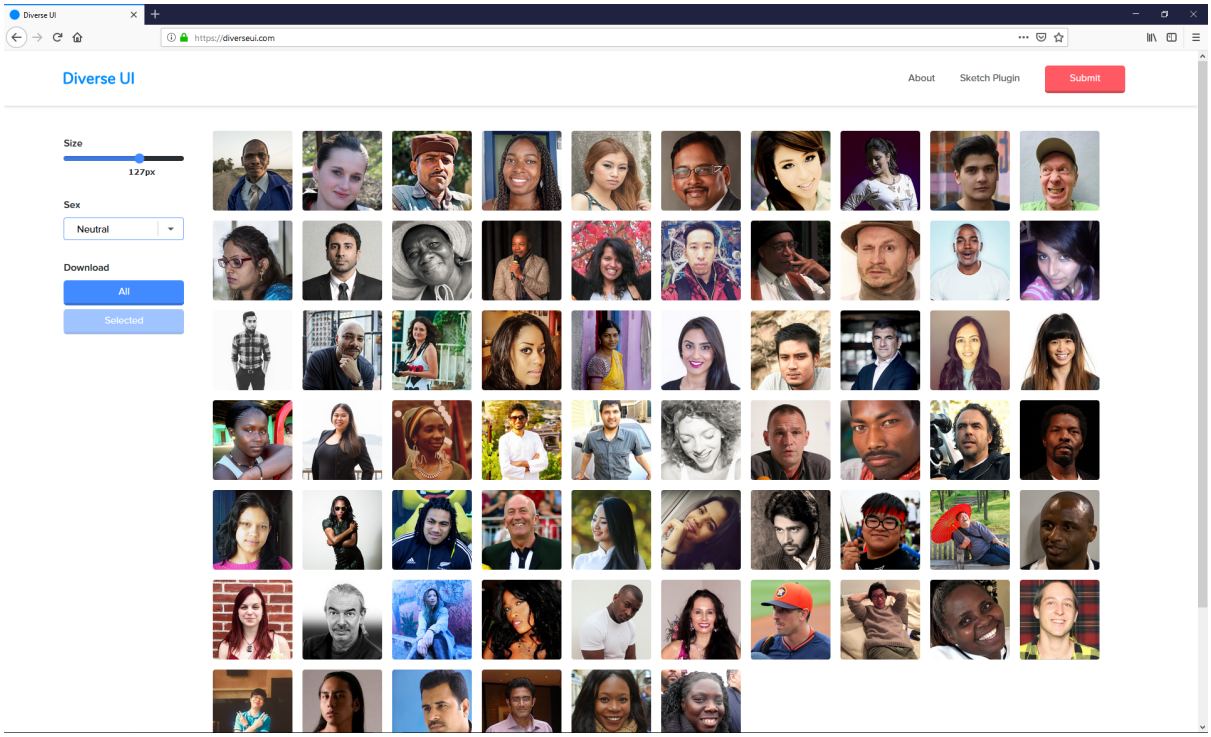


Figure 5: Images not used by scammers used in the test case

D Examples of web pages

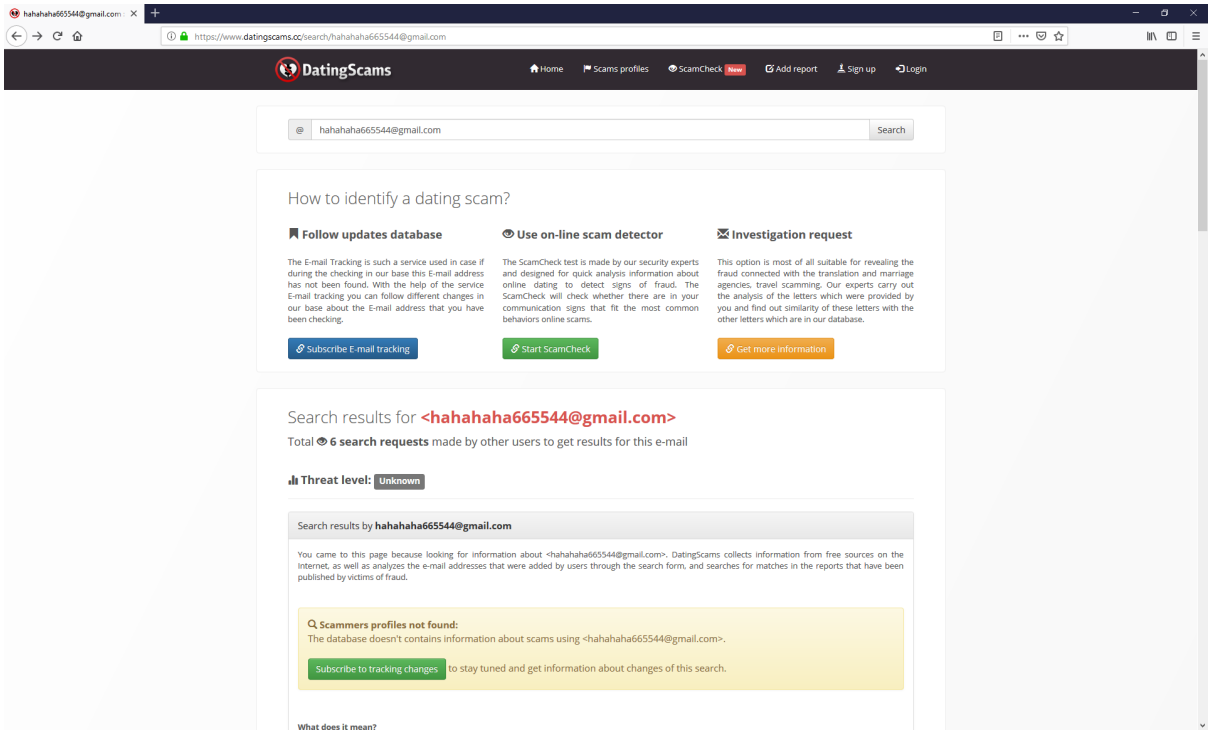
Below some examples of the web pages are given on which the selected images occur. Web pages from which the images are selected are shown in figure 7. Figure **??** shows pages which were given as results by the reverse image search engines after querying the images in the test case.

(a) Online forum warning to look out for scammers.

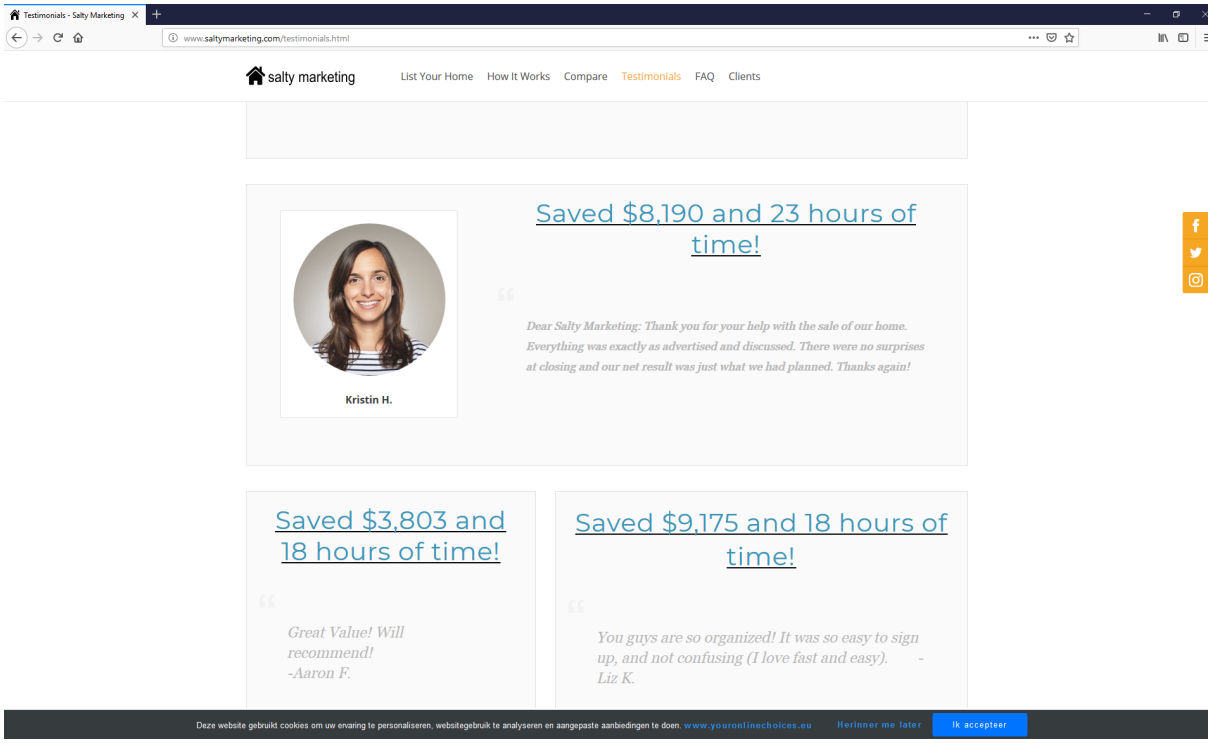


(b) A web page which collect images of people for free use.

Figure 6: Web pages from which images can be extracted.



(a) A web page on which an image used by scammers occurs.



(b) A web page on which one of the normal images occurs.

Figure 7: Web pages given as results by the reverse image search engine.

E Test case data

Below one of the text files as extracted from one of the web sites containing a queried image is shown:

"'In an interview with ', 'on Friday, US Army Lieutenant General Frederick Ben Hodges downplayed the likelihood of a direct armed clash with Russia, but said NATO must improve its logistics to bolster deterrence against Russians.', '"They only respect strength and they despise weakness," Hodges said. "If we look like weŕe not connected, that weŕe not unified, that we donť have capability, and that we cannot move quickly, then I think the potential of a miscalculation is higher."', 'Echoing calls by US President Donald Trump, who has called on Germany to do more to strengthen NATO, Hodges urged Germany to spend more on transportation and missile defense to help it meet its NATO target of 2 percent of economic output.', 'He said the large-scale Saber Guardian war game conducted this summer with thousands of troops from two dozen countries showed progress in the logistics needed to respond to a major military threat.', 'Yet more should be done in order to ease the movement of NATO military hardware and forces across Europe in the event of a real war threat, Hodges said.', '"Thereś not enough rail capacity for US, German, Polish and British forces... or for the NATO VJTF rapid response force," Hodges said. "Wed́ all be competing for the same rail cars."', "Relations between Washington and Moscow have recently plunged to their lowest point since the end of the Cold War in 1991, largely due to the Ukraine crisis. The US and its allies accuse Moscow of sending troops into eastern Ukraine in support of the pro-Russian forces. Moscow has long denied involvement in Ukraine's crisis.", 'Since the Ukraine crisis erupted in November 2013, the United States has accelerated its military build-up on Russia's doorstep. It has even deployed', 'F-35 jets to the East European countries bordering Russia. 0', 'Moscow is wary of NATO's military build-up near its borders. In response, Russia has beefed up its southwestern military capacity, deploying nuclear-capable missiles to its Baltic enclave of Kaliningrad bordering Poland and Lithuania.', 'Ties between the US and Russia further deteriorated when Moscow about two years ago launched an air offensive against Daesh terrorists, many of whom were initially trained by the CIA to fight against the Syrian government.'"

F Tables with results

In this appendix the tables displaying the performance of the different models can be found. These are ordered in the following way:
First the tables displaying the achieved accuracy for the training- and test-set are displayed. The accuracy for the training-set is the average accuracy while using 3-fold cross validation. For the uni-grams the results can be found in respectively tables 8 and 9, for bi-grams in tables 10 and 11, for tri-grams in tables 12 and 13 and for TF-IDF in tables 14 and 15.
After this the tables are sorted by used classifier and show achieved true negative (tn), false positive (fp), false negative (fn), true positive (tp), the accuracy (acc), false positive rate (FPR) and false negative rate (FNR). In the table below can be found which results can be found in which table.

Table 8: Accuracy on training set (using CV=3) for uni-grams

n-grams [uni,bi,tri]	Lemmatization Stemming	Stop word removal	NB	SVM	DT	RF
[10, 0, 0]	Lem	No	0.680	0.650	0.740	0.777
[10, 0, 0]	Lem	Yes	**0.803**	**0.794**	**0.820**	**0.830**
[10, 0, 0]	Stem	No	0.640	0.645	0.733	0.775
[10, 0, 0]	Stem	Yes	0.606	0.778	0.777	0.795
[10, 0, 0]	None	No	0.640	0.645	0.742	0.776
[10, 0, 0]	None	Yes	0.622	0.775	0.784	0.794
[50, 0, 0]	Lem	No	0.797	0.805	0.819	0.872
[50, 0, 0]	Lem	Yes	**0.863**	**0.849**	0.876	**0.904**
[50, 0, 0]	Stem	No	0.817	0.795	0.818	0.862
[50, 0, 0]	Stem	Yes	0.862	0.839	**0.879**	0.900
[50, 0, 0]	None	No	0.811	0.794	0.817	0.864
[50, 0, 0]	None	Yes	0.826	0.796	0.814	0.858
[100, 0, 0]	Lem	No	0.861	**0.849**	**0.870**	0.906
[100, 0, 0]	Lem	Yes	0.855	0.846	0.869	0.907
[100, 0, 0]	Stem	No	0.862	0.838	**0.870**	0.902
[100, 0, 0]	Stem	Yes	**0.866**	0.845	0.868	**0.912**
[100, 0, 0]	None	No	0.819	0.799	0.824	0.880
[100, 0, 0]	None	Yes	0.862	0.841	0.864	0.901
[250, 0, 0]	Lem	No	0.876	0.858	0.869	0.911
[250, 0, 0]	Lem	Yes	0.868	**0.860**	0.858	0.914
[250, 0, 0]	Stem	No	0.869	0.844	0.875	0.911
[250, 0, 0]	Stem	Yes	**0.879**	0.852	0.867	**0.916**
[250, 0, 0]	None	No	0.873	0.846	0.869	0.910
[250, 0, 0]	None	Yes	**0.879**	0.854	**0.878**	0.915
[500, 0, 0]	Lem	No	0.889	0.850	0.874	0.913
[500, 0, 0]	Lem	Yes	0.885	0.856	0.870	**0.919**
[500, 0, 0]	Stem	No	0.879	0.849	**0.882**	0.916
[500, 0, 0]	Stem	Yes	0.876	0.849	0.868	0.918
[500, 0, 0]	None	No	**0.893**	0.855	0.876	0.917
[500, 0, 0]	None	Yes	0.885	**0.862**	0.874	**0.919**
[750, 0, 0]	Lem	No	0.884	0.851	0.881	0.916
[750, 0, 0]	Lem	Yes	0.880	**0.855**	0.875	0.918
[750, 0, 0]	Stem	No	0.875	0.847	0.874	0.916
[750, 0, 0]	Stem	Yes	0.878	0.849	0.874	**0.920**
[750, 0, 0]	None	No	**0.887**	**0.855**	0.872	0.918
[750, 0, 0]	None	Yes	0.886	0.859	**0.877**	**0.920**
[1000, 0, 0]	Lem	No	0.877	0.862	**0.889**	0.916
[1000, 0, 0]	Lem	Yes	0.877	**0.864**	0.876	0.918
[1000, 0, 0]	Stem	No	0.879	0.849	0.879	0.916
[1000, 0, 0]	Stem	Yes	0.879	0.852	0.877	0.919
[1000, 0, 0]	None	No	0.888	0.853	0.879	0.918
[1000, 0, 0]	None	Yes	**0.892**	0.857	0.879	**0.922**

Table 9: Accuracy on test set for uni-grams

n-grams [uni,bi,tri]	Lemmatization Stemming	Stop word removal	NB	SVM	DT	RF
[10, 0, 0]	Lem	No	0.680	0.658	0.758	0.786
[10, 0, 0]	Lem	Yes	**0.794**	**0.807**	**0.832**	**0.824**
[10, 0, 0]	Stem	No	0.620	0.639	0.690	0.763
[10, 0, 0]	Stem	Yes	0.595	0.777	0.758	0.816
[10, 0, 0]	None	No	0.623	0.639	0.729	0.753
[10, 0, 0]	None	Yes	0.592	0.775	0.767	0.815
[50, 0, 0]	Lem	No	0.794	0.826	0.834	0.880
[50, 0, 0]	Lem	Yes	0.853	**0.853**	**0.881**	0.896
[50, 0, 0]	Stem	No	0.807	0.797	0.834	0.873
[50, 0, 0]	Stem	Yes	**0.866**	0.848	0.862	**0.897**
[50, 0, 0]	None	No	0.807	0.797	0.813	0.883
[50, 0, 0]	None	Yes	0.834	0.805	0.826	0.873
[100, 0, 0]	Lem	No	0.866	0.856	**0.883**	0.899
[100, 0, 0]	Lem	Yes	0.858	**0.869**	**0.883**	**0.903**
[100, 0, 0]	Stem	No	**0.872**	0.848	0.873	0.894
[100, 0, 0]	Stem	Yes	0.864	0.861	0.866	**0.903**
[100, 0, 0]	None	No	0.834	0.802	0.843	0.888
[100, 0, 0]	None	Yes	0.866	0.853	0.881	0.899
[250, 0, 0]	Lem	No	0.875	0.872	0.877	0.915
[250, 0, 0]	Lem	Yes	0.873	**0.883**	0.862	0.913
[250, 0, 0]	Stem	No	0.872	0.854	0.872	0.911
[250, 0, 0]	Stem	Yes	0.873	0.872	**0.881**	0.910
[250, 0, 0]	None	No	0.875	0.862	0.853	0.913
[250, 0, 0]	None	Yes	**0.881**	0.870	0.869	**0.922**
[500, 0, 0]	Lem	No	0.896	0.877	0.866	0.913
[500, 0, 0]	Lem	Yes	**0.902**	0.877	0.873	0.911
[500, 0, 0]	Stem	No	0.889	0.870	0.854	0.916
[500, 0, 0]	Stem	Yes	0.892	0.873	0.878	0.915
[500, 0, 0]	None	No	0.891	**0.884**	**0.894**	**0.919**
[500, 0, 0]	None	Yes	0.888	0.881	0.881	**0.919**
[750, 0, 0]	Lem	No	0.903	0.875	0.873	0.915
[750, 0, 0]	Lem	Yes	**0.908**	0.878	0.828	**0.919**
[750, 0, 0]	Stem	No	0.899	0.866	0.848	0.916
[750, 0, 0]	Stem	Yes	0.897	0.867	0.859	0.913
[750, 0, 0]	None	No	0.900	0.875	0.862	**0.919**
[750, 0, 0]	None	Yes	0.899	**0.881**	**0.888**	0.918
[1000, 0, 0]	Lem	No	0.897	0.875	**0.897**	0.916
[1000, 0, 0]	Lem	Yes	0.897	0.872	0.870	**0.918**
[1000, 0, 0]	Stem	No	0.892	0.851	0.891	0.913
[1000, 0, 0]	Stem	Yes	0.899	0.870	0.886	0.911
[1000, 0, 0]	None	No	**0.911**	**0.888**	0.872	0.916
[1000, 0, 0]	None	Yes	0.897	0.878	0.870	**0.918**

Table 10: Accuracy on training set (using CV=3) for bi-grams

n-grams [uni,bi,tri]	Lemmatization Stemming	Stop word removal	NB	SVM	DT	RF
[0, 10, 0]	Lem	No	0.804	0.794	0.812	**0.830**
[0, 10, 0]	Lem	Yes	**0.806**	**0.800**	**0.815**	0.807
[0, 10, 0]	Stem	No	**0.806**	0.796	0.807	0.784
[0, 10, 0]	Stem	Yes	0.802	0.797	0.807	0.807
[0, 10, 0]	None	No	**0.806**	0.796	0.808	0.788
[0, 10, 0]	None	Yes	0.803	0.797	0.807	0.810
[0, 50, 0]	Lem	No	**0.837**	0.803	0.816	**0.862**
[0, 50, 0]	Lem	Yes	0.808	**0.810**	**0.823**	0.860
[0, 50, 0]	Stem	No	0.815	0.797	0.822	0.849
[0, 50, 0]	Stem	Yes	0.822	0.800	0.815	0.803
[0, 50, 0]	None	No	0.817	0.796	**0.823**	0.847
[0, 50, 0]	None	Yes	0.810	0.796	0.811	0.801
[0, 100, 0]	Lem	No	**0.848**	0.823	0.819	0.807
[0, 100, 0]	Lem	Yes	0.829	**0.825**	**0.827**	**0.868**
[0, 100, 0]	Stem	No	0.841	0.810	0.821	0.858
[0, 100, 0]	Stem	Yes	0.789	0.793	0.816	0.813
[0, 100, 0]	None	No	0.829	0.798	0.824	0.856
[0, 100, 0]	None	Yes	0.801	0.797	0.823	0.822
[0, 250, 0]	Lem	No	**0.851**	**0.833**	0.833	0.876
[0, 250, 0]	Lem	Yes	0.830	0.828	0.845	**0.887**
[0, 250, 0]	Stem	No	0.848	0.825	0.825	0.877
[0, 250, 0]	Stem	Yes	0.806	0.809	**0.852**	0.855
[0, 250, 0]	None	No	0.841	0.818	0.831	0.872
[0, 250, 0]	None	Yes	0.806	0.812	0.849	0.856
[0, 500, 0]	Lem	No	0.841	**0.839**	0.842	0.885
[0, 500, 0]	Lem	Yes	0.835	0.827	0.849	**0.889**
[0, 500, 0]	Stem	No	**0.867**	0.827	0.843	0.887
[0, 500, 0]	Stem	Yes	0.798	0.800	**0.851**	0.854
[0, 500, 0]	None	No	0.852	0.828	0.849	0.882
[0, 500, 0]	None	Yes	0.796	0.802	0.852	0.856

Table 11: Accuracy on test set for bi-grams

n-grams [uni,bi,tri]	Lemmatization Stemming	Stop word removal	NB	SVM	DT	RF
[0, 10, 0]	Lem	No	0.812	0.801	0.809	**0.831**
[0, 10, 0]	Lem	Yes	0.815	**0.813**	**0.824**	0.821
[0, 10, 0]	Stem	No	0.810	0.799	0.813	0.809
[0, 10, 0]	Stem	Yes	0.820	0.802	0.815	0.815
[0, 10, 0]	None	No	0.816	0.799	0.821	0.813
[0, 10, 0]	None	Yes	**0.826**	0.801	0.821	0.821
[0, 50, 0]	Lem	No	0.816	0.801	0.826	0.861
[0, 50, 0]	Lem	Yes	0.815	**0.818**	**0.832**	**0.867**
[0, 50, 0]	Stem	No	0.820	0.796	0.802	0.858
[0, 50, 0]	Stem	Yes	0.818	0.810	0.815	0.807
[0, 50, 0]	None	No	0.818	0.793	0.820	0.859
[0, 50, 0]	None	Yes	**0.826**	0.799	0.823	0.804
[0, 100, 0]	Lem	No	**0.850**	0.829	0.805	**0.880**
[0, 100, 0]	Lem	Yes	0.831	**0.834**	**0.821**	0.875
[0, 100, 0]	Stem	No	0.824	0.818	0.786	0.856
[0, 100, 0]	Stem	Yes	0.820	0.805	0.804	0.818
[0, 100, 0]	None	No	0.816	0.809	0.818	0.854
[0, 100, 0]	None	Yes	0.831	0.810	0.831	0.821
[0, 250, 0]	Lem	No	0.848	**0.840**	0.831	0.875
[0, 250, 0]	Lem	Yes	0.843	0.832	0.847	**0.892**
[0, 250, 0]	Stem	No	**0.850**	0.834	0.823	0.886
[0, 250, 0]	Stem	Yes	0.843	0.821	**0.854**	0.851
[0, 250, 0]	None	No	0.839	0.831	0.847	0.878
[0, 250, 0]	None	Yes	0.834	0.824	**0.854**	0.862
[0, 500, 0]	Lem	No	0.839	**0.845**	**0.858**	0.878
[0, 500, 0]	Lem	Yes	0.848	0.843	0.856	**0.892**
[0, 500, 0]	Stem	No	**0.870**	0.837	0.829	0.883
[0, 500, 0]	Stem	Yes	0.843	0.810	0.845	0.847
[0, 500, 0]	None	No	0.872	0.837	0.853	0.888
[0, 500, 0]	None	Yes	0.834	0.812	0.851	0.875

Table 12: Accuracy on training set (using CV=3) for tri-grams

n-grams [uni,bi,tri]	Lemmatization Stemming	Stop word removal	NB	SVM	DT	RF
[0, 0, 10]	Lem	No	**0.781**	**0.780**	**0.782**	0.772
[0, 0, 10]	Lem	Yes	0.780	0.778	0.780	**0.778**
[0, 0, 10]	Stem	No	0.773	0.765	0.773	0.773
[0, 0, 10]	Stem	Yes	0.708	0.694	0.708	0.708
[0, 0, 10]	None	No	0.773	0.765	0.773	0.773
[0, 0, 10]	None	Yes	0.708	0.694	0.708	0.708
[0, 0, 50]	Lem	No	0.788	0.788	0.795	**0.814**
[0, 0, 50]	Lem	Yes	**0.794**	**0.793**	**0.797**	0.799
[0, 0, 50]	Stem	No	0.753	0.774	0.779	0.772
[0, 0, 50]	Stem	Yes	0.728	0.768	0.778	0.777
[0, 0, 50]	None	No	0.780	0.774	0.780	0.779
[0, 0, 50]	None	Yes	0.779	0.771	0.779	0.779
[0, 0, 100]	Lem	No	**0.790**	**0.793**	**0.799**	**0.832**
[0, 0, 100]	Lem	Yes	0.789	0.791	**0.799**	0.802
[0, 0, 100]	Stem	No	0.781	0.776	0.795	0.786
[0, 0, 100]	Stem	Yes	0.781	0.776	0.781	0.774
[0, 0, 100]	None	No	0.781	0.780	0.795	0.790
[0, 0, 100]	None	Yes	0.785	0.776	0.781	0.782
[0, 0, 250]	Lem	No	0.785	**0.803**	**0.810**	**0.841**
[0, 0, 250]	Lem	Yes	**0.810**	0.794	0.807	0.820
[0, 0, 250]	Stem	No	0.781	0.775	0.795	0.786
[0, 0, 250]	Stem	Yes	0.781	0.773	0.781	0.775
[0, 0, 250]	None	No	0.781	0.776	0.797	0.788
[0, 0, 250]	None	Yes	0.776	0.773	0.782	0.782
[0, 0, 500]	Lem	No	**0.798**	**0.808**	**0.816**	**0.856**
[0, 0, 500]	Lem	Yes	0.787	0.797	0.809	0.822
[0, 0, 500]	Stem	No	0.781	0.767	0.793	0.785
[0, 0, 500]	Stem	Yes	0.781	0.772	0.781	0.774
[0, 0, 500]	None	No	0.781	0.765	0.797	0.789
[0, 0, 500]	None	Yes	0.776	0.772	0.782	0.782

Table 13: Accuracy on test set for tri-grams

n-grams [uni,bi,tri]	Lemmatization Stemming	Stop word removal	NB	SVM	DT	RF
[0, 0, 10]	Lem	No	**0.786**	**0.777**	**0.786**	0.759
[0, 0, 10]	Lem	Yes	0.782	0.775	0.777	**0.778**
[0, 0, 10]	Stem	No	0.778	0.766	0.778	0.777
[0, 0, 10]	Stem	Yes	0.712	0.695	0.712	0.712
[0, 0, 10]	None	No	0.778	0.766	0.778	**0.778**
[0, 0, 10]	None	Yes	0.712	0.695	0.712	0.712
[0, 0, 50]	Lem	No	0.797	0.794	0.802	**0.821**
[0, 0, 50]	Lem	Yes	**0.802**	**0.807**	**0.804**	0.793
[0, 0, 50]	Stem	No	0.790	0.780	0.788	0.771
[0, 0, 50]	Stem	Yes	0.783	0.772	0.780	0.782
[0, 0, 50]	None	No	0.788	0.780	0.788	0.788
[0, 0, 50]	None	Yes	0.785	0.774	0.785	0.782
[0, 0, 100]	Lem	No	**0.804**	**0.807**	0.774	**0.816**
[0, 0, 100]	Lem	Yes	0.797	0.804	**0.805**	0.797
[0, 0, 100]	Stem	No	0.793	0.775	0.794	0.791
[0, 0, 100]	Stem	Yes	0.793	0.780	0.793	0.775
[0, 0, 100]	None	No	0.793	0.785	0.796	0.791
[0, 0, 100]	None	Yes	0.778	0.780	0.793	0.797
[0, 0, 250]	Lem	No	**0.794**	**0.815**	**0.810**	**0.826**
[0, 0, 250]	Lem	Yes	0.785	0.807	0.799	**0.826**
[0, 0, 250]	Stem	No	0.793	0.778	0.804	0.788
[0, 0, 250]	Stem	Yes	0.793	0.775	0.790	0.772
[0, 0, 250]	None	No	0.793	0.778	0.801	0.791
[0, 0, 250]	None	Yes	0.778	0.775	0.793	0.796
[0, 0, 500]	Lem	No	**0.796**	**0.820**	**0.815**	**0.850**
[0, 0, 500]	Lem	Yes	0.786	0.816	0.812	0.831
[0, 0, 500]	Stem	No	0.793	0.774	0.804	0.794
[0, 0, 500]	Stem	Yes	0.793	0.772	0.793	0.774
[0, 0, 500]	None	No	0.793	0.774	0.804	0.797
[0, 0, 500]	None	Yes	0.778	0.774	0.793	0.796

Table 14: Accuracy on train set for TF-IDF

TF-IDF	Lemmatization Stemming	Stop word removal	NB	SVM	DT	RF
[10]	Lem	No	0.649	0.767	0.843	0.998
[10]	Lem	Yes	0.803	**0.785**	0.843	0.993
[10]	Stem	No	0.678	0.750	0.805	0.998
[10]	Stem	Yes	0.720	0.770	0.862	0.969
[10]	None	No	0.683	0.747	0.818	0.998
[10]	None	Yes	0.726	0.765	0.841	0.962
[50]	Lem	No	**0.806**	0.782	**0.905**	**0.999**
[50]	Lem	Yes	0.855	0.818	0.900	0.998
[50]	Stem	No	0.818	0.791	0.879	**0.999**
[50]	Stem	Yes	**0.865**	**0.854**	**0.921**	**0.999**
[50]	None	No	0.814	0.793	0.892	**0.999**
[50]	None	Yes	0.838	0.822	0.890	0.998
[100]	Lem	No	0.852	0.799	**0.976**	**0.999**
[100]	Lem	Yes	0.851	0.808	0.909	0.998
[100]	Stem	No	0.853	0.816	0.921	**0.999**
[100]	Stem	Yes	0.871	**0.844**	0.929	**0.999**
[100]	None	No	0.818	0.781	0.896	**0.999**
[100]	None	Yes	**0.866**	0.835	0.887	**0.999**
[250]	Lem	No	0.860	0.776	0.911	**0.999**
[250]	Lem	Yes	0.872	0.782	0.910	0.998
[250]	Stem	No	0.869	0.789	0.908	**0.999**
[250]	Stem	Yes	**0.888**	**0.813**	**0.944**	**0.999**
[250]	None	No	0.862	0.784	0.908	**0.999**
[250]	None	Yes	0.885	0.799	0.923	**0.999**
[500]	Lem	No	0.897	0.734	0.943	**0.999**
[500]	Lem	Yes	0.894	0.713	0.913	0.998
[500]	Stem	No	0.892	0.744	0.900	**0.999**
[500]	Stem	Yes	0.897	0.731	0.933	**0.999**
[500]	None	No	0.893	**0.745**	**0.925**	**0.999**
[500]	None	Yes	**0.902**	0.736	0.906	**0.999**
[750]	Lem	No	0.904	0.639	0.935	**0.999**
[750]	Lem	Yes	0.897	0.639	0.907	**0.999**
[750]	Stem	No	0.894	0.639	0.933	**0.999**
[750]	Stem	Yes	0.906	0.639	0.931	**0.999**
[750]	None	No	0.900	0.639	0.923	**0.999**
[750]	None	Yes	**0.908**	0.639	**0.962**	**0.999**
[1000]	Lem	No	0.903	0.639	0.908	**0.999**
[1000]	Lem	Yes	0.902	0.639	0.924	**0.999**
[1000]	Stem	No	0.895	0.639	0.937	**0.999**
[1000]	Stem	Yes	0.909	0.639	**0.975**	**0.999**
[1000]	None	No	0.910	0.639	0.932	**0.999**
[1000]	None	Yes	**0.911**	0.639	0.928	**0.999**

Table 15: Accuracy on test set for TF-IDF

TF-IDF	Lemmatization Stemming	Stop word removal	NB	SVM	DT	RF
[10]	Lem	No	0.655	0.764	0.756	0.769
[10]	Lem	Yes	**0.812**	**0.788**	**0.823**	**0.84**
[10]	Stem	No	0.679	0.745	0.750	0.759
[10]	Stem	Yes	0.723	0.755	0.748	0.788
[10]	None	No	0.679	0.745	0.745	0.766
[10]	None	Yes	0.710	0.782	0.744	0.783
[50]	Lem	No	0.815	0.778	0.824	0.886
[50]	Lem	Yes	0.854	0.820	**0.884**	0.891
[50]	Stem	No	0.835	0.782	0.816	0.872
[50]	Stem	Yes	**0.864**	**0.862**	0.878	**0.908**
[50]	None	No	0.831	0.783	0.815	0.880
[50]	None	Yes	0.856	0.837	0.839	0.878
[100]	Lem	No	0.858	0.796	0.888	0.907
[100]	Lem	Yes	0.851	0.807	0.883	**0.913**
[100]	Stem	No	0.872	0.813	0.872	0.900
[100]	Stem	Yes	0.873	0.848	**0.897**	**0.913**
[100]	None	No	0.839	0.775	0.820	0.888
[100]	None	Yes	**0.875**	**0.85**	0.867	0.902
[250]	Lem	No	0.873	0.769	0.866	0.911
[250]	Lem	Yes	0.878	0.775	0.875	0.913
[250]	Stem	No	0.880	0.786	0.875	0.916
[250]	Stem	Yes	0.883	**0.813**	**0.880**	0.910
[250]	None	No	0.872	0.777	0.867	0.916
[250]	None	Yes	**0.900**	0.801	0.877	**0.918**
[500]	Lem	No	0.896	0.728	0.877	0.918
[500]	Lem	Yes	0.903	0.714	**0.892**	0.921
[500]	Stem	No	0.889	0.741	0.877	0.918
[500]	Stem	Yes	0.900	0.736	0.853	0.916
[500]	None	No	0.891	**0.741**	0.888	**0.922**
[500]	None	Yes	**0.905**	0.733	0.884	0.916
[750]	Lem	No	0.903	0.639	0.867	0.916
[750]	Lem	Yes	0.907	0.639	0.883	0.916
[750]	Stem	No	0.896	0.639	0.886	0.919
[750]	Stem	Yes	**0.913**	0.639	0.878	0.916
[750]	None	No	0.900	0.639	0.891	**0.924**
[750]	None	Yes	0.905	0.639	**0.892**	0.921
[1000]	Lem	No	0.902	0.639	**0.892**	0.918
[1000]	Lem	Yes	0.910	0.639	0.886	0.911
[1000]	Stem	No	0.884	0.639	0.880	0.916
[1000]	Stem	Yes	0.910	0.639	0.880	0.916
[1000]	None	No	**0.911**	0.639	0.867	**0.924**
[1000]	None	Yes	0.907	0.639	0.878	0.919

Table 16: Performance of the naive Bayes classifier on the test set using uni-grams

n-grams [uni,bi,tri]	Lemmatization Stemming	Stop word removal	tn	fp	fn	tp	acc	fpr	fnr
[10, 0, 0]	Lem	No	282	122	80	148	0.680	0.302	0.351
[10, 0, 0]	Lem	Yes	381	23	107	121	**0.794**	**0.057**	0.469
[10, 0, 0]	Stem	No	257	147	93	135	0.620	0.364	0.408
[10, 0, 0]	Stem	Yes	175	229	27	201	0.595	0.567	**0.118**
[10, 0, 0]	None	No	258	146	92	136	0.623	0.361	0.404
[10, 0, 0]	None	Yes	179	225	33	195	0.592	0.557	0.145
[50, 0, 0]	Lem	No	388	16	114	114	0.794	0.040	0.500
[50, 0, 0]	Lem	Yes	395	9	84	144	0.853	**0.022**	0.368
[50, 0, 0]	Stem	No	377	27	95	133	0.807	0.067	0.417
[50, 0, 0]	Stem	Yes	395	9	76	152	**0.866**	**0.022**	**0.333**
[50, 0, 0]	None	No	382	22	100	128	0.807	0.054	0.439
[50, 0, 0]	None	Yes	394	10	95	133	0.834	0.025	0.417
[100, 0, 0]	Lem	No	399	5	80	148	0.866	**0.012**	0.351
[100, 0, 0]	Lem	Yes	395	9	81	147	0.858	0.022	0.355
[100, 0, 0]	Stem	No	395	9	72	156	**0.872**	0.022	**0.316**
[100, 0, 0]	Stem	Yes	396	8	78	150	0.864	0.020	0.342
[100, 0, 0]	None	No	395	9	96	132	0.834	0.022	0.421
[100, 0, 0]	None	Yes	401	3	82	146	0.866	0.007	0.360
[250, 0, 0]	Lem	No	394	10	69	159	0.875	0.025	0.303
[250, 0, 0]	Lem	Yes	391	13	67	161	0.873	0.032	0.294
[250, 0, 0]	Stem	No	393	11	70	158	0.872	0.027	0.307
[250, 0, 0]	Stem	Yes	396	8	72	156	0.873	**0.020**	0.316
[250, 0, 0]	None	No	395	9	70	158	0.875	0.022	0.307
[250, 0, 0]	None	Yes	395	9	66	162	**0.881**	0.022	**0.289**
[500, 0, 0]	Lem	No	397	7	59	169	0.896	0.017	**0.259**
[500, 0, 0]	Lem	Yes	401	3	59	169	**0.902**	**0.007**	**0.259**
[500, 0, 0]	Stem	No	398	6	64	164	0.889	0.015	0.281
[500, 0, 0]	Stem	Yes	397	7	61	167	0.892	0.017	0.268
[500, 0, 0]	None	No	398	6	63	165	0.891	0.015	0.276
[500, 0, 0]	None	Yes	401	3	68	160	0.888	**0.007**	0.298
[750, 0, 0]	Lem	No	398	6	55	173	0.903	0.015	0.241
[750, 0, 0]	Lem	Yes	399	5	53	175	**0.908**	0.012	0.232
[750, 0, 0]	Stem	No	389	15	49	179	0.899	0.037	**0.215**
[750, 0, 0]	Stem	Yes	389	15	50	178	0.897	0.037	0.219
[750, 0, 0]	None	No	400	4	59	169	0.900	**0.010**	0.259
[750, 0, 0]	None	Yes	390	14	50	178	0.899	0.035	0.219
[1000, 0, 0]	Lem	No	387	17	48	180	0.897	0.042	0.211
[1000, 0, 0]	Lem	Yes	387	17	48	180	0.897	0.042	0.211
[1000, 0, 0]	Stem	No	384	20	48	180	0.892	0.050	0.211
[1000, 0, 0]	Stem	Yes	386	18	46	182	0.899	0.045	**0.202**
[1000, 0, 0]	None	No	400	4	52	176	**0.911**	**0.010**	0.228
[1000, 0, 0]	None	Yes	388	16	49	179	0.897	0.040	0.215

Table 17: Performance of the naive Bayes classifier on the test set using bi-grams

n-grams [uni,bi,tri]	Lemmatization Stemming	Stop word removal	tn	fp	fn	tp	acc	fpr	fnr
[0, 10, 0]	Lem	No	401	3	116	112	0.812	0.007	0.509
[0, 10, 0]	Lem	Yes	403	1	116	112	0.815	**0.002**	0.509
[0, 10, 0]	Stem	No	402	2	118	110	0.810	0.005	0.518
[0, 10, 0]	Stem	Yes	395	9	105	123	0.820	0.022	0.461
[0, 10, 0]	None	No	403	1	115	113	0.816	**0.002**	0.504
[0, 10, 0]	None	Yes	395	9	101	127	**0.826**	0.022	**0.443**
[0, 50, 0]	Lem	No	395	9	107	121	0.816	0.022	0.469
[0, 50, 0]	Lem	Yes	402	2	115	113	0.815	0.005	0.504
[0, 50, 0]	Stem	No	403	1	113	115	0.820	**0.002**	0.496
[0, 50, 0]	Stem	Yes	399	5	110	118	0.818	0.012	0.482
[0, 50, 0]	None	No	402	2	113	115	0.818	0.005	0.496
[0, 50, 0]	None	Yes	400	4	106	122	**0.826**	0.010	**0.465**
[0, 100, 0]	Lem	No	401	3	92	136	**0.850**	0.007	**0.404**
[0, 100, 0]	Lem	Yes	398	6	101	127	0.831	0.015	0.443
[0, 100, 0]	Stem	No	396	8	103	125	0.824	0.020	0.452
[0, 100, 0]	Stem	Yes	399	5	109	119	0.820	0.012	0.478
[0, 100, 0]	None	No	398	6	110	118	0.816	0.015	0.482
[0, 100, 0]	None	Yes	402	2	105	123	0.831	**0.005**	0.461
[0, 250, 0]	Lem	No	398	6	90	138	0.848	0.015	**0.395**
[0, 250, 0]	Lem	Yes	393	11	88	140	0.843	0.027	0.386
[0, 250, 0]	Stem	No	400	4	91	137	**0.850**	0.010	0.399
[0, 250, 0]	Stem	Yes	403	1	98	130	0.843	**0.002**	0.430
[0, 250, 0]	None	No	394	10	92	136	0.839	0.025	0.404
[0, 250, 0]	None	Yes	402	2	103	125	0.834	0.005	0.452
[0, 500, 0]	Lem	No	400	4	98	130	0.839	0.010	0.430
[0, 500, 0]	Lem	Yes	395	9	87	141	0.848	0.022	0.382
[0, 500, 0]	Stem	No	393	11	71	157	**0.870**	0.027	0.311
[0, 500, 0]	Stem	Yes	403	1	98	130	0.843	**0.002**	0.430
[0, 500, 0]	None	No	389	15	66	162	0.872	0.037	**0.289**
[0, 500, 0]	None	Yes	402	2	103	125	0.834	0.005	0.452

Table 18: Performance of the naive Bayes classifier on the test set using tri-grams

n-grams [uni,bi,tri]	Lemmatization Stemming	Stop word removal	tn	fp	fn	tp	acc	fpr	fnr
[0, 0, 10]	Lem	No	402	2	133	95	**0.786**	0.005	**0.583**
[0, 0, 10]	Lem	Yes	402	2	136	92	0.782	0.005	0.596
[0, 0, 10]	Stem	No	404	0	140	88	0.778	**0.000**	0.614
[0, 0, 10]	Stem	Yes	404	0	182	46	0.712	**0.000**	0.798
[0, 0, 10]	None	No	404	0	140	88	0.778	**0.000**	0.614
[0, 0, 10]	None	Yes	404	0	182	46	0.712	**0.000**	0.798
[0, 0, 50]	Lem	No	400	4	124	104	0.797	0.010	0.544
[0, 0, 50]	Lem	Yes	400	4	121	107	**0.802**	0.010	**0.531**
[0, 0, 50]	Stem	No	404	0	133	95	0.790	**0.000**	0.583
[0, 0, 50]	Stem	Yes	404	0	137	91	0.783	**0.000**	0.601
[0, 0, 50]	None	No	404	0	134	94	0.788	**0.000**	0.588
[0, 0, 50]	None	Yes	404	0	136	92	0.785	**0.000**	0.596
[0, 0, 100]	Lem	No	403	1	123	105	**0.804**	0.002	**0.539**
[0, 0, 100]	Lem	Yes	402	2	126	102	0.797	0.005	0.553
[0, 0, 100]	Stem	No	404	0	131	97	0.793	**0.000**	0.575
[0, 0, 100]	Stem	Yes	404	0	131	97	0.793	**0.000**	0.575
[0, 0, 100]	None	No	404	0	131	97	0.793	**0.000**	0.575
[0, 0, 100]	None	Yes	389	15	125	103	0.778	0.037	0.548
[0, 0, 250]	Lem	No	404	0	130	98	**0.794**	**0.000**	0.570
[0, 0, 250]	Lem	Yes	389	15	121	107	0.785	0.037	**0.531**
[0, 0, 250]	Stem	No	404	0	131	97	0.793	**0.000**	0.575
[0, 0, 250]	Stem	Yes	404	0	131	97	0.793	**0.000**	0.575
[0, 0, 250]	None	No	404	0	131	97	0.793	**0.000**	0.575
[0, 0, 250]	None	Yes	389	15	125	103	0.778	0.037	0.548
[0, 0, 500]	Lem	No	398	6	123	105	**0.796**	0.015	0.539
[0, 0, 500]	Lem	Yes	390	14	121	107	0.786	0.035	**0.531**
[0, 0, 500]	Stem	No	404	0	131	97	0.793	**0.000**	0.575
[0, 0, 500]	Stem	Yes	404	0	131	97	0.793	**0.000**	0.575
[0, 0, 500]	None	No	404	0	131	97	0.793	**0.000**	0.575
[0, 0, 500]	None	Yes	389	15	125	103	0.778	0.037	0.548

Table 19: Performance of the naive Bayes classifier on the test set using TF-IDF

TF-IDF	Lemmatization Stemming	Stop word removal	tn	fp	fn	tp	acc	fpr	fnr
[10]	Lem	No	233	171	47	181	0.655	0.423	**0.206**
[10]	Lem	Yes	387	17	102	126	**0.812**	0.042	0.447
[10]	Stem	No	255	149	54	174	0.679	0.369	0.237
[10]	Stem	Yes	299	105	70	158	0.723	**0.260**	0.307
[10]	None	No	254	150	53	175	0.679	0.371	0.232
[10]	None	Yes	291	113	70	158	0.710	0.280	0.307
[50]	Lem	No	395	9	108	120	0.815	0.022	0.474
[50]	Lem	Yes	396	8	84	144	0.854	**0.020**	0.368
[50]	Stem	No	393	11	93	135	0.835	0.027	0.408
[50]	Stem	Yes	395	9	77	151	**0.864**	0.022	**0.338**
[50]	None	No	395	9	98	130	0.831	0.022	0.430
[50]	None	Yes	395	9	82	146	0.856	0.022	0.360
[100]	Lem	No	399	5	85	143	0.858	**0.012**	0.373
[100]	Lem	Yes	394	10	84	144	0.851	0.025	0.368
[100]	Stem	No	399	5	76	152	0.872	**0.012**	0.333
[100]	Stem	Yes	398	6	74	154	0.873	0.015	0.325
[100]	None	No	396	8	94	134	0.839	0.020	0.412
[100]	None	Yes	396	8	71	157	**0.875**	0.020	**0.311**
[250]	Lem	No	396	8	72	156	0.873	0.020	0.316
[250]	Lem	Yes	397	7	70	158	0.878	0.017	0.307
[250]	Stem	No	395	9	67	161	0.880	0.022	0.294
[250]	Stem	Yes	393	11	63	165	0.883	0.027	0.276
[250]	None	No	392	12	69	159	0.872	0.030	0.303
[250]	None	Yes	400	4	59	169	**0.900**	**0.010**	**0.259**
[500]	Lem	No	397	7	59	169	0.896	0.017	0.259
[500]	Lem	Yes	394	10	51	177	0.903	0.025	0.224
[500]	Stem	No	398	6	64	164	0.889	**0.015**	0.281
[500]	Stem	Yes	390	14	49	179	0.900	0.035	**0.215**
[500]	None	No	398	6	63	165	0.891	**0.015**	0.276
[500]	None	Yes	395	9	51	177	**0.905**	0.022	0.224
[750]	Lem	No	398	6	55	173	0.903	0.015	0.241
[750]	Lem	Yes	393	11	48	180	0.907	0.027	0.211
[750]	Stem	No	393	11	55	173	0.896	0.027	0.241
[750]	Stem	Yes	393	11	44	184	**0.913**	0.027	**0.193**
[750]	None	No	400	4	59	169	0.900	**0.010**	0.259
[750]	None	Yes	393	11	49	179	0.905	0.027	0.215
[1000]	Lem	No	394	10	52	176	0.902	0.025	0.228
[1000]	Lem	Yes	392	12	45	183	0.910	0.030	0.197
[1000]	Stem	No	384	20	53	175	0.884	0.050	0.232
[1000]	Stem	Yes	390	14	43	185	0.910	0.035	**0.189**
[1000]	None	No	400	4	52	176	**0.911**	**0.010**	0.228
[1000]	None	Yes	392	12	47	181	0.907	0.030	0.206

Table 20: Performance of the SVM classifier on the test set using uni-grams

n-grams [uni,bi,tri]	Lemmatization Stemming	Stop word removal	tn	fp	fn	tp	acc	fpr	fnr
[10, 0, 0]	Lem	No	401	3	213	15	0.658	0.007	0.934
[10, 0, 0]	Lem	Yes	404	0	122	106	**0.807**	**0.000**	**0.535**
[10, 0, 0]	Stem	No	403	1	227	1	0.639	0.002	0.996
[10, 0, 0]	Stem	Yes	399	5	136	92	0.777	0.012	0.596
[10, 0, 0]	None	No	402	2	226	2	0.639	0.005	0.991
[10, 0, 0]	None	Yes	400	4	138	90	0.775	0.010	0.605
[50, 0, 0]	Lem	No	402	2	108	120	0.826	0.005	0.474
[50, 0, 0]	Lem	Yes	403	1	92	136	**0.853**	**0.002**	**0.404**
[50, 0, 0]	Stem	No	402	2	126	102	0.797	0.005	0.553
[50, 0, 0]	Stem	Yes	403	1	95	133	0.848	**0.002**	0.417
[50, 0, 0]	None	No	402	2	126	102	0.797	0.005	0.553
[50, 0, 0]	None	Yes	403	1	122	106	0.805	**0.002**	0.535
[100, 0, 0]	Lem	No	401	3	88	140	0.856	0.007	0.386
[100, 0, 0]	Lem	Yes	402	2	81	147	**0.869**	0.005	**0.355**
[100, 0, 0]	Stem	No	401	3	93	135	0.848	0.007	0.408
[100, 0, 0]	Stem	Yes	402	2	86	142	0.861	0.005	0.377
[100, 0, 0]	None	No	401	3	122	106	0.802	0.007	0.535
[100, 0, 0]	None	Yes	403	1	92	136	0.853	**0.002**	0.404
[250, 0, 0]	Lem	No	402	2	79	149	0.872	**0.005**	0.346
[250, 0, 0]	Lem	Yes	401	3	71	157	**0.883**	0.007	**0.311**
[250, 0, 0]	Stem	No	402	2	90	138	0.854	**0.005**	0.395
[250, 0, 0]	Stem	Yes	402	2	79	149	0.872	**0.005**	0.346
[250, 0, 0]	None	No	402	2	85	143	0.862	**0.005**	0.373
[250, 0, 0]	None	Yes	402	2	80	148	0.870	**0.005**	0.351
[500, 0, 0]	Lem	No	401	3	75	153	0.877	0.007	0.329
[500, 0, 0]	Lem	Yes	401	3	75	153	0.877	0.007	0.329
[500, 0, 0]	Stem	No	403	1	81	147	0.87	**0.002**	0.355
[500, 0, 0]	Stem	Yes	402	2	78	150	0.873	0.005	0.342
[500, 0, 0]	None	No	402	2	71	157	**0.884**	0.005	**0.311**
[500, 0, 0]	None	Yes	402	2	73	155	0.881	0.005	0.320
[750, 0, 0]	Lem	No	402	2	77	151	0.875	0.005	0.338
[750, 0, 0]	Lem	Yes	401	3	74	154	0.878	0.007	**0.325**
[750, 0, 0]	Stem	No	402	2	83	145	0.866	0.005	0.364
[750, 0, 0]	Stem	Yes	403	1	83	145	0.867	**0.002**	0.364
[750, 0, 0]	None	No	402	2	77	151	0.875	0.005	0.338
[750, 0, 0]	None	Yes	403	1	74	154	**0.881**	**0.002**	**0.325**
[1000, 0, 0]	Lem	No	399	5	74	154	0.875	0.012	0.325
[1000, 0, 0]	Lem	Yes	400	4	77	151	0.872	0.010	0.338
[1000, 0, 0]	Stem	No	392	12	82	146	0.851	0.030	0.360
[1000, 0, 0]	Stem	Yes	403	1	81	147	0.870	**0.002**	0.355
[1000, 0, 0]	None	No	403	1	70	158	**0.888**	**0.002**	**0.307**
[1000, 0, 0]	None	Yes	401	3	74	154	0.878	0.007	0.325

Table 21: Performance of the SVM classifier on the test set using bi-grams

n-grams [uni,bi,tri]	Lemmatization Stemming	Stop word removal	tn	fp	fn	tp	acc	fpr	fnr
[0, 10, 0]	Lem	No	402	2	124	104	0.801	0.005	0.544
[0, 10, 0]	Lem	Yes	398	6	112	116	**0.813**	0.015	0.491
[0, 10, 0]	Stem	No	404	0	127	101	0.799	**0.000**	**0.557**
[0, 10, 0]	Stem	Yes	404	0	125	103	0.802	**0.000**	0.548
[0, 10, 0]	None	No	404	0	127	101	0.799	**0.000**	**0.557**
[0, 10, 0]	None	Yes	404	0	126	102	0.801	**0.000**	0.553
[0, 50, 0]	Lem	No	398	6	120	108	0.801	0.015	0.526
[0, 50, 0]	Lem	Yes	396	8	107	121	**0.818**	0.020	**0.469**
[0, 50, 0]	Stem	No	403	1	128	100	0.796	0.002	0.561
[0, 50, 0]	Stem	Yes	404	0	120	108	0.810	**0.000**	0.526
[0, 50, 0]	None	No	402	2	129	99	0.793	0.005	0.566
[0, 50, 0]	None	Yes	404	0	127	101	0.799	**0.000**	0.557
[0, 100, 0]	Lem	No	403	1	107	121	0.829	0.002	0.469
[0, 100, 0]	Lem	Yes	401	3	102	126	**0.834**	0.007	**0.447**
[0, 100, 0]	Stem	No	402	2	113	115	0.818	0.005	0.496
[0, 100, 0]	Stem	Yes	404	0	123	105	0.805	**0.000**	0.539
[0, 100, 0]	None	No	401	3	118	110	0.809	0.007	0.518
[0, 100, 0]	None	Yes	404	0	120	108	0.810	**0.000**	0.526
[0, 250, 0]	Lem	No	402	2	99	129	**0.840**	0.005	**0.434**
[0, 250, 0]	Lem	Yes	398	6	100	128	0.832	0.015	0.439
[0, 250, 0]	Stem	No	401	3	102	126	0.834	0.007	0.447
[0, 250, 0]	Stem	Yes	404	0	113	115	0.821	**0.000**	0.496
[0, 250, 0]	None	No	401	3	104	124	0.831	0.007	0.456
[0, 250, 0]	None	Yes	404	0	111	117	0.824	**0.000**	0.487
[0, 500, 0]	Lem	No	403	1	97	131	**0.845**	0.002	**0.425**
[0, 500, 0]	Lem	Yes	402	2	97	131	0.843	0.005	**0.425**
[0, 500, 0]	Stem	No	401	3	100	128	0.837	0.007	0.439
[0, 500, 0]	Stem	Yes	404	0	120	108	0.810	**0.000**	0.526
[0, 500, 0]	None	No	401	3	100	128	0.837	0.007	0.439
[0, 500, 0]	None	Yes	404	0	119	109	0.812	**0.000**	0.522

Table 22: Performance of the SVM classifier on the test set using tri-grams

n-grams [uni,bi,tri]	Lemmatization Stemming	Stop word removal	tn	fp	fn	tp	acc	fpr	fnr
[0, 0, 10]	Lem	No	400	4	137	91	**0.777**	0.010	**0.601**
[0, 0, 10]	Lem	Yes	401	3	139	89	0.775	0.007	0.610
[0, 0, 10]	Stem	No	404	0	148	80	0.766	**0.000**	0.649
[0, 0, 10]	Stem	Yes	404	0	193	35	0.695	**0.000**	0.846
[0, 0, 10]	None	No	404	0	148	80	0.766	**0.000**	0.649
[0, 0, 10]	None	Yes	404	0	193	35	0.695	**0.000**	0.846
[0, 0, 50]	Lem	No	399	5	125	103	0.794	0.012	0.548
[0, 0, 50]	Lem	Yes	400	4	118	110	**0.807**	0.010	**0.518**
[0, 0, 50]	Stem	No	404	0	139	89	0.780	**0.000**	0.610
[0, 0, 50]	Stem	Yes	404	0	144	84	0.772	**0.000**	0.632
[0, 0, 50]	None	No	404	0	139	89	0.780	**0.000**	0.610
[0, 0, 50]	None	Yes	404	0	143	85	0.774	**0.000**	0.627
[0, 0, 100]	Lem	No	398	6	116	112	**0.807**	0.015	**0.509**
[0, 0, 100]	Lem	Yes	401	3	121	107	0.804	0.007	0.531
[0, 0, 100]	Stem	No	404	0	142	86	0.775	**0.000**	0.623
[0, 0, 100]	Stem	Yes	404	0	139	89	0.780	**0.000**	0.610
[0, 0, 100]	None	No	403	1	135	93	0.785	0.002	0.592
[0, 0, 100]	None	Yes	404	0	139	89	0.780	**0.000**	0.610
[0, 0, 250]	Lem	No	403	1	116	112	**0.815**	0.002	**0.509**
[0, 0, 250]	Lem	Yes	401	3	119	109	0.807	0.007	0.522
[0, 0, 250]	Stem	No	404	0	140	88	0.778	**0.000**	0.614
[0, 0, 250]	Stem	Yes	404	0	142	86	0.775	**0.000**	0.623
[0, 0, 250]	None	No	404	0	140	88	0.778	**0.000**	0.614
[0, 0, 250]	None	Yes	404	0	142	86	0.775	**0.000**	0.623
[0, 0, 500]	Lem	No	402	2	112	116	**0.820**	0.005	**0.491**
[0, 0, 500]	Lem	Yes	402	2	114	114	0.816	0.005	0.500
[0, 0, 500]	Stem	No	404	0	143	85	0.774	**0.000**	0.627
[0, 0, 500]	Stem	Yes	404	0	144	84	0.772	**0.000**	0.632
[0, 0, 500]	None	No	404	0	143	85	0.774	**0.000**	0.627
[0, 0, 500]	None	Yes	404	0	143	85	0.774	**0.000**	0.627

Table 23: Performance of the SVM classifier on the test set using TF-IDF

TF-IDF	Lemmatization Stemming	Stop word removal	tn	fp	fn	tp	acc	fpr	fnr
[10]	Lem	No	355	49	100	128	0.764	0.121	**0.439**
[10]	Lem	Yes	401	3	131	97	**0.788**	**0.007**	0.575
[10]	Stem	No	366	38	123	105	0.745	0.094	0.539
[10]	Stem	Yes	356	48	107	121	0.755	0.119	0.469
[10]	None	No	368	36	125	103	0.745	0.089	0.548
[10]	None	Yes	375	29	109	119	0.782	0.072	0.478
[50]	Lem	No	402	2	138	90	0.778	0.005	**0.605**
[50]	Lem	Yes	404	0	114	114	0.820	**0.000**	0.500
[50]	Stem	No	401	3	135	93	0.782	0.007	0.592
[50]	Stem	Yes	404	0	87	141	**0.862**	**0.000**	0.382
[50]	None	No	402	2	135	93	0.783	0.005	0.592
[50]	None	Yes	403	1	102	126	0.837	0.002	0.447
[100]	Lem	No	404	0	129	99	0.796	**0.000**	0.566
[100]	Lem	Yes	404	0	122	106	0.807	**0.000**	0.535
[100]	Stem	No	404	0	118	110	0.813	**0.000**	0.518
[100]	Stem	Yes	404	0	96	132	0.848	**0.000**	0.421
[100]	None	No	403	1	141	87	0.775	0.002	0.618
[100]	None	Yes	404	0	95	133	**0.850**	**0.000**	**0.417**
[250]	Lem	No	404	0	146	82	0.769	**0.000**	0.640
[250]	Lem	Yes	404	0	142	86	0.775	**0.000**	0.623
[250]	Stem	No	404	0	135	93	0.786	**0.000**	0.592
[250]	Stem	Yes	404	0	118	110	**0.813**	**0.000**	**0.518**
[250]	None	No	404	0	141	87	0.777	**0.000**	0.618
[250]	None	Yes	404	0	126	102	0.801	**0.000**	0.553
[500]	Lem	No	404	0	172	56	0.728	0.000	0.754
[500]	Lem	Yes	404	0	181	47	0.714	0.000	0.794
[500]	Stem	No	404	0	164	64	**0.741**	0.000	**0.719**
[500]	Stem	Yes	404	0	167	61	0.736	0.000	0.732
[500]	None	No	404	0	164	64	**0.741**	0.000	**0.719**
[500]	None	Yes	404	0	169	59	0.733	0.000	0.741
[750]	Lem	No	404	0	228	0	0.639	0.000	1.000
[750]	Lem	Yes	404	0	228	0	0.639	0.000	1.000
[750]	Stem	No	404	0	228	0	0.639	0.000	1.000
[750]	Stem	Yes	404	0	228	0	0.639	0.000	1.000
[750]	None	No	404	0	228	0	0.639	0.000	1.000
[750]	None	Yes	404	0	228	0	0.639	0.000	1.000
[1000]	Lem	No	404	0	228	0	0.639	0.000	1.000
[1000]	Lem	Yes	404	0	228	0	0.639	0.000	1.000
[1000]	Stem	No	404	0	228	0	0.639	0.000	1.000
[1000]	Stem	Yes	404	0	228	0	0.639	0.000	1.000
[1000]	None	No	404	0	228	0	0.639	0.000	1.000
[1000]	None	Yes	404	0	228	0	0.639	0.000	1.000

Table 24: Performance of the decision tree classifier on the test set using uni-grams

n-grams [uni,bi,tri]	Lemmatization Stemming	Stop word removal	tn	fp	fn	tp	acc	fpr	fnr
[10, 0, 0]	Lem	No	340	64	89	139	0.758	0.158	0.390
[10, 0, 0]	Lem	Yes	381	23	83	145	**0.832**	**0.057**	**0.364**
[10, 0, 0]	Stem	No	293	111	85	143	0.690	0.275	0.373
[10, 0, 0]	Stem	Yes	374	30	123	105	0.758	0.074	0.539
[10, 0, 0]	None	No	354	50	121	107	0.729	0.124	0.531
[10, 0, 0]	None	Yes	368	36	111	117	0.767	0.089	0.487
[50, 0, 0]	Lem	No	391	13	92	136	0.834	0.032	0.404
[50, 0, 0]	Lem	Yes	394	10	65	163	**0.881**	**0.025**	**0.285**
[50, 0, 0]	Stem	No	383	21	84	144	0.834	0.052	0.368
[50, 0, 0]	Stem	Yes	394	10	77	151	0.862	**0.025**	0.338
[50, 0, 0]	None	No	375	29	89	139	0.813	0.072	0.390
[50, 0, 0]	None	Yes	389	15	95	133	0.826	0.037	0.417
[100, 0, 0]	Lem	No	390	14	60	168	**0.883**	0.035	0.263
[100, 0, 0]	Lem	Yes	394	10	64	164	**0.883**	**0.025**	0.281
[100, 0, 0]	Stem	No	380	24	56	172	0.873	0.059	**0.246**
[100, 0, 0]	Stem	Yes	381	23	62	166	0.866	0.057	0.272
[100, 0, 0]	None	No	373	31	68	160	0.843	0.077	0.298
[100, 0, 0]	None	Yes	399	5	70	158	0.881	0.012	0.307
[250, 0, 0]	Lem	No	389	15	63	165	0.877	0.037	0.276
[250, 0, 0]	Lem	Yes	384	20	67	161	0.862	0.050	0.294
[250, 0, 0]	Stem	No	391	13	68	160	0.872	0.032	0.298
[250, 0, 0]	Stem	Yes	368	36	39	189	**0.881**	0.089	**0.171**
[250, 0, 0]	None	No	400	4	89	139	0.853	**0.010**	0.390
[250, 0, 0]	None	Yes	385	19	64	164	0.869	0.047	0.281
[500, 0, 0]	Lem	No	367	37	48	180	0.866	0.092	0.211
[500, 0, 0]	Lem	Yes	376	28	52	176	0.873	0.069	0.228
[500, 0, 0]	Stem	No	393	11	81	147	0.854	0.027	0.355
[500, 0, 0]	Stem	Yes	399	5	72	156	0.878	**0.012**	0.316
[500, 0, 0]	None	No	394	10	57	171	**0.894**	0.025	0.250
[500, 0, 0]	None	Yes	373	31	44	184	0.881	0.077	**0.193**
[750, 0, 0]	Lem	No	384	20	60	168	0.873	0.050	0.263
[750, 0, 0]	Lem	Yes	346	58	51	177	0.828	0.144	0.224
[750, 0, 0]	Stem	No	352	52	44	184	0.848	0.129	**0.193**
[750, 0, 0]	Stem	Yes	377	27	62	166	0.859	0.067	0.272
[750, 0, 0]	None	No	373	31	56	172	0.862	0.077	0.246
[750, 0, 0]	None	Yes	397	7	64	164	**0.888**	**0.017**	0.281
[1000, 0, 0]	Lem	No	389	15	50	178	**0.897**	**0.037**	0.219
[1000, 0, 0]	Lem	Yes	385	19	63	165	0.870	0.047	0.276
[1000, 0, 0]	Stem	No	382	22	47	181	0.891	0.054	0.206
[1000, 0, 0]	Stem	Yes	386	18	54	174	0.886	0.045	0.237
[1000, 0, 0]	None	No	384	20	61	167	0.872	0.050	0.268
[1000, 0, 0]	None	Yes	365	39	43	185	0.870	0.097	**0.189**

Table 25: Performance of the decision tree classifier on the test set using bi-grams

n-grams [uni,bi,tri]	Lemmatization Stemming	Stop word removal	tn	fp	fn	tp	acc	fpr	fnr
[0, 10, 0]	Lem	No	366	38	83	145	0.809	0.094	0.364
[0, 10, 0]	Lem	Yes	402	2	109	119	**0.824**	0.005	**0.478**
[0, 10, 0]	Stem	No	403	1	117	111	0.813	**0.002**	0.513
[0, 10, 0]	Stem	Yes	401	3	114	114	0.815	0.007	0.500
[0, 10, 0]	None	No	401	3	110	118	0.821	0.007	0.482
[0, 10, 0]	None	Yes	401	3	110	118	0.821	0.007	0.482
[0, 50, 0]	Lem	No	399	5	105	123	0.826	**0.012**	0.461
[0, 50, 0]	Lem	Yes	394	10	96	132	**0.832**	0.025	0.421
[0, 50, 0]	Stem	No	356	48	77	151	0.802	0.119	**0.338**
[0, 50, 0]	Stem	Yes	394	10	107	121	0.815	0.025	0.469
[0, 50, 0]	None	No	370	34	80	148	0.820	0.084	0.351
[0, 50, 0]	None	Yes	399	5	107	121	0.823	0.012	0.469
[0, 100, 0]	Lem	No	358	46	77	151	0.805	0.114	**0.338**
[0, 100, 0]	Lem	Yes	385	19	94	134	**0.821**	0.047	0.412
[0, 100, 0]	Stem	No	350	54	81	147	0.786	0.134	0.355
[0, 100, 0]	Stem	Yes	403	1	123	105	0.804	**0.002**	0.539
[0, 100, 0]	None	No	369	35	80	148	0.818	0.087	0.351
[0, 100, 0]	None	Yes	403	1	106	122	0.831	0.002	0.465
[0, 250, 0]	Lem	No	383	21	86	142	0.831	0.052	0.377
[0, 250, 0]	Lem	Yes	389	15	82	146	0.847	0.037	0.360
[0, 250, 0]	Stem	No	384	20	92	136	0.823	0.050	0.404
[0, 250, 0]	Stem	Yes	399	5	87	141	**0.854**	**0.012**	0.382
[0, 250, 0]	None	No	388	16	81	147	0.847	0.040	**0.355**
[0, 250, 0]	None	Yes	395	9	83	145	**0.854**	0.022	0.364
[0, 500, 0]	Lem	No	378	26	64	164	**0.858**	0.064	**0.281**
[0, 500, 0]	Lem	Yes	389	15	76	152	0.856	0.037	0.333
[0, 500, 0]	Stem	No	384	20	88	140	0.829	0.050	0.386
[0, 500, 0]	Stem	Yes	394	10	88	140	0.845	0.025	0.386
[0, 500, 0]	None	No	394	10	83	145	0.853	0.025	0.364
[0, 500, 0]	None	Yes	396	8	86	142	0.851	**0.020**	0.377

Table 26: Performance of the decision tree classifier on the test set using tri-grams

n-grams [uni,bi,tri]	Lemmatization Stemming	Stop word removal	tn	fp	fn	tp	acc	fpr	fnr
[0, 0, 10]	Lem	No	401	3	132	96	**0.786**	0.007	**0.579**
[0, 0, 10]	Lem	Yes	395	9	132	96	0.777	0.022	**0.579**
[0, 0, 10]	Stem	No	404	0	140	88	0.778	**0.000**	0.614
[0, 0, 10]	Stem	Yes	404	0	182	46	0.712	**0.000**	0.798
[0, 0, 10]	None	No	404	0	140	88	0.778	**0.000**	0.614
[0, 0, 10]	None	Yes	404	0	182	46	0.712	**0.000**	0.798
[0, 0, 50]	Lem	No	385	19	106	122	0.802	0.047	**0.465**
[0, 0, 50]	Lem	Yes	397	7	117	111	**0.804**	0.017	0.513
[0, 0, 50]	Stem	No	404	0	134	94	0.788	**0.000**	0.588
[0, 0, 50]	Stem	Yes	404	0	139	89	0.780	**0.000**	0.610
[0, 0, 50]	None	No	404	0	134	94	0.788	**0.000**	0.588
[0, 0, 50]	None	Yes	404	0	136	92	0.785	**0.000**	0.596
[0, 0, 100]	Lem	No	348	56	87	141	0.774	0.139	**0.382**
[0, 0, 100]	Lem	Yes	396	8	115	113	**0.805**	0.020	0.504
[0, 0, 100]	Stem	No	403	1	129	99	0.794	0.002	0.566
[0, 0, 100]	Stem	Yes	404	0	131	97	0.793	**0.000**	0.575
[0, 0, 100]	None	No	389	15	114	114	0.796	0.037	0.500
[0, 0, 100]	None	Yes	404	0	131	97	0.793	**0.000**	0.575
[0, 0, 250]	Lem	No	376	28	92	136	**0.810**	0.069	**0.404**
[0, 0, 250]	Lem	Yes	393	11	116	112	0.799	0.027	0.509
[0, 0, 250]	Stem	No	397	7	117	111	0.804	0.017	0.513
[0, 0, 250]	Stem	Yes	402	2	131	97	0.790	0.005	0.575
[0, 0, 250]	None	No	394	10	116	112	0.801	0.025	0.509
[0, 0, 250]	None	Yes	404	0	131	97	0.793	**0.000**	0.575
[0, 0, 500]	Lem	No	398	6	111	117	**0.815**	0.015	**0.487**
[0, 0, 500]	Lem	Yes	399	5	114	114	0.812	0.012	0.500
[0, 0, 500]	Stem	No	396	8	116	112	0.804	0.020	0.509
[0, 0, 500]	Stem	Yes	404	0	131	97	0.793	**0.000**	0.575
[0, 0, 500]	None	No	399	5	119	109	0.804	0.012	0.522
[0, 0, 500]	None	Yes	404	0	131	97	0.793	**0.000**	0.575

Table 27: Performance of the decision tree classifier on the test set using TF-IDF

TF-IDF	Lemmatization Stemming	Stop word removal	tn	fp	fn	tp	acc	fpr	fnr
[10]	Lem	No	344	60	94	134	0.756	0.149	**0.412**
[10]	Lem	Yes	387	17	95	133	**0.823**	0.042	0.417
[10]	Stem	No	357	47	111	117	0.750	**0.116**	0.487
[10]	Stem	Yes	340	64	95	133	0.748	0.158	0.417
[10]	None	No	353	51	110	118	0.745	0.126	0.482
[10]	None	Yes	352	52	110	118	0.744	0.129	0.482
[50]	Lem	No	367	37	74	154	0.824	0.092	0.325
[50]	Lem	Yes	398	6	67	161	**0.884**	**0.015**	0.294
[50]	Stem	No	381	23	93	135	0.816	0.057	0.408
[50]	Stem	Yes	382	22	55	173	0.878	0.054	**0.241**
[50]	None	No	366	38	79	149	0.815	0.094	0.346
[50]	None	Yes	385	19	83	145	0.839	0.047	0.364
[100]	Lem	No	384	20	51	177	0.888	0.050	0.224
[100]	Lem	Yes	394	10	64	164	0.883	0.025	0.281
[100]	Stem	No	382	22	59	169	0.872	0.054	0.259
[100]	Stem	Yes	388	16	49	179	**0.897**	0.040	**0.215**
[100]	None	No	370	34	80	148	0.820	0.084	0.351
[100]	None	Yes	395	9	75	153	0.867	**0.022**	0.329
[250]	Lem	No	386	18	67	161	0.866	0.045	0.294
[250]	Lem	Yes	393	11	68	160	0.875	**0.027**	0.298
[250]	Stem	No	387	17	62	166	0.875	0.042	0.272
[250]	Stem	Yes	380	24	52	176	**0.880**	0.059	**0.228**
[250]	None	No	390	14	70	158	0.867	0.035	0.307
[250]	None	Yes	386	18	60	168	0.877	0.045	0.263
[500]	Lem	No	380	24	54	174	0.877	0.059	**0.237**
[500]	Lem	Yes	395	9	59	169	**0.892**	0.022	0.259
[500]	Stem	No	397	7	71	157	0.877	**0.017**	0.311
[500]	Stem	Yes	371	33	60	168	0.853	0.082	0.263
[500]	None	No	392	12	59	169	0.888	0.030	0.259
[500]	None	Yes	386	18	55	173	0.884	0.045	0.241
[750]	Lem	No	382	22	62	166	0.867	0.054	0.272
[750]	Lem	Yes	393	11	63	165	0.883	0.027	0.276
[750]	Stem	No	389	15	57	171	0.886	0.037	0.250
[750]	Stem	Yes	388	16	61	167	0.878	0.040	0.268
[750]	None	No	394	10	59	169	0.891	**0.025**	0.259
[750]	None	Yes	386	18	50	178	**0.892**	0.045	**0.219**
[1000]	Lem	No	396	8	60	168	**0.892**	**0.020**	0.263
[1000]	Lem	Yes	385	19	53	175	0.886	0.047	0.232
[1000]	Stem	No	384	20	56	172	0.880	0.050	0.246
[1000]	Stem	Yes	369	35	41	187	0.880	0.087	**0.180**
[1000]	None	No	378	26	58	170	0.867	0.064	0.254
[1000]	None	Yes	386	18	59	169	0.878	0.045	0.259

Table 28: Performance of the random forest classifier on the test set using uni-grams

n-grams [uni,bi,tri]	Lemmatization Stemming	Stop word removal	tn	fp	fn	tp	acc	fpr	fnr
[10, 0, 0]	Lem	No	359	45	90	138	0.786	0.111	0.395
[10, 0, 0]	Lem	Yes	367	37	74	154	**0.824**	0.092	**0.325**
[10, 0, 0]	Stem	No	357	47	103	125	0.763	0.116	0.452
[10, 0, 0]	Stem	Yes	381	23	93	135	0.816	**0.057**	0.408
[10, 0, 0]	None	No	350	54	102	126	0.753	0.134	0.447
[10, 0, 0]	None	Yes	376	28	89	139	0.815	0.069	0.390
[50, 0, 0]	Lem	No	396	8	68	160	0.880	0.020	0.298
[50, 0, 0]	Lem	Yes	400	4	62	166	0.896	0.010	**0.272**
[50, 0, 0]	Stem	No	394	10	70	158	0.873	0.025	0.307
[50, 0, 0]	Stem	Yes	401	3	62	166	**0.897**	**0.007**	**0.272**
[50, 0, 0]	None	No	396	8	66	162	0.883	0.020	0.289
[50, 0, 0]	None	Yes	393	11	69	159	0.873	0.027	0.303
[100, 0, 0]	Lem	No	400	4	60	168	0.899	0.010	0.263
[100, 0, 0]	Lem	Yes	401	3	58	170	**0.903**	**0.007**	0.254
[100, 0, 0]	Stem	No	402	2	65	163	0.894	0.005	0.285
[100, 0, 0]	Stem	Yes	400	4	57	171	**0.903**	0.010	**0.250**
[100, 0, 0]	None	No	398	6	65	163	0.888	0.015	0.285
[100, 0, 0]	None	Yes	400	4	60	168	0.899	0.010	0.263
[250, 0, 0]	Lem	No	402	2	52	176	0.915	0.005	0.228
[250, 0, 0]	Lem	Yes	402	2	53	175	0.913	0.005	0.232
[250, 0, 0]	Stem	No	402	2	54	174	0.911	0.005	0.237
[250, 0, 0]	Stem	Yes	402	2	55	173	0.910	0.005	0.241
[250, 0, 0]	None	No	402	2	53	175	0.913	0.005	0.232
[250, 0, 0]	None	Yes	403	1	48	180	**0.922**	**0.002**	**0.211**
[500, 0, 0]	Lem	No	402	2	53	175	0.913	0.005	0.232
[500, 0, 0]	Lem	Yes	401	3	53	175	0.911	0.007	0.232
[500, 0, 0]	Stem	No	402	2	51	177	0.916	0.005	0.224
[500, 0, 0]	Stem	Yes	402	2	52	176	0.915	0.005	0.228
[500, 0, 0]	None	No	402	2	49	179	**0.919**	**0.005**	**0.215**
[500, 0, 0]	None	Yes	402	2	49	179	**0.919**	**0.005**	**0.215**
[750, 0, 0]	Lem	No	402	2	52	176	0.915	0.005	0.228
[750, 0, 0]	Lem	Yes	403	1	50	178	**0.919**	**0.002**	0.219
[750, 0, 0]	Stem	No	400	4	49	179	0.916	0.010	0.215
[750, 0, 0]	Stem	Yes	402	2	53	175	0.913	0.005	0.232
[750, 0, 0]	None	No	401	3	48	180	**0.919**	0.007	**0.211**
[750, 0, 0]	None	Yes	401	3	49	179	0.918	0.007	0.215
[1000, 0, 0]	Lem	No	400	4	49	179	0.916	0.010	0.215
[1000, 0, 0]	Lem	Yes	401	3	49	179	**0.918**	0.007	0.215
[1000, 0, 0]	Stem	No	401	3	52	176	0.913	0.007	0.228
[1000, 0, 0]	Stem	Yes	402	2	54	174	0.911	**0.005**	0.237
[1000, 0, 0]	None	No	401	3	50	178	0.916	0.007	0.219
[1000, 0, 0]	None	Yes	400	4	48	180	**0.918**	0.010	**0.211**

Table 29: Performance of the random forest classifier on the test set using bi-grams

n-grams [uni,bi,tri]	Lemmatization Stemming	Stop word removal	tn	fp	fn	tp	acc	fpr	fnr
[0, 10, 0]	Lem	No	375	29	78	150	**0.831**	0.072	**0.342**
[0, 10, 0]	Lem	Yes	377	27	86	142	0.821	0.067	0.377
[0, 10, 0]	Stem	No	381	23	98	130	0.809	0.057	0.430
[0, 10, 0]	Stem	Yes	401	3	114	114	0.815	**0.007**	0.500
[0, 10, 0]	None	No	378	26	92	136	0.813	0.064	0.404
[0, 10, 0]	None	Yes	401	3	110	118	0.821	**0.007**	0.482
[0, 50, 0]	Lem	No	395	9	79	149	0.861	0.022	0.346
[0, 50, 0]	Lem	Yes	392	12	72	156	**0.867**	0.030	**0.316**
[0, 50, 0]	Stem	No	393	11	79	149	0.858	0.027	0.346
[0, 50, 0]	Stem	Yes	374	30	92	136	0.807	0.074	0.404
[0, 50, 0]	None	No	396	8	81	147	0.859	**0.020**	0.355
[0, 50, 0]	None	Yes	372	32	92	136	0.804	0.079	0.404
[0, 100, 0]	Lem	No	397	7	69	159	**0.880**	0.017	**0.303**
[0, 100, 0]	Lem	Yes	398	6	73	155	0.875	**0.015**	0.320
[0, 100, 0]	Stem	No	396	8	83	145	0.856	0.020	0.364
[0, 100, 0]	Stem	Yes	374	30	85	143	0.818	0.074	0.373
[0, 100, 0]	None	No	390	14	78	150	0.854	0.035	0.342
[0, 100, 0]	None	Yes	377	27	86	142	0.821	0.067	0.377
[0, 250, 0]	Lem	No	398	6	73	155	0.875	**0.015**	0.320
[0, 250, 0]	Lem	Yes	397	7	61	167	**0.892**	0.017	**0.268**
[0, 250, 0]	Stem	No	398	6	66	162	0.886	**0.015**	0.289
[0, 250, 0]	Stem	Yes	382	22	72	156	0.851	0.054	0.316
[0, 250, 0]	None	No	397	7	70	158	0.878	0.017	0.307
[0, 250, 0]	None	Yes	385	19	68	160	0.862	0.047	0.298
[0, 500, 0]	Lem	No	398	6	71	157	0.878	0.015	0.311
[0, 500, 0]	Lem	Yes	400	4	64	164	**0.892**	0.010	**0.281**
[0, 500, 0]	Stem	No	401	3	71	157	0.883	**0.007**	0.311
[0, 500, 0]	Stem	Yes	381	23	74	154	0.847	0.057	0.325
[0, 500, 0]	None	No	401	3	68	160	0.888	**0.007**	0.298
[0, 500, 0]	None	Yes	390	14	65	163	0.875	0.035	0.285

Table 30: Performance of the random forest classifier on the test set using tri-grams

n-grams [uni,bi,tri]	Lemmatization Stemming	Stop word removal	tn	fp	fn	tp	acc	fpr	fnr
[0, 0, 10]	Lem	No	356	48	104	124	0.759	0.119	**0.456**
[0, 0, 10]	Lem	Yes	395	9	131	97	**0.778**	0.022	0.575
[0, 0, 10]	Stem	No	404	0	141	87	0.777	**0.000**	0.618
[0, 0, 10]	Stem	Yes	404	0	182	46	0.712	**0.000**	0.798
[0, 0, 10]	None	No	404	0	140	88	**0.778**	**0.000**	0.614
[0, 0, 10]	None	Yes	404	0	182	46	0.712	**0.000**	0.798
[0, 0, 50]	Lem	No	384	20	93	135	**0.821**	0.050	**0.408**
[0, 0, 50]	Lem	Yes	381	23	108	120	0.793	0.057	0.474
[0, 0, 50]	Stem	No	390	14	131	97	0.771	0.035	0.575
[0, 0, 50]	Stem	Yes	404	0	138	90	0.782	**0.000**	0.605
[0, 0, 50]	None	No	404	0	134	94	0.788	**0.000**	0.588
[0, 0, 50]	None	Yes	404	0	138	90	0.782	**0.000**	0.605
[0, 0, 100]	Lem	No	377	27	89	139	**0.816**	0.067	**0.390**
[0, 0, 100]	Lem	Yes	382	22	106	122	0.797	0.054	0.465
[0, 0, 100]	Stem	No	378	26	106	122	0.791	0.064	0.465
[0, 0, 100]	Stem	Yes	392	12	130	98	0.775	0.030	0.570
[0, 0, 100]	None	No	384	20	112	116	0.791	0.050	0.491
[0, 0, 100]	None	Yes	403	1	127	101	0.797	**0.002**	0.557
[0, 0, 250]	Lem	No	380	24	86	142	**0.826**	0.059	**0.377**
[0, 0, 250]	Lem	Yes	383	21	89	139	**0.826**	0.052	0.390
[0, 0, 250]	Stem	No	379	25	109	119	0.788	0.062	0.478
[0, 0, 250]	Stem	Yes	390	14	130	98	0.772	0.035	0.570
[0, 0, 250]	None	No	385	19	113	115	0.791	0.047	0.496
[0, 0, 250]	None	Yes	402	2	127	101	0.796	**0.005**	0.557
[0, 0, 500]	Lem	No	387	17	78	150	**0.850**	0.042	**0.342**
[0, 0, 500]	Lem	Yes	385	19	88	140	0.831	0.047	0.386
[0, 0, 500]	Stem	No	381	23	107	121	0.794	0.057	0.469
[0, 0, 500]	Stem	Yes	392	12	131	97	0.774	0.030	0.575
[0, 0, 500]	None	No	388	16	112	116	0.797	0.040	0.491
[0, 0, 500]	None	Yes	402	2	127	101	0.796	**0.005**	0.557

Table 31: Performance of the decision tree classifier on the test set using TF-IDF

TF-IDF	Lemmatization Stemming	Stop word removal	tn	fp	fn	tp	acc	fpr	fnr
[10]	Lem	No	361	43	103	125	0.769	0.106	0.452
[10]	Lem	Yes	380	24	77	151	**0.840**	**0.059**	**0.338**
[10]	Stem	No	356	48	104	124	0.759	0.119	0.456
[10]	Stem	Yes	374	30	104	124	0.788	0.074	0.456
[10]	None	No	355	49	99	129	0.766	0.121	0.434
[10]	None	Yes	372	32	105	123	0.783	0.079	0.461
[50]	Lem	No	392	12	60	168	0.886	0.030	0.263
[50]	Lem	Yes	399	5	64	164	0.891	0.012	0.281
[50]	Stem	No	390	14	67	161	0.872	0.035	0.294
[50]	Stem	Yes	400	4	54	174	**0.908**	**0.010**	**0.237**
[50]	None	No	390	14	62	166	0.880	0.035	0.272
[50]	None	Yes	394	10	67	161	0.878	0.025	0.294
[100]	Lem	No	402	2	57	171	0.907	**0.005**	0.250
[100]	Lem	Yes	402	2	53	175	**0.913**	**0.005**	**0.232**
[100]	Stem	No	401	3	60	168	0.900	0.007	0.263
[100]	Stem	Yes	402	2	53	175	**0.913**	**0.005**	**0.232**
[100]	None	No	391	13	58	170	0.888	0.032	0.254
[100]	None	Yes	401	3	59	169	0.902	0.007	0.259
[250]	Lem	No	401	3	53	175	0.911	0.007	0.232
[250]	Lem	Yes	401	3	52	176	0.913	0.007	0.228
[250]	Stem	No	403	1	52	176	0.916	0.002	0.228
[250]	Stem	Yes	400	4	53	175	0.910	0.010	0.232
[250]	None	No	401	3	50	178	0.916	0.007	**0.219**
[250]	None	Yes	403	1	51	177	**0.918**	**0.002**	0.224
[500]	Lem	No	402	2	50	178	0.918	**0.005**	0.219
[500]	Lem	Yes	402	2	48	180	0.921	**0.005**	0.211
[500]	Stem	No	401	3	49	179	0.918	0.007	0.215
[500]	Stem	Yes	402	2	51	177	0.916	**0.005**	0.224
[500]	None	No	402	2	47	181	**0.922**	**0.005**	**0.206**
[500]	None	Yes	402	2	51	177	0.916	**0.005**	0.224
[750]	Lem	No	403	1	52	176	0.916	**0.002**	0.228
[750]	Lem	Yes	403	1	52	176	0.916	**0.002**	0.228
[750]	Stem	No	401	3	48	180	0.919	0.007	0.211
[750]	Stem	Yes	402	2	51	177	0.916	0.005	0.224
[750]	None	No	401	3	45	183	**0.924**	0.007	**0.197**
[750]	None	Yes	402	2	48	180	0.921	0.005	0.211
[1000]	Lem	No	401	3	49	179	0.918	0.007	0.215
[1000]	Lem	Yes	400	4	52	176	0.911	0.010	0.228
[1000]	Stem	No	402	2	51	177	0.916	**0.005**	0.224
[1000]	Stem	Yes	401	3	50	178	0.916	0.007	0.219
[1000]	None	No	401	3	45	183	**0.924**	0.007	**0.197**
[1000]	None	Yes	401	3	48	180	0.919	0.007	0.211

G Ethical and legal issues

This appendix contains the full consideration on the ethical and legal issues regarding the conducted research. These considerations were made before the start of the research and are copied from the Research Topics report which was made as preparation for this thesis project.

G.1 Ethical Issues

The images used in this research are taken from websites and online forums which warn that these images have been used by scammers. It is likely to assume that the scammers used images of others without getting their consent. By using these images, they become involved in the research as (anonymous) subjects. As the identity of the subjects is not known, they cannot be informed that they are participating in this research. This raises both ethical and legal issues.
As the research involves human beings, a proposal has been submitted by the ethical committee of the EEMCS faculty of the University of Twente[10]. This proposal can be found in appendix G.3. The ethical committee has approved the research under reference number *RP 2019-09*.
The biggest ethical issues and an explanation on why this research is justifiable are explained below:

In research involving human beings it is common to get informed consent of the subjects. However, in this case the images of the subjects are automatically selected from online forums. The identity of the subjects is unknown and not relevant for the scope of the research. If we would like to get consent of these subjects anyway, we should actively try to retrieve their identity. This would first of all be a violation of privacy. It could also cause distress when the subjects are asked to participate in the research for those subjects that are not aware that their image has been used in an online romance scam.
In this research the use of Personal Identifiable Information (PII) will be kept to a minimum and will not be used for training the classifier. This minimises the risk of loss of anonymity.
Considering this, it is reasonable to say that the benefits of this research, helping people from becoming victimised in the online romance scam, out weights the loss of privacy of the subjects.

G.2 Legal Issues

As explained in the last subsection, subjects are involved in the research by the use of their image. This means that personal data is being processed and we should consider if this research is justifiable considering the GDPR. The processing of data has been reported with the DPO team of the University of Twente[11]. Below an explanation is given why the use of the images in this research is justifiable considering the GDPR:

First of all, the principles relating to processing of personal data as described in Article 5 should be obeyed. To make sure that processing of the data is lawful, at least one of the points mentioned in Article 6(1) of the GDPR should apply. In this case processing is necessary for the purpose of preventing fraud. As stated before, both the financial as well as the emotional impact of the romance scam on victims are high. As little is known about which people become victimised and awareness campaigns do not necessarily prevent people from becoming victimised, other ways of prevention are needed. (Software) tools which recognise (signals of) the scam and raise red flags will are suggested to be useful in prevention, but these tools do to our knowledge not yet exist. This research aims to explore techniques useful for such tools.
The processing of the personal data is needed to explore these techniques to prevent people from being victimised in the romance scam. Considering whereas 47, the processing can be considered as lawful based on Article 6(1.f).

Considering whereas 51 we should consider the processing of the selected images in this research as processing of special categories of personal data, as processing by using reverse image search engines does, in some cases, allow the unique identification of a natural person.
As described in Article 9(1) processing of special categories of personal data is prohibited, unless one of the cases as described in Article 9(2) is applicable. In this case the research is done as scientific research in the form of a master thesis project at the University of Twente. Processing of special categories of

[10] `https://www.utwente.nl/en/eemcs/research/ethics/`
[11] `https://www.utwente.nl/en/cyber-safety/privacy/`

personal data is allowed for amongst others scientific research purposes in accordance with Article 89(1). As we process special categories of personal data, it might be needed to do a data protection impact assessment (DPIA), as referred to in Article 35(1), due to the requirements described in Article 35(3.b). A Pre-DPIA form of the University of Twente has been filled out[12]. The responsible Privacy Contact Person (PCP) has decided that a DPIA is not needed in this case.

Articles 14(1-4) describes that a data subject needs to be informed in the case personal data has not been obtained from the data subject, which information needs to be provided and how and in which scope of time this needs to be done.
However, to inform the data subject, we need to know the identity of the data subject. This would require collecting extra information of the data subject purely to be able to identify the data subject, although it is not needed for the aim of this research. Considering whereas 57 and Article 11 we are not obliged to acquire this additional information.
Besides, Article 15(5.b) states that Articles 14(1-4) shall not apply for scientific research purposes, subject to the conditions and safeguards referred to in Article 89(1) if provision of this information proves impossible or would involve a disproportionate effort. As stated above to inform data subjects, we would have to collect extra information about the subject. Collecting this extra information would include (trying to) identify the data subject and retrieving contact details. This would be either impossible or would involve disproportionate effort compared to the minimal use of the image in which the data subject is displayed.

As stated above, processing of personal data for the purpose our research is lawful regarding the GDPR if Article 89(1) is obeyed. This implies that appropriate safeguards need to be in place to ensure respect for the principle of data minimisation. Those measures should include pseudonymisation from the point where the purposes of the research can be fulfilled in that manner. This can for example include extracting features from text and URLs, such that these are no longer traceable to their source. As long as the data is not anonymous, it should be stored in a safe way when it is not used. Looking at how vulnerable the data is, an encrypted USB-stick should be an appropriate safeguard.
Considering the above, this research is legally justifiable. However, it should be kept in mind that this justification is only applicable for the purposes of this (scientific) research. If one would like to use the results of this research for development of a (commercial) tool, the justification of lawfulness for this research does no longer apply and one should consider the lawfulness regarding the GDPR for the development and use of such a tool.

G.3 Proposal Ethics Committee

The proposal for the research as submitted to the Ethics Committee can be found on the next pages. It has been approved under reference number *RP 2019-09*.

[12] `https://www.utwente.nl/en/cyber-safety/privacy/pre_dpia_form/`

Faculty of Electrical Engineering,
Mathematics and Computer Science

Appendix 6. Checklist for submitting a research proposal to the Ethics Committee
(See Chapter 3)
Checklist for the principal researcher when submitting a request to the EC or the EC member for an assessment of the ethical permissibility of a research proposal

1. General

1. Title of the project: ***A classifier for recognition of images used in the (online) romance scam (Final project, master computer science, 192199978)***
2. Principal researcher (with doctoral research also a professor): ***Koen de Jong***
3. Researchers/research assistants (PhD students, students etc. where known): ***Dona Bucur (as chairman of the graduation committee), Roeland Kegel (as part of the graduation committee)***
4. Department responsible for the research: ***Computer Science***
5. Location where research will be conducted: ***University of Twente***
6. Short description of the project (about 100 words): ***In this research I will try to build a classifier to recognise images used in the (online) romance scam (also known as catfishing). To do so, images that were used by scammers and are published on (online) forums will be reverse image searched. The URL of the image will be given as query to the reverse image search engine, so that downloading and uploading of the image is not needed. Information retrieved from web page, which are given as result by the reverse image search engine, will be used to train the classifier. The people who uploaded the images to the forums are usually the victims of the online romance scam. This implicates that they are most likely not the rightful owner of the image nor the person in the image.***
7. Expected duration of the project and research period: ***5 to 6 months***
8. Number of experimental subjects: ***Not yet known***
9. EC member of the department (if available): ***Not applicable***

2. Questions about fulfilled general requirements and conditions

1. Has this research or similar research by the department been previously submitted to the EC?
   ☐ Yes,
   ☒ **No**
   If yes, what was the number allocated to it by the EC?
   Explanatory notes:

2. Is the research proposal to be considered as medical research (Also see Appendix 4)
   ☐ Yes
   ☒ **No**
   ☐ Uncertain
   Explanatory notes:

3. Are adult, competent subjects selected? (§3.2)
   ☐ Yes, indicate in which of the ways named in the general requirements and conditions this is so
   ☐ No, explain
   ☒ **Uncertain, explain why**
   Explanatory notes: ***The participants are not aware of being subjects in an experiment. Their images are automatically from online forums. The identity of the subjects is unknown and not relevant for the scope of the research. It is not desirable nor the goal of the research to retrieve the identity of subjects.***

4. Are the subjects completely free to participate in the research, and to withdraw from participation whenever they wish and for whatever reason? (§3.2)
   ☐ Yes
   ☒ **No, explain why not**
   ☐ Uncertain, explain why
   Explanatory notes: ***The participants are not aware of being subjects in an experiment. Their images are selected through search activities on the internet.***

5. In the event that it may be necessary to screen experimental subjects in order to reduce the risks of adverse effects of the research: Will the subjects be screened? (§3.4)
   ☒ **Screening is not necessary, explain why not**
   ☐ Yes, explain how
   ☐ No, explain why not
   ☐ Uncertain, explain why
   Explanatory notes: ***The identity of the subjects is not known. Retrieving the identity for the goal of screening would be unethical.***

6. Does the method used allow for the possibility of making an accidental diagnostic finding which the experimental subject should be informed about? (§3.6 and Appendix 4)
   ☒ **No, the method does not allow for this possibility**
   ☐ Yes, and the subject has given signed assent for the method to be used
   ☐ Yes, but the subject has not given signed assent for the method to be used
   ☐ Uncertain, explain why
   Explanatory notes:

7. Are subjects briefed before participation and do they sign an informed consent beforehand in accordance with the general conditions? (§3.2, §3.3, §3.7, §3.8)
   ☐ Yes, attach the information brochure and the form to be signed
   ☒ **No, explain why not**
   ☐ Uncertain, explain why
   Explanatory notes: ***Images are selected from online forums using a bot. Subjects might not be aware that their image has been used for a scam and is uploaded to a forum and their identity is unknown. This makes it impossible to get consent of these subjects. If we would like to get a consent of these subjects, we should try to retrieve their identity which would be a violation of their privacy and would probably cause distress as they would become aware of the fact that their image has been used for the online romance scam.***

8. Are the requirements with regard to anonymity and privacy satisfied as stipulated in (§3.8)?
   ☒ **Yes**
   ☐ No, explain why not
   ☐ Uncertain, explain why
   Explanatory notes:

9. If any deception should take place, does the procedure comply with the general terms and conditions (no deception regarding risks, accurate debriefing) (§3.10)?
   ☒ **No deception takes place**
   ☐ The deception which takes place complies fully with the conditions (explain)
   ☐ The deception which takes place does not comply with the conditions (explain)
   If deception does take place, attach the method of debriefing
   Explanatory notes:

10. Is it possible that after the recruitment of experimental subjects, a substantial number will withdraw from participating because, for one reason or another, the research is unpleasant? (§3.5)
    ☒ **No**
    ☐ Yes, that is possible

If yes, then attach the recruitment text paying close attention to what is stated about this in the protocol.
Explanatory notes:

3. Questions regarding specific types of standard research

Answer the following questions based on the department to which the research belongs.

11. Does the research fall ***entirely*** under one of the descriptions of standard research as set out in the described standard research of the department? (Chapter 4)
    ☐ Yes, go to question 12
    ☒ **No, go to question 13**
    ☐ Uncertain, explain what about, and go to question 13
    Explanatory notes:

12. If yes, what type of research is it? Give a more detailed specification of parts of the research which are not mentioned by name in this description (for example: What precisely are the stimuli? Or: What precisely is the task?)

13. If no, or if uncertain, give as complete a description as possible of the research. Refer where appropriate to the standard descriptions and indicate the differences with your research. In any case, all possible relevant data for an ethical consideration should be provided.

    This research aims to develop a classifier which shows how likely it is that an image is used in a (online) romance scam. In the (online) romance scam, which is also known as catfishing, a scammer develops a (false) romantic relationship with a victim. After the victim has fallen in love with the scammer, the scammer often tries to steel money from the victim using platforms such as Western Union. Losses in the USA exceed $200 million on yearly basis. The Dutch Fraude Helpdesk received 134 reports from victims, together loosing around €1,5 million in 2017. Although these numbers indicate high financial losses, the victims experience the loss of the relationship once they find out that they are scammed more upsetting than the loss of money. Victims are seriously traumatised, sometimes even showing signs of PTSS or feeling suicidal.
    Awareness campaigns have not proven useful in prevention of the (online) romance scam. Neither is law very effective as these scams often go cross-border, which makes it hard to catch scammers. (Software) tools which recognise (signals of) the scam and raise red flags are suggested as a useful way of prevention, but to our knowledge, such tools do not yet exist.

    In this research images used as profile pictures by scammers in (online) romance scams will be automatically selected from online forums. Besides this a second set of images will be constructed from image databases which are freely available (for research purposes). Those images will be queried in a reverse image search engine (Google, Yandex, TinEye). The URLs of the images can be used, so that no downloading or uploading is needed. The output of the queries will be links to pages where these images or 'similar looking' images will appear. These pages will be mined with the goal to extract useful features. These features will be used to train a machine learning algorithm, which can be used as a classifier.
    In the end the classifier should be able to tell how likely it is, that an image is used in the (online) romance scam. If results are successful, this method can be implemented in a tool which prevents people from becoming victim in the (online) romance scam. However, implementing such a tool is not part of the research. The 'pipeline' of the research is visualised in figure 1.

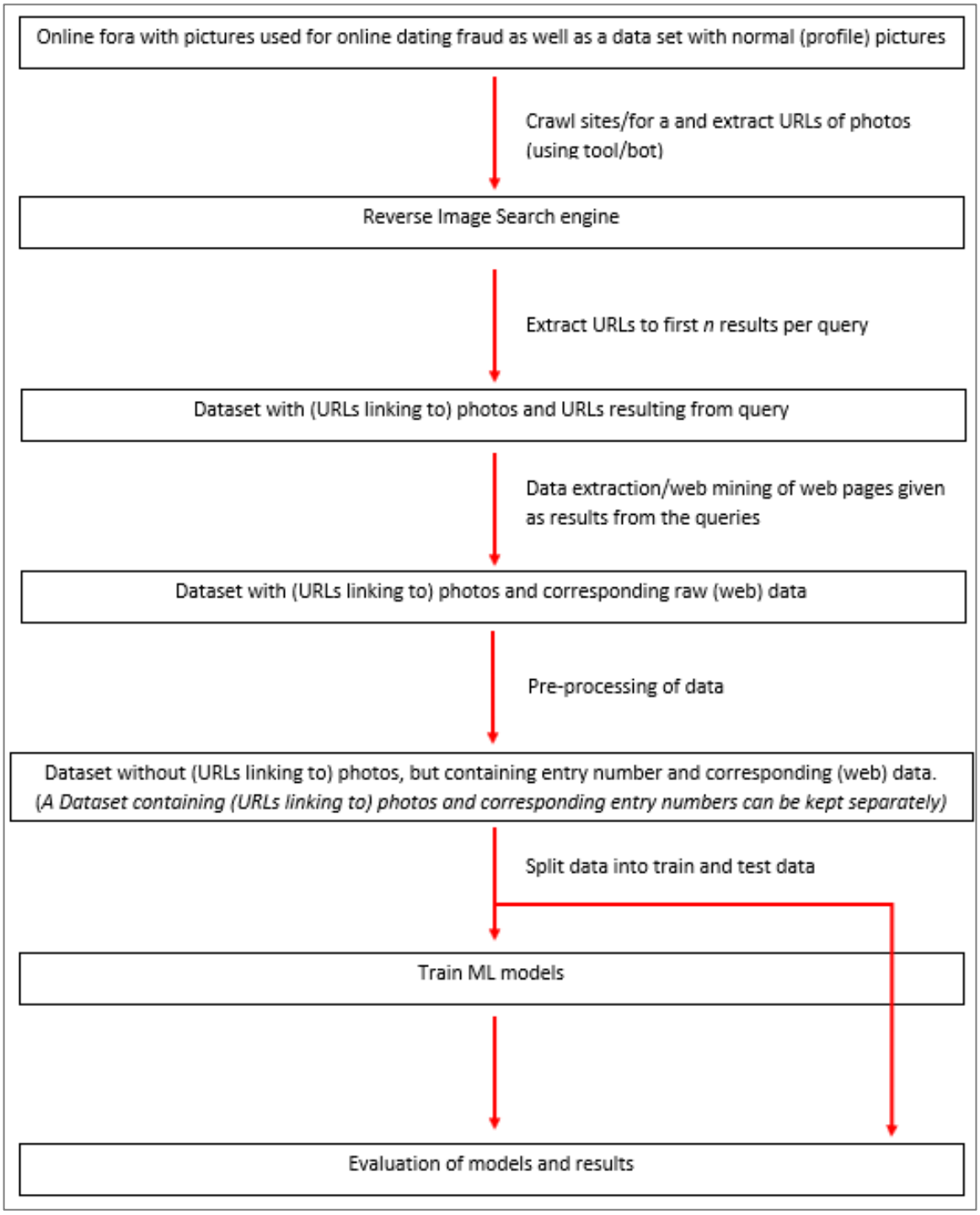


Figure 1: Pipeline of the research

It is likely to assume that the scammers used images of others without getting their consent. By using the images in which those victims of identity theft occur, we involve them as (anonymous) subjects in this research. As we do not know the identity of the subjects, we are not able to inform them that they are being subject in the research. However, considering the first paragraph of §3.2 and the explanation given at question 7, I would consider that this way of subject selection is ethical justifiable.

The images used in this research should be considered as Personal Identifiable Information (PII). A statement concerning the use of PII has been attached to this document. The use of PII, should be kept to a minimum. Data extraction and feature selection will be designed in such a way that no PII will be saved and/or used in training the classifier as this data is irrelevant (in accordance with §3.9). This also minimises the risk of loss of anonymity (in accordance with §3.8).

Although the subjects are in no way to our interest, their pictures are. In my opinion the ethical issues regarding the subjects in this research are minimal. Distress and other ethical issues such as the risk of loss of anonymity are basically non-existent and sufficient attention will be given to keep these minimal. In my opinion the social importance of creating a good classifier which potentially can decrease the number of victims of the (online) romance scam outweighs these ethical issues.

As a last note, I should mention that the use of the images can be seen as processing personal data in the GDPR. However, the processing of personal data can be justified for academic work by article 85 of the GDPR. Before starting the actual research, I will discuss my justification with Lesley Broos (Teacher ICT & Law and E-law). The research will only be started if this justification is sufficient in his opinion.

Statement concerning the use of PII

Personal Identifiable Information (PII) is information that can be used to uniquely identify, contact, or locate a single person, household, enterprise or institution or can be used with other sources to uniquely identify a single person, household, enterprise or institution,

The undersigned hereby undertakes to carry out the Final Project (192199978) as part of the master Computer Science at the University of Twente, in accordance with the following conditions:

1. He undertakes to keep confidential any PII which comes to his knowledge during the work on the Final Project.
2. He undertakes not to distribute any PII to others without written permission of (one of the members of) the Ethical Committee of EEMCS.
3. He undertakes to use the PII for purely scientific research only
4. This statement shall remain valid, even after conclusion of the work specified

Name and contact information:
Koen de Jong
s1367285
k.dejong-1@student.utwente.nl

Signature:

Place and Date: