RECOGNITION OF PERSONAL OPINIONS

in Dutch Public Records Requests

C. G. van Veen

MSc Thesis into the possibilities of Automatically Recognising Personal Opinions in Dutch Public Records

TU Delft

ZY LAB

Recognition of Personal Opinions
in Dutch Public Records Requests

by

C.G. van Veen

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Thursday November 26, 2020 at 09:30 AM.

Student number:	4300904	
Project duration:	Februari 10, 2020 – November 26, 2020	
Thesis committee:	Dr. ir. C. Lofi,	TU Delft, supervisor
	Prof. dr. ir. A. Bozzon	TU Delft, chair
	Dr. J.G.H. Cockx	TU Delft, member
	Prof. dr. ir. J. Scholtes,	ZyLAB Technologies B.V.

This thesis is confidential and cannot be made public until November 26, 2020.

An electronic version of this thesis is available at `http://repository.tudelft.nl/`.

Preface

Before you lies the thesis Recognition of Personal Opinions in Dutch Public Records Requests. This thesis aimed to find the possibility of automatically recognising personal opinions about policy within documents requested to be made public under the Dutch Public Records Request, which is called '*Wet van Openbaarheid van Bestuur*'. This research was performed over many months between February and November 2020 and was written to obtain a Master of Science degree at the Delft University of Technology within the Computer Science programme.

This research was conducted in collaboration with ZyLAB Technologies B.V. ZyLAB has developed an e-discovery solution which provides insight into vast amounts of data in an effective manner. In addition to the opportunity I received to start the development of a tool that could be helpful to many people, I additionally had the opportunity to have a look behind the scenes of such an innovating company. I would like to thank Jan, Jeroen and Zoe for all the time they made available for all my questions during the completion of my thesis. I would further like to thank Daniel and Carel, who made it possible for me to directly speak to the individuals responsible for handling Wob-requests.

I would additionally like to thank Dr. Lofi for his guidance, feedback and positivity over the past few months, including not only his advice about the techniques on which this research was built but also his feedback about my method of presenting ideas and results. Furthermore, I am grateful for Professor Bozzon and Dr. Cockx being part of my thesis committee.

Finally, I would like to thank my family for supporting me over the last 20 years of my study. It has been amazing.

C.G. van Veen
Amsterdam, November 2020

Abbreviations

biLSTM	bidiretional Long Short Term Memory
CNN	Convolutional Neural Network
BERT	Bidirectional Encoder Representations from Transformers
GRU	Gated Recurrent Units
IR	Information Retrieval
LSTM	Long Short Term Memory
MLM	Masked Language Modelling
NB	Naive Bayes
NER	Named Entity Recognition
NLP	Natural Language Processing
NSP	Next Sentence Prediction
OCR	Optical Character Recognition
PoS	Part-of-Speech
ReLu	Rectified Linear Unit
RNN	Recurrent Neural Network
SOTA	State-of-the-Art
SOP	Sentence Order Prediction
SVM	Support Vector Machine
Wob	Wet Openbaarheid van Bestuur

Summary

The Dutch version of the Public Records Request is named the *'Wet Openbaarheid van Bestuur'* (**hereinafter** Wob), which provides the public with the right to request access to records from any governmental institution (Ministerie van Algemene Zaken, 2019). The government has the obligation to provide information about policy and the execution of policy; however individuals who wish to obtain more detailed information, can request that a governmental body publicly disclose certain information. This is done by submitting a Wob-request. For a majority of governmental agencies, the number of Wob-requests and the size of the requests is substantial, to the degree that many of these governmental agencies hire full-time Wob-specialists to handle Wob-requests. ZyLAB Technologies B.V. has recognised the challenges that the Wob-specialists encounter on a daily basis and has examined the optimisation and automation of the process of dealing with a Wob-request. Nonetheless, certain steps that require large amounts of time continue to exists when processing Wob-requests. One of these tasks is described in this thesis, namely the redaction of personal opinions within internal deliberations.

The goal of this thesis was to investigate the possibility of automatically recognising personal opinions within internal deliberations in order to speed up the process of handling a Wob-request.

The research question is was follows:

> *How can the automatic recognition of personal opinions within internal deliberations in governmental documents be realised?*

Chapter 1 of this thesis will provide an introduction on the subject matter, and an explanation will be provided for why the redaction of personal opinions currently causes the process of handling a Wob-request to be slow.

Background information on the Wob will be provided in **Chapter 2**. In this chapter, the various grounds of refusal on which the disclosure of information can be refused – one of which is personal opinions within internal deliberations – will be discussed

The current methods of redacting personal opinions within internal deliberations will be explained in **Chapter 3**. During interviews held with Wob-specialists, it became apparent that ready-made rules did not exist to either classify documents as part of internal deliberation or choose which sentences to classify as personal opinions. Nonetheless, similarities exist between the different Wob-specialists as to how they handle the redaction of personal opinions. The majority of the time, Wob-specialists search for certain combinations of words, which could indicate that a sentence can be considered a personal opinion. This method of redacting can be considered to be closely related to a rule-based approach. For this reason, a rule-based approach was implemented as a

baseline for the results of this thesis. The remainder of this chapter will explain the challenges that are encountered when an attempt is made to automate the redaction of personal opinions. Several of these challenges are also encountered by humans, several challenges that only hold true when classification algorithms attempt to recognise personal opinions and several challenges that are encountered during implementation of the various algorithms.

In **Chapter 4**, based on predefined criteria, the relevant literature will be discussed. The goals was to find the most promising classification algorithms in order to improve on the set baseline. The classification algorithms were divided into three different categories based on their computational complexity, namely traditional machine learning approaches, deep learning approaches and bidirectional encoder representations from transformer based (**hereinafter** BERT-based) approaches. For each category, multiple classification algorithms will be discussed, and both differences and similarities among the classification algorithms within each category will be provided. The final set of classification techniques that were implemented are as follows:

- Traditional machine learning approach proposed by Kamal (2014) - an approach using different features such as term frequency-inverse document frequency, part of speech, position, opinion indicator seed words and negations. The classifier that was used was a Naïve Bayes classifier; however, due to promising results in other literature, the support vector machine classifier was used as well.

- Deep learning approach proposed by Liu and Guo (2019) - an approach using a convolutional neural network in combination with a recurrent neural network with attention.

- BERT-based approach proposed by Delobelle et al. (2020) - an approach using a Dutch pre-trained BERT model, called RobBERT, which is only trained on the masked language modelling task .

Chapter 5 will provide insight into how the rule-based classification was implemented to recognise the personal opinions within internal deliberations. For the rules four different categories of personal opinions have been created, namely advice, suggestions, opinions and expectation. For every category, different sentence constructions and dictionaries were created.

In order to obtain and validate the results, training and testing data had to be created. In an ideal situation, the texts that were redacted by the Wob-specialists would be used in the datasets as personal opinions. However, it seemed unfeasible to extract the redaction information from the ZyLAB ONE platform, which the documents were stored on. In **Chapter 6**, an explanation will be provided for how the training and test datasets were created. The Snorkel framework was used to label the training data with labelling functions, which were based on the rules created for the rule-based approach, as well as augment the data with transformation functions. The test dataset was created by manually rewriting the sentences from the documents that were redacted by the Wob-specialists.

In **Chapter 7, Chapter 8 and Chapter 9**, the implementation and results will be discussed for the traditional machine learning, deep learning and BERT-based classifications, respectively. For all the classification algorithms, experiments were performed to research which combination of variables and parameter settings would lead to ideal results for recognising personal opinions. In **Chapter 10**, the results of the different classification algorithms will be discussed. When flexible constraints were used, which meant that only one sentence had to be recognised as a personal opinion for a redaction that contained multiple sentences, the BERT-based approach proposed by Delobelle et al. (2020) showed the most promising results with a precision-value of 49% and a recall-value of 89%.

The classification results by the BERT-based approach were not and probably will not be ideal at any point due to the complexity of the automatic redaction of personal opinions. Plainly incorporating the BERT-based classification into the ZyLAB ONE platform could lead to a situation where the recognition of personal opinions could make the redaction process more complicated. During the conversations held with Wob-specialists, it was clear that, in order for the recognition of personal opinions to become a helpful tool, a few different requirements would need to be met. These will be discussed in **Chapter 11**. The visual requirements were simplicity and consistency, which were similar to the requirements for the information that needed to be available. The main requirements for processing a suggestion to a final redaction were that the suggestion had to be simple to change and enlarge and and it had to be simple to ask for a second opinion.

This thesis took the first steps towards the automatic recognition of personal opinions within internal deliberations. Nonetheless, further research would be necessary before this thesis' results could be provided to the clients of ZyLAB on the platform. The main steps for future research are as follows:

- Export sentences together with the redaction information from the ZyLAB ONE platform to avoid the need to label the data with labelling functions
- More data is needed in order to avoid the need for augmentation
- Create different training datasets and classifiers for different types of documents
- Conduct a more in-depth analysis of the most promising algorithms

Contents

Abbreviations v

Summary vii

List of Figures xiii

List of Tables xv

1 Introduction 1

2 'Wet Openbaarheid van Bestuur'
Policy Background of the Legal Framework of the Dutch Public Records Request 5
2.1 Absolute Grounds for Refusal 5
2.2 Relative Grounds for Refusal 6
2.3 Personal Opinions within Internal Deliberation 6

3 The Redaction of Personal Opinions 9
3.1 The Current Redaction Process of Personal Opinions. 9
3.2 Challenges of the Automatic Recognition of Personal Opinions 10
3.3 Approach to Automate the Recognition of Personal Opinions 12

4 Literature Review -
Classification Techniques 13
4.1 The Search for the Appropriate Algorithms 13
4.2 Evaluation Metrics 15
4.2.1 Trade-off Between Recall and Precision. 15
4.3 Traditional Machine Learning Algorithms 16
4.4 Deep Learning Algorithms. 18
4.5 Bidirectional Encoder Representations from Transformers (BERT)-based Algorithms 20
4.6 Conclusion on Classification Techniques 23

5 Rules-Based Redaction of Personal Opinions 25
5.1 Implementation Rules-based Classification 25
5.1.1 Error Analysis 26
5.2 Starting Point - Baseline with the Rules-based Approach 26

6 Creation of the Datasets 29
6.1 Creation of Training Data 29
6.1.1 Data Preprocessing 31
6.1.2 Data Labelling 32
6.1.3 Data Augmentation. 35
6.1.4 Data Slicing 36

6.2 Test Dataset. 36
6.2.1 E-mail Subset of Total Test Dataset 37
6.3 Conclusion Datasets 38

7 Traditional Machine Learning 41
7.1 Traditional Machine Learning Implementation 41
7.2 Traditional Machine Learning Classification Results 43
7.2.1 Traditional Machine Learning Test Dataset Results. 44
7.2.2 Traditional Machine Learning E-mail Dataset Results 45

8 Deep Learning Classification 49
8.1 Deep Learning Classification Implementation. 49
8.1.1 Word Embedding 49
8.1.2 Convolutional Layer 50
8.1.3 Bidirectional Long Short-Term Memory and Attention Layers 51
8.2 Deep Learning Classification Results 52
8.2.1 Deep Learning Classification Test Dataset Results 52
8.2.2 Deep Learning Classification E-mail Dataset Results. 53

9 BERT-based Classification 57
9.1 BERT-based Classification Implementation 57
9.2 BERT-based Classification Results 58
9.2.1 Results BERT-based Classification Test Dataset 58
9.2.2 Results BERT-based Classification E-mail Test Dataset 59

10 Comparing the Results 63
10.1 Another Method of Evaluating: Flexible Constraint Versus Strict Constraints . 65

11 Creating a Tool 67
11.1 Visual Requirements 67
11.2 Information Requirements 68
11.3 Processing Requirements 69

12 General Discussion 71
12.1 Main Findings 71
12.2 Future Research 75

References 79

A Wet Openbaarheid van Bestuur:
Legal Background 81

B Rule Based Implementation 87

C Overview Machine Learning Results 95

D Overview Deep Learning Results 99

E Overview BERT-based Results 101

List of Figures

4.1 Different Classification Categories 14
4.2 The architecture of BERT-FC (left) and TD-BERT (right) (Gao et al., 2019) . . . 21
4.3 Multi-dimension Information Integration using Highway Network (Song, 2020) 22

6.1 Different Sets of Data on the ZyLAB ONE Platform 30
6.2 Different Steps to Create Training Dataset 31
6.3 Data Preprocessing Process as mentioned by (Anandarajan et al., 2019) 32
6.4 The Snorkel Framework (Snorkel, 2019) 33
6.5 Labelling Pipeline in Snorkel (Snorkel, 2019) 33
6.6 Lengths of Sentences Total Test Dataset 37
6.7 Lengths Sentences Mail Test Dataset 38

7.1 Traditional Machine Learning Algorithm proposed by Kamal (2014) 42
7.2 Precision-recall (PR) Curve Traditional ML - Test 45
7.3 PR-Curve Traditional Machine Learning - Mail 47

8.1 Neural Network Proposed by Liu and Guo (2019) 50
8.2 PR-Curve DL Classification -TEST 54
8.3 PR-Curve DL Classification - MAIL 55

9.1 PR-Curve BERT-based Classification -TEST 60
9.2 PR-Curve BERT-based Classification - Mail 61

List of Tables

4.1 Overview Classification Techniques per Category 24

5.1 Results Rule-based Classification 27

6.1 Results Different Labelling Functions 34
6.2 Results Labelling Functions Focusing on either Recall or Precision 34
6.3 Overview Transformations Functions 35
6.4 Statistics Sentences Total Test Dataset 37
6.5 Statistics Sentences Mail Test Dataset 38
6.6 Overview Training Datasets 39
6.7 Used Variables for Creating the Training Data 39
6.8 Overview Test Datasets 39

7.1 Features proposed by Kamal (2014) 42
7.2 Information Gain per Feature described by Kamal (2014) 43
7.3 Different N-grams 43
7.4 Ranking Results Traditional Machine Learning - Test 44
7.5 Ranking Results Traditional Machine Learning - Mail 46
7.6 Comparison Final Results Test and Mail for Traditional Machine Learning . . 46

8.1 Initial Results Original Setup Deep Learning 51
8.2 Ranking Results Deep Learning - Test 52
8.3 Ranking Results Deep Learning - Mail 53
8.4 Comparison Final Results Test and Mail for Deep Learning 55

9.1 Ranking Results BERT-based Classification - Test 59
9.2 Ranking Results BERT-based Classification - Mail 60
9.3 Comparison Final Results TEST and MAIL for BERT-based Classification . . . 61

10.1 Overview Final Results - Test 64
10.2 Overview Final Results - Mail 64
10.3 Overview Final Results - Flexible Constraints 64
10.4 Difference Between Strict and Flexible Constraints 66

12.1 BERT-based Final Results - Flexible Constraints 74

C.1 Traditional ML Results (High Recall - Test Dataset - Lemmatized) 95
C.2 Traditional ML Results (High Recall - Test Dataset - Not Lemmatized) 96
C.3 Traditional ML Results (High Precision - Test Dataset - Lemmatized) 96
C.4 Traditional ML Results (High Precision - Test Dataset - Not Lemmatized) . . . 96
C.5 Traditional ML Results (High Recall - Mail Dataset - Lemmatized) 97
C.6 Traditional ML Results (High Recall - Mail Dataset - Not Lemmatized) 97

C.7 Traditional ML Results (High Precision - Mail Dataset - Lemmatized) 97
C.8 Traditional ML Results (High Precision - Mail Dataset - Not Lemmatized) . . . 98

D.1 DL Results (High Recall - Test Dataset - Lemmatized) 99
D.2 DL Results (High Recall - Test Dataset - Not Lemmatized) 99
D.3 DL Results (High Precision - Test Dataset - Lemmatized) 100
D.4 DL Results (High Precision - Test Dataset - Not Lemmatized) 100
D.5 DL Results (High Recall - Mail Dataset - Lemmatized) 100
D.6 DL Results (High Recall - Mail Dataset - Not Lemmatized) 100
D.7 DL Results (High Precision - Mail Dataset - Lemmatized) 100
D.8 DL Results (High Precision - Mail Dataset - Not Lemmatized) 100

E.1 BERT Results (High Recall - Test Dataset - Lemmatized) 101
E.2 BERT Results (High Recall - Test Dataset - Not Lemmatized) 101
E.3 BERT Results (High Precision - Test Dataset - Lemmatized) 102
E.4 BERT Results (High Precision - Test Dataset - Not Lemmatized) 102
E.5 BERT Results (High Recall - Mail Dataset - Lemmatized) 102
E.6 BERT Results (High Recall - Mail Dataset - Not Lemmatized) 102
E.7 BERT Results (High Precision - Mail Dataset - Lemmatized) 102
E.8 BERT Results (High Precision - Mail Dataset - Not Lemmatized) 102

1

Introduction

The Dutch version of the Public Records Request (United States Department of Justice, 2020) is named the *'Wet Openbaarheid van Bestuur'*, which provides the public with the right to request access to records from any governmental institution (Ministerie van Algemene Zaken, 2019). This right is a result of the principle laid down in the Dutch Constitution, which reads as follows:

> *'In the performance of its task, the government shall endeavour to ensure that public access is provided in accordance with the law'.* (Nederlandse Grondwet, 2020)

The Wob is frequently described as the law that keeps citizens updated on their government (United States Department of Justice, 2020). The government has the obligation to provide information about any policy and the execution of that policy. However, individuals who wish to obtain more detailed information can request that a governmental body publicly discloses certain information. This is done by submitting a Wob-request (Ministerie van Algemene Zaken, 2019). Confidentiality is the exception to the rule and must generally be justified on the grounds of refusal, which will be discussed in more detail in Chapter 2.

Between 2010 and 2019, nearly 23,000 Wob-requests were filed at the ministries alone, thereby excluding provinces and municipalities, among others (Overheid.nl, 2020). In addition to the vast amounts of Wob-requests, it is highly labour intensive to process a Wob-request. To complete a Wob-request, multiple steps have to take place. Several of these steps involve collecting all the media that is relevant to the request. Other steps are deleting duplicates from the total set of documents to provide a clear perspective of the administrative matter about which information is requested as well as removing all personal information from the relevant documents. For a majority of governmental agencies, the number of Wob-requests and the sizes of the requests is substantial to the degree that many of these agencies hire full-time Wob-specialists. For a majority of these Wob-specialists, a legal education is required. This is caused by the difficulty of the trade-off that has to be made between information that should be publicised and information that should not be disclosed to the public. The complexity of this decision stems from protecting officiaries need to speak freely to each other but simultaneously

providing the public with insight into the creation and implementation of policy.

ZyLAB Technologies B.V. (**hereinafter** ZyLAB) has recognised these challenges and has examined optimising and automating the process of dealing with a Wob-request. ZyLAB has developed a platform where governmental bodies can process documents relevant to a Wob-request. Clients can upload the required documents to the platform to ensure that each document is stored in the same shared place. Subsequently, the software removes (near) duplicates automatically, which occurs when documents are uploaded multiple times by different officiaries or when different versions of e-mail conversations or documents are uploaded. With the help of several initial keywords and a machine learning algorithm, the documents that are not relevant to the specific Wob-request are filtered from the set of uploaded documents. This is why it is not problematic when an excessive number of documents are initially uploaded, and, as a result, it is plausible that all the relevant documents are in the final set of documents. Thereafter, the redaction phase begins. With the help of entity extraction software, personal information such as e-mail addresses, names, phone numbers, and bank account numbers, among others, can be automatically redacted. As a result, the redactions only have to be manually checked. All of these optimisations reduce the total amount of time needed to handle a Wob-request. Nonetheless, there are several steps to processing a Wob-request that still take a large amount of time. In this research, one of these steps was analysed.

The task that will be researched in this thesis is the automatic recognition of personal opinions within internal deliberations (**hereinafter** personal opinions). It is important for information concerning the personal opinions of ministers, directors or officiaries in documents that were drawn up for internal deliberation to be redacted when this information can be harmful to the officiary who mentioned it or when the information could damage current policy. The term 'internal deliberation' refers to the deliberation on an administrative matter, which includes views and opinions that are supported by one or more individuals (Wetten.nl, 2015). Automating the process of redacting personal opinions is complex for various reasons. An important reason why it can difficult to automate the redaction of personal opinions is that it can be possible for different people to redact the same sentence differently because a ready-made answer on what should be considered a personal opinion does not exist. In addition to this challenge, other challenges can be encountered. These will be discussed in more detail in Section 3.2.

The main purpose of this research was to investigate the possibility of automatically recognising personal opinions. Firstly, it was necessary to collect knowledge about how personal opinions are currently redacted. Then research was done into what the current state-of-the-art (**hereinafter** SOTA) techniques are that have been developed for similar fields of research. Subsequently, those techniques had to be implemented for the Dutch language. Finally, based on the results of the different implemented classification techniques, a suggestion could be made based on expert interviews regarding how these algorithms could be translated into a helpful tool for Wob-specialists.

The task analysed in this thesis can be considered to be a sentence classification problem where, for each sentence, an assessment is made on whether that sentence should be

considered a personal opinion. The reason this is considered to be a sentence classification problem will be explained in Chapter 3. Automatic recognition is important for the redaction process because nearly all other previously mentioned steps within the Wob-request process can be (partly) performed automatically. This led to the following research question:

> *How can the automatic recognition of personal opinions within internal deliberations in governmental documents be realised?*

This research question was divided into six different sub-questions. These sub-questions provided the structure for this research. In each chapter, at least one sub-question will be covered. By combining the information in all the chapters, this thesis aimed to provide an answer to the main research question. The sub-questions were as follows:

1. How is the redaction of personal opinions within internal deliberation currently addressed by Wob-specialists?

2. What are the main challenges regarding the automatic recognition of personal opinions within deliberations?

3. Which algorithms can be used for recognising personal opinions within deliberations?

4. How should a dataset be obtained to train and subsequently test the chosen algorithms?

5. How should the chosen algorithms be implemented, and what combination of variables and parameter settings will lead to the most effective results for recognising personal opinions?

6. How should the automatic recognition of personal opinions be incorporated into the ZyLAB One platform in order to be a helpful tool for Wob-specialists?

The main contributions of this research are the following:

- An extensive literature review on neighbouring fields of research, such as recognising subjectivity versus objectivity, polarity and sentiment analysis. By fixing the search parameters, how the relevant literature was identified is clear, and, based on predefined criteria, a set of promising literature was selected.

- The implementation of the most promising algorithms, which were developed for the English language, were transformed for the Dutch language.

- The method by which the dataset was created, labelled and augmented with the help of the Snorkel framework.

- Finally, the research carried out to investigate how to create a tool that may be helpful for improving the process of redaction.

In the following chapters, all the sub-questions stated in this chapter will be covered to provide an answer to the main research question. In Chapter 2, all the context information concerning the policy of the Wob will be provided. In Chapter 3, the current redaction process of personal opinions will be discussed. Thereafter, the challenges that can be encountered while attempting to automate the redaction process of personal opinions will be discussed. Continuing to Chapter 4, an overview will be provided of classification techniques which have proven themselves to perform well in neighbouring disciplines. Subsequently, in Chapter 5, the creation of the rule-based approach will be discussed; this approach functioned as a baseline for this thesis. Chapter 6 will explain how the datasets used for training and testing were created. Thereafter, the implementation and results are explained of the traditional machine learning approach, deep learning approach and BERT-based approach in Chapter 7, Chapter 8, Chapter 9 respectively. An overview of all the results of the different algorithms will be provided in Chapter 10. Chapter 11 will present the investigation of how to transform the most effective algorithm into a helpful tool for Wob-specialists. Finally, Chapter 12 will be dedicated to the final conclusions and recommendations for future research.

2

‘Wet Openbaarheid van Bestuur’ Policy Background of the Legal Framework of the Dutch Public Records Request

The basic principle of the Wob is that information on administrative matters is public. Therefore, the government should provide information by its own initiative as soon as this is in the interest of ideal and democratic governance. Nonetheless, the Wob gives citizens the right to request information either on all policies that have not been published or as more detail is desired. It is important to understand that confidentiality is the exception to the rule. Nonetheless, occasionally, it is necessary to keep certain information confidential. When this is the case, the confidentiality needs to be justified.

In this section, global background information on the grounds of refusal for the Wob will be provided. The entire legal framework of the Wob can be found in Appendix A. Firstly, the absolute followed by the relative grounds of refusal will be explained. Finally, the internal deliberation will be discussed.

2.1. Absolute Grounds for Refusal

In Chapter IV of the Wob, described in Appendix A, several exceptions are set regarding which provisions of information shall be refused. When refusing to provide information, it is unnecessary to weigh the importance of the disclosure against the importance of keeping the information confidential. The withholding of information applies only to that part of the requested information where the absolute grounds for exception are relevant. There are multiple grounds on which information can be refused, which will be discussed separately:

- **Unity of the Crown** - information shall not be provided when it could jeopardise the unity of the Crown, referring to the unity of the King and the ministers. This unity would be jeopardised in the event that information were to be provided, which would reveal differences of opinion between the King and the Cabinet.

- **Safety of the State** – information shall not be provided when it could harm the safety of the State.

- **Business and Manufacturing Data** – information shall not be provided when the information concerns business and manufacturing data which have been confidentially communicated to the government by natural or legal entities.
- **Personal Data** – information shall not be provided when the information contains personal data as referred to in Paragraph 2 of Chapter II of the Personal Data Protection Act (Wetten.nl, 2018).

2.2. Relative Grounds for Refusal

Apart from the absolute ground of refusal, it is possible to refuse to disclose information on relative grounds for refusal. In doing this, a trade-off should be made between the public interest in the provision of information and the special interests that need to be protected. The provision of information shall be omitted to the extent that its interest does not outweigh the following relative grounds of refusal:

- **Relations with other States** – information can be kept confidential when it could harm Dutch relations with other states and international organisations. Concrete instructions are needed for this, which may consist of an explicit indication to the Netherlands that the confidentiality of the documents in question is expected.
- **Economic or Financial Interest of the State** – information can be kept confidential when it could harm the economic or financial interests of the State.
- **Investigations and Prosecution** – information can be kept confidential when it could harm the investigation and prosecution of criminal offences. Those investigations could be hindered by the disclosure of information that investigators or the Public Prosecution Office has previously collected. This additionally includes the means by which the officials implement strategies.
- **Inspection and Control** – information can be kept confidential when it could harm inspection, control and supervision needed by governmental bodies to establish non-criminal offences.
- **Information of Personal Nature** – information can be kept confidential when it could harm citizens who provided information of a personal nature in the confidence that it would only be used by public authorities for the purpose intended.
- **Disproportionate Favouritism** – information can be kept confidential when it could cause disproportionate favouritism or disadvantage to natural or legal persons involved in the matter or to third parties.

2.3. Personal Opinions within Internal Deliberation

In addition to the grounds of refusal mentioned in Section 2.1 and Section 2.2, there is one other ground on which information can be refused regarding its disclosure to the public. The Wob stipulates that no information is provided about the personal opinions about policy of ministers, directors or officiaries in documents written for internal deliberation.

By 'internal deliberation', the Wob refers to the following: 'the deliberation on an

administrative matter within an administrative body, or within a circle of administrative bodies in the context of joint responsibility for an administrative matter' (Lexman Advocaten, 2020). This means that views and opinions that are supported by more than one person, a group of persons or a governmental body may additionally fall under the concept of personal policy views. However, there is an exception to this rule. Since the aim is ideal and democratic governance, information about personal policy views can be provided but in a form that cannot be traced back to individuals. This is different in the event that the person who has expressed these opinions has consented to the provision of information in a traceable form because, in that case, the disclosure is allowed.

3

The Redaction of Personal Opinions

In Section 3.1, an overview will be provided regarding how the current process of redacting personal opinions occurs. This section will provide an answer to the next sub-question of the main research question:

> *How is the redaction of personal opinions within internal deliberation currently addressed by Wob-specialists?*

When automating the redaction process of personal opinions, multiple challenges are encountered. In order to find a solution to automatically recognise the personal opinions in such a way that it can be helpful during the redaction process, these challenges need to be resolved. The possible challenges will be discussed in Section 3.2, and the information in that section was used to answer the following sub-question:

> *What are the main challenges regarding the automatic recognition of personal opinions within deliberations?*

After discussing the various challenges that can be encountered, an approach on how to automate the recognition of personal opinions will be discussed in Section 3.3.

3.1. The Current Redaction Process of Personal Opinions

To complete this thesis, various interviews were held with multiple officiaries employed as full-time Wob-specialists at governmental bodies of provinces and municipalities. During these interviews, it was important to obtain a clear perspective regarding how these officiaries approached the redaction of personal opinions. The interviews were held with the current clients of ZyLAB, which meant they already used the ZyLAB platform. The clients already made use of certain improvements to the Wob-process, such as the automatic removal of duplicate documents and the automatic redaction of personal information, such as names, telephone numbers and e-mail addresses. Despite the improvements made by ZyLAB for the process of handling a Wob-request, it seemed that, nonetheless, each document had to be read entirely to identify the remaining texts that had to be redacted in order to remove personal opinions.

The first step in the process of redacting personal opinions is assessing whether the

document is part of any kind of internal deliberation. From the interviews with Wob-specialists, it could be seen that all documents written among officiaries were considered a part of internal deliberations. The most frequently used media for internal deliberations were e-mails among officiaries. Less common but additionally crucial documents contained transcriptions from consultations or meetings. However, per document, an assessment was made by the Wob-specialist instead of using predefined rules about what documents should fall under internal deliberations.

The next step is removing the personal opinions from the documents that are considered part of internal deliberations. During the interviews, it became apparent that, for this part as well, the redaction of personal opinions was subjective to the officiary that redacted the documents. Frequently, background knowledge was required to assess whether or not a sentence was a personal opinion. Wob-specialists frequently became accustomed to the writing styles of their colleagues, which could facilitate the redaction process. However, this differed per governmental body, which led to the same sentence being recognised as a personal opinion in several cases but not in other cases. During the interviews, several documents that were already redacted were discussed. In certain cases, a sentence that seemed to be a personal opinion was not redacted as such. When the reason was asked of the Wob-specialists, the answer was frequently that this specific sentence could have been redacted as a personal opinion, but, for a certain reason, the decision was made that it was not necessary to redact the information. It was apparent that, additionally, for redacting personal opinions, no ready-made answer existed. Nonetheless, several similarities could be identified about how the Wob-specialists addressed the redaction of personal opinions. In a majority of cases, the Wob-specialists searched for certain combinations of words, such as follows: '`I expect ...`', '`I advise ...`', '`we think...`', '`he expects...`'. This way of working can be compared to a rule-based approach because, when using a rule-based classification algorithm, an 'IF condition THEN conclusion' construction is used.

3.2. Challenges of the Automatic Recognition of Personal Opinions

Solving the problem of automatically recognising personal opinions cannot be trivial for a number of reasons. The challenges that can be encountered were divided into the following categories: challenges that humans encountered, challenges encountered due to the lack of understanding of a language when Natural Language Programming (**hereinafter** NLP) techniques are used and challenges encountered during the implementation of the different algorithms.

Two of the challenges that are encountered when automating the redaction of personal opinions additionally hold true for humans when they redact personal opinions:

- The first challenge is that a different decision can be made about the same sentence by different Wob-specialists. As explained in Chapter 2, absolute and relative grounds of refusal exist, which make it possible to keep sensitive information confidential. Redacting personal opinions falls under the relative grounds of refusal, which means that a trade-off should be made between the public interest to disclose

the information and the importance of keeping the information confidential. This means that different Wob-specialists can make different decisions about the same sentences.

- Another challenge that is encountered while automatically redacting personal opinions is that, in certain cases, background information on the administrative matter that has been requested is needed. For individuals who perform the redacting and who are additionally familiar with the specific administrative matter, the process of redacting is simpler. When automatically redacting personal information, this knowledge is not available to the algorithm that classifies the sentences, which increases the difficulty of classifying certain sentences.

However, several of the challenges that are encountered are done proficiently and absentmindedly by humans:

- One of these challenges is the detection of the boundaries of personal opinions. The data that is used to train the algorithm is of a fixed size, which could either be a paragraph or a sentence; this additionally holds true for the data that the algorithm is tested on. However, in reality, the personal opinion can extend over multiple sentences, paragraphs or, in certain cases, entire documents. Sentences are the minimum units of text that must be redacted, which means that it is unnecessary to redact parts of sentences. When a part of a sentence is considered a personal opinion, the entire sentence can be redacted. For this reason, in this thesis, the focus was on sentence classification. However, a solution has to be found when the personal opinion extends beyond more than one sentence.

- Another advantage to humans redacting personal opinions is that humans can readily recognise ambiguity in texts. A challenge that is encountered during the automatic redaction of personal opinions is that, in certain cases, opinions can be stated as facts and facts stated as personal opinions. In the first example below, the sentence is stated as a fact because certain words are missing. If the same sentence were to be formulated as in the second example below, it would be apparent that it could be considered a personal opinion. However, this first sentence was classified as a personal opinion, which meant that the Wob-specialist was certain that this was actually not a fact but an opinion, including without the important words ‘we assume...’.

  1. The concentrations of [CHEMICAL SUBSTANCE] will not change much.

  2. We assume the concentrations of [CHEMICAL SUBSTANCE] will not change much.

Furthermore, several challenges exist that are encountered during the implementation of different algorithms:

- The first challenge that needs to be resolved is the absence of data which can train the various algorithms. In an ideal situation, data that already has been redacted would be extracted from the ZyLAB ONE platform. This data should then contain sentences

that were redacted as personal opinions and sentences that were not redacted. This data could be used for training and testing. However, currently, no method exists to extract this information; therefore, a solution needs to found to label the data in such a way that a large amount of reliable training data can be generated.

- Another challenge that is encountered during implementation is that nearly all traditional and SOTA techniques have been developed for the English language. This means that the common language-dependent dictionaries and word embeddings, amongst other tools, cannot be used for this problem. This means that substitutions need to be found for the Dutch language, which, in certain, cases could be difficult because less research has been done in Dutch.

To conclude, as discussed in this section, multiple challenges need to be resolved when attempting to automate the recognition of personal opinions. In this research, an attempt was made to further analyse these challenges, and, in order to automate the recognition of personal opinions, an attempt was made to resolve these challenges.

3.3. Approach to Automate the Recognition of Personal Opinions

In Section 3.1, the current techniques used by Wob-specialists to redact personal opinions were discussed. During a number of interviews, it became clear that a ready-made answer on how to redact the relevant documents did not exist. However, the most important observation from these interviews was that all the Wob-specialists searched for certain combinations of words. This method of working can be compared to a rule-based classification approach, and, for this reason, rules were created to automatically recognise personal opinions. This rule-based approach functioned as a baseline for this thesis. In a majority of cases, a rule-based approach performs effectively on recall; however, in a majority of cases, a rule-based approach lacks precision. In order to increase precision, more advanced algorithms have to be implemented. In the next chapter, based on relevant literature, various SOTA algorithms will be compared. By setting different parameters, the most promising algorithms can be chosen. Those algorithms were implemented in order to automatically redact personal opinions, and experiments were performed to investigate whether or not the implemented algorithms would show similar results in recall yet show increased precision.

4

Literature Review - Classification Techniques

In this chapter, an overview of the relevant literature will be provided, from which, based on related disciplines, the most promising algorithms will be discussed. From each category, which will be described in Section 4.1, the most promising algorithm was chosen. This could provide an answer to the sub-question stated in Chapter 1:

> *Which algorithms can be used for recognising personal opinions within deliberations?*

Apart from the various categories that will be described in Section 4.1, how the most promising literature was found will additionally be explained. With the help of this information, the results could be reproduced. Subsequently, in Sections 4.3, 4.4 and 4.5 the various algorithms using a traditional machine learning approach, a deep learning approach and a BERT-based approach, respectively, will be explained.

4.1. The Search for the Appropriate Algorithms

In order to identify the appropriate algorithms for the problem of recognising personal opinions, the relevant literature was searched based on various criteria:

- Only literature focused on related disciplines will be discussed in this thesis, as this specific task has not yet been studied. Tasks that are closely related to this task are sentiment analysis, objectivity versus subjectivity recognition and polarity classification because, in all of these disciplines, a distinction is made between factual data and other versions of texts.
- The data that had to be processed for this research was written in Dutch, which is different from English. For this reason, the literature focused on a Dutch dataset with valuable results prevailed over algorithms tested on an English dataset with equal or possibly more ideal results.
- Literature focused on classification at the sentence level was preferred over algorithms focused on document-level or aspect-level classification because the automatic recognition of personal opinions is considered to be a sentence-level classification problem.

- Literature focused on datasets written in the same context as personal opinions was preferred. For this thesis, the documents were written at the moment of internal deliberation to discuss certain policies. The closest dataset to the dataset of this task was the 'SUBJ'-dataset, which is a subjectivity dataset based on movie reviews which contains objective and subjective human-written sentences (Cornell, 2019). The texts in the SUBJ-dataset were closest to the data that was used for this research because a personal opinion can be considered to be a special version of a subjective sentence. Another related dataset which is commonly used is the Internet Movie Database dataset. However, this dataset consists of positive and negative sentiments, which means that both classes contain subjective sentences, and the objective sentences are missing. This further holds true for the third most commonly used dataset, namely the Stanford Sentiment Treebank, which additionally consists of sentences with positive and negative sentiments. Finally, custom datasets were used which were not publicly available and which contained scraped Twitter data or reviews about various products or services.

- The literature had to be reliable. For this, it was important to investigate the metadata of the literature, such as the number of citations, in which journal articles were published and articles' years of publication. All articles used in this research were found via Google Scholar.

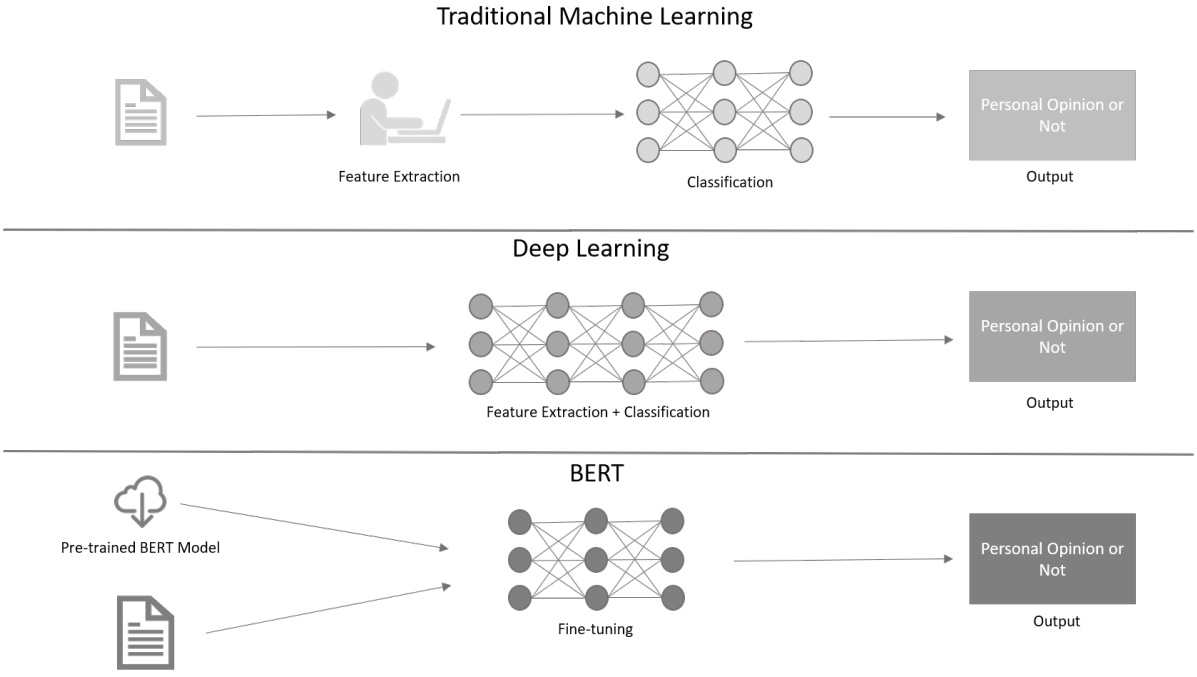

Figure 4.1: Different Classification Categories

The algorithms chosen for this research varied in computational complexity. The most promising algorithms had the highest computational complexity. However, if algorithms that were less computationally complex performed equally well, then it was possible for the less computational complex algorithm to be preferable. This is why, additionally, more traditional techniques were experimented with. The different categories from which

algorithms were tested were traditional machine learning approaches, deep learning approaches and approaches using versions of the pre-trained BERT model. It should be noted that BERT-based models can be considered a specific version of a deep neural network, however, due to the promising results BERT-based models show, it will be treated as a separate category. In Figure 4.1 a schematic overview is provided to show how these different approaches function.

In Table 4.1, an overview is provided of the studied approaches, starting with the least computationally complex category, namely the traditional machine learning algorithms. Subsequently, the deep learning algorithms are shown, which had an increase in computational complexity, and, finally, the different algorithms using a version of BERT are shown, which were the most computational complex. For each study, the classifier, the researchers, the year, the dataset that was used, the scores of the algorithm, the number of citations and the journal in which an article was published are shown. The rule-based approaches are not included, and this is because the rule-based algorithm that was used for this research was custom-made for this task.

4.2. Evaluation Metrics

In this section, the metrics to validate the results will be discussed. Additionally, the importance of the different metrics will be discussed in Section 4.2.1.

The equations of the recall, precision and F1 score are stated in Equations 4.1, 4.2 and 4.3 respectively.

$$\text{Recall} = \frac{\text{true positives}}{\text{true positives} + \text{false negatives}} \tag{4.1}$$

$$\text{Precision} = \frac{\text{true positives}}{\text{true positives} + \text{false positives}} \tag{4.2}$$

$$F_1 = 2 * \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}} \tag{4.3}$$

To provide additional information on how the classification performed on the sentences labelled as the negative class, the accuracy metric could be calculated, shown in Equation 4.4. However, it is important to note when imbalanced datasets are used, the accuracy can provide a skewed view on the performance of the classifier.

$$\text{Accuracy} = \frac{\text{true positives} + \text{true negatives}}{\text{true positives} + \text{false positives} + \text{true negatives} + \text{false negatives}} \tag{4.4}$$

4.2.1. Trade-off Between Recall and Precision

In an ideal situation, classifiers would perform well on both recall measures and precision measures. However, frequently, the classifier performed well on one of the metrics. When choosing certain algorithms or setting certain parameters, it was possible to focus on either recall or precision. When evaluating the results, occasionally, a trade-off had to be made between recall and precision. In order to make the correct decision to either focus on recall or precision, Wob-specialists were involved in this decision.

Based on the interviews held with the Wob-specialists, a decision was made to consider the F1-score as most important because, in this way, both recall and precision were included in the decision. The reason precision was considered important by the Wob-specialists was that, when redacting the documents, a low precision would slow down the process, because in that case many incorrect suggestions were made by the classification algorithm. However, recall was considered more important than precision. The extra work as a consequence of low precision only had an influence on the speed of redacting personal opinions. However, the consequence of a lower recall was that the chance of missing a personal opinion would increase. The cost of missing a personal opinion was higher than a slower redaction process as long as it was faster than manual redaction.

4.3. Traditional Machine Learning Algorithms

Traditional machine learning approaches simulate the way humans learn from their past experiences to acquire knowledge and apply it to making future decisions; therefore, traditional machine learning teaches the model using a training dataset from which certain features are extracted, and then the trained model is applied to the test dataset (Tripathi and Naganna, 2015).

A common way of classifying sentences within NLP is the use of a dictionary, such as WordNet used by Chong et al. (2014) and Montejo-Ráez et al. (2013). WordNet is a dictionary in which words are grouped in synsets to ensure that words in the same synsets can be called synonyms. However, hypernyms, hyponyms and meronyms, among others, can be found in WordNet as well. Another commonly used dictionary is SentiWordNet, which is based on WordNet. However, every synset in SentiWordNet has three scores, namely a positivity score, negativity score and a objectivity score. As stated by Chong et al. (2014) a sentence is classified as subjective if SentiWordNet contained a word from that sentence and had a positivity or negativity score. Montejo-Ráez et al. have used the dictionary in a more complex manner, namely by using the scores from SentiWordnet with an added random walk analysis of the concepts found in the text over the WordNet graph. For the random walk, it is assumed that a provided word is more likely to hit another word with the same semantic orientation before hitting a word with a different polarity. Every tweet is represented as a vector of weighted synsets that are semantically close to the terms included in the post. The final score is the weights associated with synsets after the random walk, which are normalised to vectors of 'concepts'. The final polarity score is obtained by the product of this vector with the associated SentiWordNet vector of scores:

$$p = \frac{\mathbf{r} * s}{|t|} \tag{4.5}$$

where p is the final score, r is the vector of weighted synsets computed by the random walk algorithm of the tweet text over WordNet, s is the vector of polarity scores from SentiWordNet, and t is the set of concepts derived from the tweet. A support vector machine (**hereinafter** SVM) was used to classify the sentences with the known vector space model to build the vector of features. Both the algorithms described by Chong et al. and by Montejo-Ráez et al. resulted in a F1-score of 55% and 62.8% respectively, which

was slightly more ideal than random. In addition, the dictionaries used were not available in Dutch, were lower in quality or missed certain information. For these two reasons, both algorithms were not used further in this research.

Another type of feature consists of values about how frequently certain words or, further, combinations of words occur. Tripathi and Naganna (2015) have calculated the following features for unigrams, bigrams and trigrams:

- **Term occurrence** - defines the absolute number of occurrences of a term.
- **Term frequency** - defines the relative frequency of a term in a document.
- **Binary term occurrence** - term occurrence as a binary value, where the value is zero if the word does not occur in the text and one if the term does occur in the text.
- **Term frequency - inverse document frequency (hereinafter TFIDF)** - reflects how important a term is, as seen in Equation 4.6.

$$w_{i,j} = tf_{i,j} * \log \frac{N}{df_i} \tag{4.6}$$

where $tf_{i,j}$ is the number of occurrences of i in j, df_i is the number of documents containing i and N is the number of documents in the corpus. Other examples are features used by Kamal (2014)), who proposes the following features for the various unigrams:

- **TF-IDF** - reflects how important a term is, as seen in Equation 4.6.
- **Part-of-Speech**: indicates what type of speech the unigram is, such as a noun, verb or adjective, among others.
- **Position** - indicates at which part of the sentence the unigram is written. A possibility could be that if a word appears at the beginning of the sentence, it is assigned with value zero. If the term appears in the middle of the sentence, it is assigned the value one and, finally, at the end of the sentence, a value of two.
- **Opinion Indicator Seed Word** - indicates if the word is commonly used for expressing positive or negative sentiment. Examples of such terms are 'frightening', 'happiness' and 'funny'.
- **Negation** - indicates if the unigram is a negation word.

Finally, the obtained vectors should be separated in the different classes, such as objective versus subjective or negative versus positive, by a classifier. Sharma and Dey (2012) have tested a few commonly used classifiers, namely SVM, Naive Bayes (**hereinafter** NB), decision tree, maximum entropy, k-nearest neighbour and Winnow. For all different feature selection methods, the SVM classifier performed most effectively, followed by the NB classifier. This was not unexpected, as all the other algorithms, that were mentioned previously, made use of either a SVM classifier or a NB classifier.

To conclude, the algorithm proposed by Kamal (2014) was chosen as the traditional machine learning algorithm and was implemented as one of the final algorithms. The reason for this decision was that the features used by Kamal (2014) were based on both word statistics and dictionaries, the dataset was close to the dataset used in this research and the final F1-score was higher than all other algorithms. Kamal (2014) only used the NB classifier; however, because of the promising results in all the remaining literature, the SVM classifier was used in this research as well.

4.4. Deep Learning Algorithms

Existing studies of sentiment classification are dominated by two main directions: traditional feature-based methods, as was discussed in Section 4.3, and deep learning methods. Traditional feature-based methods extract manually designed features from the text. However, deep learning models have achieved remarkable results in tasks such as computer vision and speech recognition in recent years (Zhanga et al., 2019). Generally speaking, deep learning methods perform slightly more ideally than traditional feature-based methods in terms of classification accuracy on a majority of sentiment classification tasks, but the traditional feature-based methods have advantages with respect to interpretability and computational complexity.

Zhanga et al. (2019) have created a method which combines both deep learning and traditional feature-based methods in order to improve overall performance. The proposed method by (Zhanga et al., 2019) is a three-way enhanced convolutional neural network (**hereinafter** CNN) method. It is divided into three parts, namely accept, reject and delay the decision. These boundaries correspond in binary sentiment classification to the positive or negative class. In order to divide the classification results of CNN into positive, negative and boundary region, a confidence divider is constructed which can measure the confidence values. The results with weak confidence are divided into the boundary region. Another classification model is used to reclassify the instances in the boundary region. This reclassification model needs to be sufficiently different from that of the CNN. Finally, the results are combined.

For text classification of nearly all cases of neural networks, a combination is made between a CNN and a recurrent neural network (**hereinafter** RNN). A standard RNN is a kind of artificial neural network in which connections among the units form a bidirectional cycle, and they perform the same task for each element in the sequence. The RNN technique performs more ideally in sequential tasks, for example, capturing temporal information. The RNN uses a recurrent hidden state whose activation depends on the previous time step, while performing the sequential information and the current state depends on the current input. In this way, the current hidden state makes complete use of past information. Therefore, standard RNNs can handle sequences of variable length and compute sequential data in a dynamic process (Zulqarnain et al., 2020). The long short-term memory (**hereinafter** LSTM) is a special kind of RNN and has largely been applied in NLP. LSTM consists of a gated mechanism and internal cell memory that helps to address the well-known issues relevant to vanishing or exploding gradients. A common LSTM architecture unit is composed of an internal memory cell and three gates in a recurrent connection that help the model to determine how much information can be

passed away and extract more information for every time step (Zulqarnain et al., 2020). The combination between a RNN and a CNN is made because CNNs are able to learn local responses from the temporal or spatial data but lack the ability to learn sequential correlations. In contrast to CNNs, RNNs are specialised in sequential modelling but unable to extract features in a parallel way (Liu and Guo, 2019).

The most simple combination of a CNN and a RNN has been proposed by Hassan and Mahmood (2017). The literature states that using only a CNN has a disadvantage, as the network must have many layers in order to capture long-term dependencies. A RNN is able to capture long-term dependencies with one single layer, particularly with a bidirectional RNN, because each hidden state is computed based on the entire input sequence. The algorithm proposed by Hassan and Mahmood trains a simple CNN with one layer of convolution on top of word vectors that is obtained from an unsupervised neural language model. Then, the algorithm utilises a bidirectional RNN as an alternative for the pooling layers to potentially reduce the loss of detailed local information.

Then Liu and Guo (2019) have proposed another method which combines a RNN and a CNN. For the RNN, a bidirectional long short-term memory (**hereinafter** biLSTM) is chosen because it combines the forward hidden layer and the backward hidden layer, which can access both the preceding and succeeding context. The CNN layer can be used to extract features of the text vector and reduce the dimensions of the vector. Although biLSTM can obtain the contextual information of the text, it is not possible to focus on the important information in the obtained contextual information. For this, an attention mechanism is used which can highlight the important information from the contextual information by setting different weights.

The final and most complex architecture using both CNN and RNN is proposed by Liu et al. (2019). It states that representation learning is a key issue in the field of NLP. Existing models can be roughly divided into the following four types: bag-of-words representation models, sequence representation models, structure representation models and attention-based models. Bag-of-words representation is a statistical-based model which considers the frequency of occurrence of words to distinguish among different texts, but it does not encode the word order and syntactic structures. Sequence representation ignores the structure information of texts, although they consider the order of the words. Structure representation models take the structure information into account. Attention-based models score the input words respectively by the attention mechanism to generate a text representation. A majority of existing structure representation models either learn minimal structure information or rely heavily on the parser, which results in the poor performance of the model. For this, Liu et al. (2019) have proposed a novel model named a hierarchical local and global attention network. It consists of three major components: 1) the local attention part, which generates semantic and structure representations through a CNN layer and a biLSTM layer with local attention; 2) the global attention part, which fuses the semantic and structure representations with global attention; 3) the loss function part, which utilises cross-entropy objective function to train the model.

The only research that has solely used RNNs has been described by Zulqarnain et al.

(2020). It uses gated recurrent units (**hereinafter** GRU), which is another advanced kind of RNN. It is a more simplified version of a LSTM. A GRU contains only two gates, namely an update gate and a reset gate, that handle the flow of information inside the unit without having individual memory cells. Zulqarnain et al. (2020) have proposed an architecture that contains an encoder-based gated recurrent unit with word embedding, which is called an E-TGRU, because it combines an encoder GRU and a two-state GRU. The T-GRU learns to extract forward and backward context features through different time steps. The outputs are provided by the E-GRU model and, finally, followed by a softmax classifier. The T-GRU network contains two directions, one for the positive time direction, namely the forward state, and one for the negative time direction, namely the backward state.

To conclude, the algorithm proposed by Liu and Guo (2019) was chosen as the most promising deep learning algorithm and was implemented as the second SOTA algorithm. The reason for this decision was that it was tested on the SUBJ-dataset, and it performed well, with an accuracy score of 94%, as seen in 4.1. However, it should be noted that, in this research, only accuracy scores were considered, and it is unclear whether a balanced or an imbalanced dataset was used. If the latter was the case, the accuracy score possibly provided a skewed view of the performance.

4.5. Bidirectional Encoder Representations from Transformers (BERT)-based Algorithms

BERT, which stands for bidirectional encoder representations from transformers, was created by Google artificial intelligence (AI) researchers, and it solved the shortage of training data. NLP is a diverse field with many distinct tasks, and a majority of task-specific datasets contain only a few thousand human-labelled training examples. However, modern deep-learning-based NLP models require larger amounts of data. They perform most effectively when trained on millions or billions of annotated training samples. However, these are, in a majority of cases, unavailable. To solve the problem of the absence of training data, BERT was developed, and, with this release, anyone in the world could then train their own SOTA model in about 30 minutes on a single Cloud Tensor processing unit by fine-tuning the pre-trained model to a specific NLP task (Google AI, 2018).

BERT is the first deeply bidirectional and unsupervised language representation that has been pre-trained using only a plain text corpus, namely the entirety of Wikipedia, among others. Pre-trained representations can either be context-free or contextual, and contextual representations can either be unidirectional or bidirectional. Word2vec or GloVe are context-free and generate a single word embedding representation for each word in a vocabulary (Google AI, 2018). Therefore, words with different meanings in different contexts cannot be distinguished. Contextual models instead generate a representation of each word that is based on the other words in the sentence, which solves the problem of words with different meanings in different contexts. In a unidirectional model, only the words prior to the word matter for the representation of the word. However, in a bidirectional model, all words surrounding the word have an influence on the representation of that word. BERT is trained on two different tasks. The first is the

masked language modelling task (**hereinafter** MLM), which forces the BERT model to embed each word based on the surrounding words. The second is the next sentence prediction (**hereinafter** NSP), which forces the model to learn semantic coherence between sentences (de Vries et al., 2019).

Since BERT has been published in 2018, multiple architectures using BERT have been proposed for different tasks. Gao et al. (2019) have proposed an architecture for the task of target-dependent sentiment classification. Target-dependent sentiment predicts the polarity of a tuple (s, t), which consists of a sentence and a target and t as a sub-sequence of s. The goal is to predict the sentiment polarity y of sentence s towards the target t. For the task of target-dependent sentiment classification, Gao et al. (2019) have proposed target-dependent BERT, which takes the output from the target terms, and, therefore, the architecture focuses on those. When there are multiple target terms, a max pooling operation is taken before the data is fed to the next fully-connected layer, as seen in Figure 4.2.

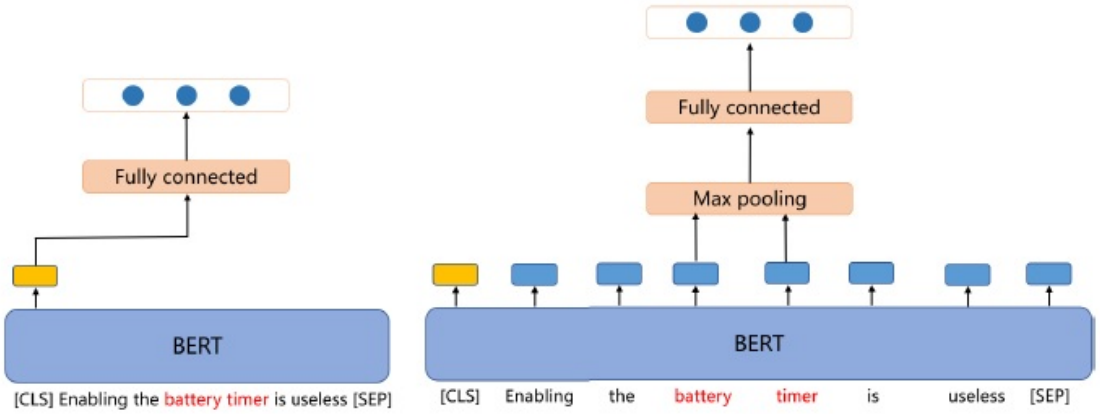

Figure 4.2: The architecture of BERT-FC (left) and TD-BERT (right) (Gao et al., 2019)

To make BERT task-specific, it is important to fine-tune the model. Song (2020) has stated that, if BERTs fine-tuning strategy is adapted to a text classification task, the results will generally be more ideal than classic text classification models for a majority of datasets. To improve performance, a proper strategy combined with BERT is desired. Therefore MIHNet has been proposed by Song (2020), which is designed to achieve the most effective performance by reasonably combining three kinds of information from different dimensions, namely N-grams, sequential and global information. MIHNet is a hierarchical multi-stage process and consists of five layers:

- **Embedding Layer** - responsible for mapping each word to a high-dimensional vector space using BERT. The output of this layer is a sentence of d-dimensional vectors.
- **Contextual Embedding Layer** - a biLSTM is used to model the temporal interactions between words.
- **Highway Network Layer** -the concatenation of the word and contextual embedding vectors are passed to a two-layer highway network. Highway networks are neural networks that allow for unimpeded information flow across several layers on

information highways. The architecture is characterised by the use of gating units which learn to regulate the flow of information through a network (Srivastava et al., 2015). Finally, an attention mechanism is used to find the final representation of the information.

- **Convolution Layer** - a convolution layer is responsible for capturing n-gram information from context words. Deep convolution can capture n-gram information of different sizes without manually setting the convolution kernel size.

- **Output Layer** - the output of the highway network layer is connected to the output of the convolution layer and fed into the softmax layer.

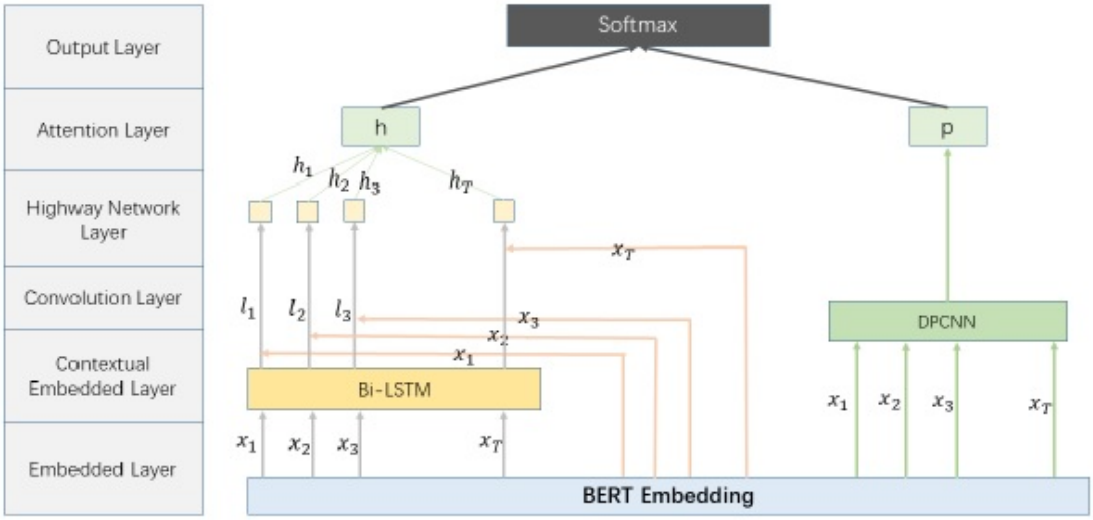

Figure 4.3: Multi-dimension Information Integration using Highway Network (Song, 2020)

However, the success of BERT on NLP tasks has primarily been limited to the English language since the main BERT model is trained on English texts. For other languages, it is possible to either train language-specific models with the same BERT architecture or use the existing multilingual BERT model, which is trained on all Wikipedia pages of 104 different languages, including Dutch. Nonetheless, de Vries et al. (2019) have stated that a monolingual model will probably perform more ideally at tasks in a specific language and that Wikipedia is a specific domain which is not representative of general language use. For this reason, de Vries et al. have developed a similar model to BERT called BERTje. BERTje is based on a large and diverse dataset of 2.4 billion tokens. However, it seemed, instead, that BERT learned inter-sentence coherence, while training on the NSP task, the model apparently learned topic similarity. Because of this, the authors of RoBERTa removed the NSP task from the pre-training process. The developers of ALBERT, instead, implemented a different solution by replacing the NSP task with a sentence-order-prediction task. In sentence-order-prediction, two sentences are either consecutive or swapped. This change has resulted in improved downstream task performance (de Vries et al., 2019).

Delobelle et al. (2020) have proposed a version of RoBERTa called RobBERT, which is pre-trained on the Dutch section of the Open Super-large Crawled ALMAnaCh (**hereinafter** OSCAR) corpus, which is a large multilingual corpus which was obtained by language classification in the Common Crawl corpus. This Dutch corpus has 6.6 billion words totalling 39 GB of text. RobBERT shares its architecture with RoBERTa's base model, which itself is a replication and improvement of BERT because it is only pre-trained on the MLM task rather than the NSP task.

To conclude, the algorithm proposed by Delobelle et al. (2020) was chosen as the most promising BERT-based algorithm and was implemented as the third SOTA algorithm. Because the dataset that was used for this research was in Dutch, it was therefore important that the model be pre-trained on a Dutch dataset as monolingual versions of BERT outperform the multilingual versions. Furthermore, as RobBERT has outperformed BERTje, this seemed to be the appropriate algorithm to use.

4.6. Conclusion on Classification Techniques

In this section, an overview will be provided of the existing algorithms used for text classification in sentiment analysis, objectivity versus subjectivity recognition and polarity analysis. The classification techniques were divided into three different categories, namely traditional machine learning approaches, deep learning approaches and BERT-based approaches. Per category, the most promising algorithm was chosen. These approaches were as follows:

- **Traditional machine learning** approach proposed by Kamal (2014) - an approach using different features such as TF-IDF, PoS, position, opinion indicator seed words and negation. The classifier that was used was a NB classifier.
- **Deep learning** approach proposed by Liu and Guo (2019) - an approach using a CNN in combination with a RNN with attention.
- **BERT** based approach proposed by Delobelle et al. (2020) - an approach using a Dutch pre-trained BERT model only trained on the MLM task, called RobBERT.

In Table 4.1, an overview is provided of all the different classification techniques that were described in this chapter. For all algorithms the F1-score is stated if it was mentioned in the literature. If the F1-score was not mentioned, the accuracy is stated in 4.1. The chosen algorithms are printed in bold.

Table 4.1: Overview Classification Techniques per Category

Category	Classifier	Study	Dataset	Score	Citations	Journal
ML	SVM/NB	(Chong et al., 2014)	Twitter	F1: 55%	23	4th international Conference on Artificial Intelligence with Applications in Engineering and Technology
	NB	**(Kamal, 2014)**	**Reviews**	**F1: 96%**	**26**	**International Journal of Computer Science Issues (IJCSCI), Volume 10 Issue 5**
	SVM	(Montejo-Ráez et al., 2013)	Twitter	F1: 62.8%	147	Elsevier
	SVM	(Sharma and Dey, 2012)	IMDb	Acc: 85.15%	127	Proceedings of the 2012 ACM Research in Applied Computation Symposium
	NB/SVM	(Tripathi and Naganna, 2015)	Reviews	F1: 86%	48	Machine Learning and Applications: An International Journal (MLAIJ) Vol.2, No.2
	NB	(Trupthi et al., 2016)	Reviews	F1: 93%	15	International Conference on Advances in Human Machine Interactions
DL	CNN+ biRNN	(Hassan and Mahmood, 2017)	Reviews	Acc: 89.6%	21	16th IEEE International Conference on Machine Learning and Applications
	3W-CNN	(Zhanga et al., 2019)	SUBJ	Acc: 93.5%	30	Elsevier
	MAN	(Jiang et al., 2020)	Twitter	F1: 70.13%	8	IEEE Internet of Things Journal, Vol. 7, No.4
	biLSTM with attention and CNN	**(Liu and Guo, 2019)**	**SUBJ**	**Acc: 94%**	**46**	**Neurocomputing 337**
	E-TGRU	(Zulqarnain et al., 2020)	IMDb	Acc: 89.37%	2	(IJACSA) International Journal of Advanced Computer Science and Applications, Vol.11, No.1
	HGLAN	(Liu et al., 2019)	SUBJ	Acc: 93.9%	0	IJCNN 2019. International Joint Conference on Neural Networks
BERT	TD-BERT-QA-CON	(Gao et al., 2019)	Twitter	F1: 75.6%	5	IEEE Access
	$BERT_{large}$	(Munikar et al., 2019)	SST-2	Acc: 94.7%	8	2019 Artificial Intelligence for Transforming Business and Society (AITB)
	RobBERT	**(Delobelle et al., 2020)**	**Reviews**	**F1: 86.73%**	**3**	**arXiv**
	MIHNET	(Song, 2020)	IMDb	Acc: 94.91%	0	Journal of Physics: Conference Series
	BERTje	(de Vries et al., 2019)	Reviews	Acc: 93.6%	13	arXiv

5

Rules-Based Redaction of Personal Opinions

In this chapter, an explanation will be provided for how the rule-based classification was developed to automatically recognise personal opinions. This will provide the first part of the answer to the following sub-question that was stated in Chapter 1, namely:

> *Which algorithms can be used for recognising personal opinions at the moment of deliberations?*

5.1. Implementation Rules-based Classification

Starting from the first interviews that were held with Wob-specialists and during the first examinations of previously redacted documents, it became apparent that the ideal way to describe how Wob-specialists executed the redaction process was as a rule-based approach. For this reason, the results obtained from the rule-based approach functioned as a baseline for the remaining algorithms that were tested. As was described in Section 3.1, when the Wob-specialists were asked how they recognised personal opinions, the answer was that personal opinions were recognised by searching for combinations of words such as follows: '`I expect` ...', '`I advise` ...', '`we think`...', '`he suggests`...'. After the first interviews, time was spent to further analyse previously redacted documents. During these sessions, an observation was made that personal opinions could be divided into four different subcategories:

- **Advice** - when a person gives advice to another person. An example could be as follows: `My recommendation would be to [ACTION].`
- **Opinion** - when a person states his or her opinion about a topic to another person. An example could be as follows: `We think [ACTION] is not a good idea.`
- **Suggestion** - when a person suggests a topic to another person. An example could be as follows: `Isn't it sensible to [ACTION]?`
- **Expectation** - when someone states his or her expectation about a topic to another person. An example could be as follows: `I don't expect a good outcome to [INVESTIGATION].`

Based on those categories, different sentence constructions were conceived. For each sentence, a subject or verb was identified with help of PoS. Those had to be the same as the words which would be frequently used to express one of the previously mentioned categories of personal opinions.

5.1.1. Error Analysis

After the first version of rules was completed, the rules were run on different sets of documents which were already redacted. Based on the personal opinions that were missed by the rules or sentences that were found by the rules but not classified by the rules as personal opinions, improvements were made to the rules. Approximately five iterations similar to this took place. In this way certain keywords could be added or removed from the various dictionaries and sometimes new constructions of sentences could be identified in the new documents. Finally, the Wob-specialists as well were asked to use the rules in new Wob-requests to test how the rules performed on new documents. The conclusions from these interviews were that the rules performed well in e-mail conversations. However, they performed less effectively on more official documents. After examining this result in more detail, this could be explained because, within e-mails, officiaries commonly used pronouns within a sentence. This made a sentence a personal opinion. In more official documents such as notes during meetings, officiaries were used to writing shorter sentences, which could be frequently interpreted as facts. An example sentence that has been written in such a note is:

> ‘Had a good meeting, planning/scheduling research is a point of attention, maybe a second opinion research?’

This sentence may not sound like a personal opinions, however that statement was redacted as a personal opinion by the Wob-specialist. In an e-mail, the same sentence would often be written as follows:

> ‘Had a good meeting, we think planning/scheduling research is a point of attention, we are thinking about second opinion research?’

The first example will not be identified as a personal opinion by the rules, however the second sentence would be identified as one. The final set of rules can be found in Appendix B, which are written in Dutch.

5.2. Starting Point - Baseline with the Rules-based Approach

The results of the rule-based classification functioned as the minimum values that the more advanced algorithms had to achieve. The rules were tested on two different test datasets, namely the entire test dataset and a subset, which only contained sentences written in e-mails. The creation of these datasets will be discussed in more detail in Chapter 6. From Table 6.1, it can be noted that the rule-based approach performed similarly to what was expected, namely resulting in a lower score for precision in comparison to the recall-value. Another interesting observation that could be made was that the recognition of personal opinions performed more ideally on the e-mail subset of

the test dataset when the F1-scores were considered. Regarding the recall-values, the results were similar for the entire test dataset and the e-mail subset. However, the precision of the results were more ideal for the e-mail subset.

Table 5.1: Results Rule-based Classification

Test Dataset	**Precision**	**Recall**	**F1**	**Accuracy**
Test	0.33	0.58	0.42	0.59
Mail	0.4	0.57	0.47	0.54

6

Creation of the Datasets

In this chapter, how the dataset was created to train and test the different algorithms and the challenges that were encountered during that process will be explained. This will give an answer to the following sub-question, as was stated in Chapter 1:

> *How should a dataset be obtained to train and subsequently test the chosen algorithms?*

In Section 6.1, the method by which the training dataset was created and the characteristics of the dataset will be discussed. Subsequently, in Section 6.2 how the test dataset was created will be explained.

6.1. Creation of Training Data

The process of creating a dataset that was of sufficient quality to train and test the different algorithms was not straightforward. The first challenge that was encountered was gaining access to data. Clients of ZyLAB stored their data on remote servers. Through a web browser, each officiary who was allowed access to a so-called 'matter', which was the set of documents belonging to a specific Wob-request, was able to change the data. Therefore, it was important to go further than providing individuals with access to matters when absolutely necessary. To gain access to the matters, it was important to convince the Wob-specialists of the added value of this thesis regarding how they handled Wob-requests. Fortunately, several clients had a sufficient amount of confidence in this thesis and therefore provided permission to use their data.

In an ideal situation, the texts that were redacted by the Wob-specialists would be used in the dataset as personal opinions, and the texts that were not redacted would then function as the negative class, which would be all other types of texts excluding personal opinions. Unfortunately, this was not possible because the Wob-specialists redacted the documents within the ZyLAB ONE platform, where the redaction took place by placing rectangles over the text, whose format was changed from a text document to an image. To extract the redacted text from the platform, a match had to be made between the coordinates of the rectangles placed on the image and the text that corresponded with those specific coordinates. However, in the time-frame of this thesis, it was not possible to develop such

a tool.

In order to circumvent the problem of not having a human-labelled training dataset, the Snorkel framework was used, which is a weakly supervised labelling algorithm. Snorkel applies labelling functions, which can be compared to rules, to label training data. However, before it is possible to apply Snorkel labelling functions, a training dataset needs to be created, although it may be unlabelled.

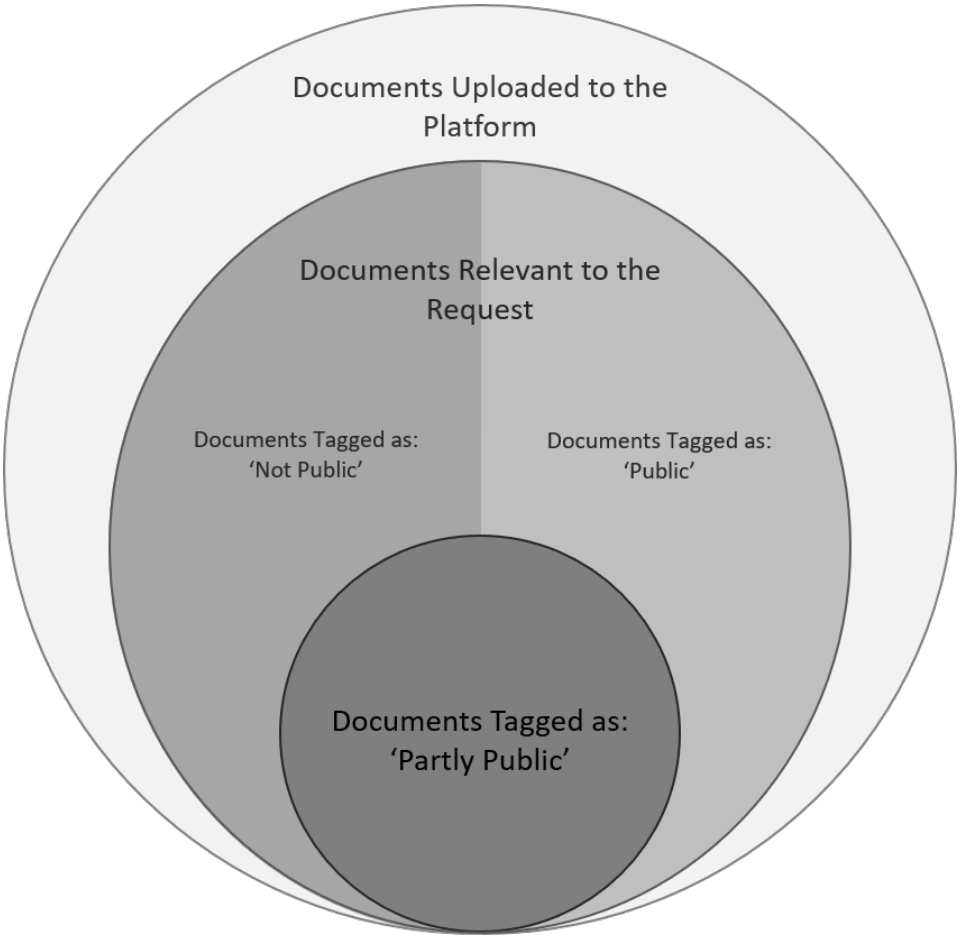

Figure 6.1: Different Sets of Data on the ZyLAB ONE Platform

A variety of media is uploaded to the ZyLAB ONE platform, such as e-mails, PDFs and images. Moreover, from the entire set of uploaded documents, a machine learning algorithm filters for the documents that are relevant to the request. This means that many irrelevant documents are initially uploaded. When all these documents are to be used as training data, that data would contain much noise. The documents relevant to the request can be divided into three different sets. The first set of documents are documents that are already available to the public and have, therefore, been tagged as public'. Then there are documents that are tagged as 'not public', which means that, for a certain reason, the entire document has been considered as sensitive information on one of the grounds of refusal. Finally, the most interesting set of documents are the documents tagged as 'partly public' by the Wob-specialists. With certainty, it can be said that parts of those documents were redacted as a personal opinion while certain parts of those documents were

disclosed to the public.

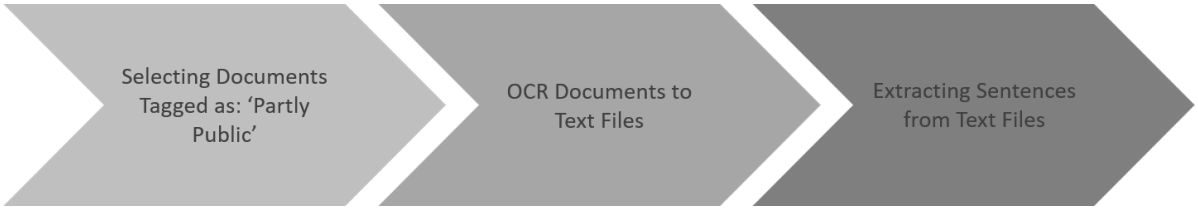

Figure 6.2: Different Steps to Create Training Dataset

Hence, to create a training dataset containing as minimal an amount of noise as possible, only the documents tagged as 'partly public' were used as training data. This was the first step in creating the training dataset, as can be seen in Figure 6.2. Thereafter, the documents that were tagged as 'partly public' were converted from images to text files using optical character recognition (**hereinafter** OCR).). From those text files, sentences were extracted, which functioned as the data points within the training dataset. This dataset contained 8,212 different Dutch sentences.

6.1.1. Data Preprocessing

Text preprocessing takes an input of raw text and returns cleansed tokens. Multiple steps exist in this process, as shown in Figure 6.3. These steps have been described by Anandarajan et al. (2019). These steps standardise data and thereby reduce the number of dimensions in a dataset. However, there is a balance between, on the one hand, retaining information and, on the other hand, reducing complexity. Each step shown in Figure 6.3 removes unnecessary information.

During the first step, a unit is chosen, as seen in Figure 6.3 of data preprocessing. The unit for this research was at the sentence level. For this, entire sentences were extracted from the text files in order to divide useful data from useless data. For this, ZyLAB has a script, which identifies text segments using SegTok. SegTok filters text segments based on segment length and alphanumerical characters. Sentences nonetheless existed from this extraction that were not useful. However, the majority of useless data was filtered out.

In the next step, cases are standardised because, for example, `Dog` and `dog` should be considered the same word. Then numbers, punctuation and special characters are removed with help of regular expressions. In the next step, the stop words, which are words without value for the content and used frequently, are to be removed. However, in this research, several of the stop words were important, such as personal pronouns (`I`, `We`, `He`, etc.). For this reason, stop words were not removed.

Subsequently, the remaining words in the sentence can be lemmatized and stemmed. The goal of lemmatization and stemming is to break down words to their root word. During stemming, suffixes of words are removed. For example, the word `train` has multiple forms: `train, trains, trained, training and trainer` (Anandarajan et al., 2019). Some words, such as `studies`, are stemmed to `studi` and another version of the same word, such as `studying` would be stemmed to `study`. This problem is solved by using

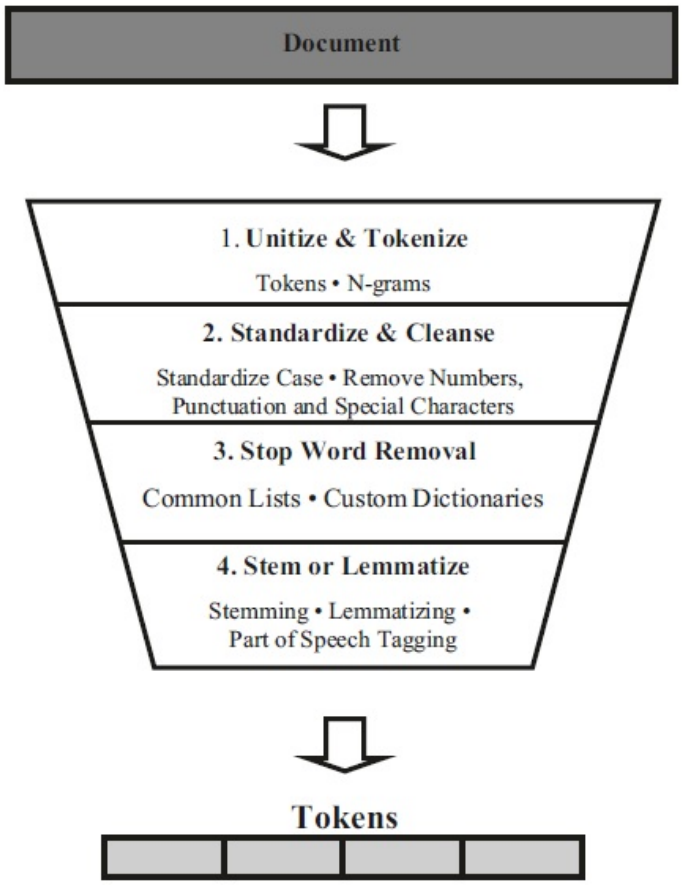

Figure 6.3: Data Preprocessing Process as mentioned by (Anandarajan et al., 2019)

lemmatization, which takes the morphological analysis of words into consideration. During lemmatization, all words are transformed to their most simple form. Therefore, `studies` and `studying` would both be transformed to `Study`. Because lemmatization transforms more words in the same form than stemming, lemmatization was used during this research with the Stanza library created by Stanford NLP Group. It could have been possible that several of the algorithms used in this research would perform more ideally without lemmatization. Therefore, experiments were performed with and without lemmatization. The final steps consisted of deleting duplicate sentences. After this final step, the final preprocessed dataset contained 6,559 different sentences.

6.1.2. Data Labelling

After performing the preprocessing steps, as shown in Figure 6.3, a dataset is without any useless information. Nonetheless, the dataset is not ready to be used because it remains unknown which sentences are personal opinions and which sentences are not. One way of solving this problem is by labelling each sentence manually. However, this was not feasible for the amount of data that was necessary for the majority of algorithms that were used for this research. Another way of solving this problem is by using a framework called Snorkel, which is shown in Figure 6.4.

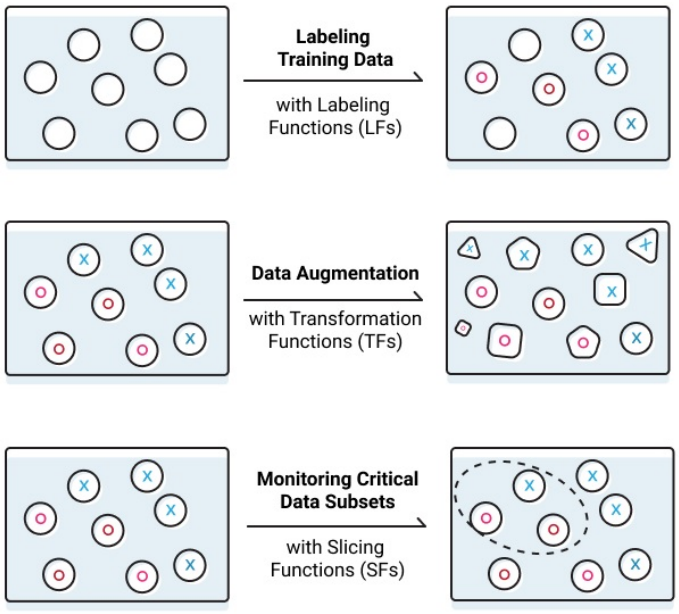

Figure 6.4: The Snorkel Framework (Snorkel, 2019)

By writing labelling functions, a dataset can be noisily labelled (Snorkel, 2019). The rules created for the rule-based algorithm were used for the labelling functions, which was described in detail in Chapter 5. There were two additional labelling functions, which were not initially used for the rule-based classification, namely a subjectivity recogniser within the TextBlob library in Dutch and in English. When a certain sentence had a subjectivity score higher than the threshold value, it was assigned a positive label, which could indicate that it was a personal opinion.

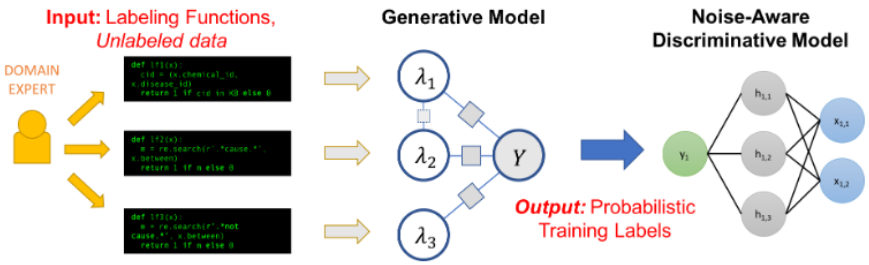

Figure 6.5: Labelling Pipeline in Snorkel (Snorkel, 2019)

In Snorkel, the labelling functions are applied to unlabelled data. A generative model is used to learn the accuracies of the labelling functions without using any labelled data, and weight the output of the model accordingly. The generative model used in this thesis is LabelModel, which takes into account that the different labelling functions should be treated differently. In addition to having varied accuracies and coverages, several labelling functions can be correlated, resulting in certain signals being over-represented. The outputs of the LabelModel were used as training labels to train a classifier which can

generalise beyond the labelling function outputs. This pipeline is shown in Figure 6.5.

Experiments were performed with the different versions of the labelling functions. The rules, as described in Chapter 5, were transformed into English and Dutch labelling functions. The reason why English versions were additionally used was that the PoS taggers from the SpaCy library performed more ideally than the Stanza PoS taggers. The SpaCy library can, for example, tag dependency information, which means that it can recognise if a word is the subject of a sentence. This can be used in labelling functions to check if the subject of the sentence has a pronoun PoS-tag. This was an indication that a sentence could be a personal opinion. However, for the English labelling functions, being able to label the sentences, the Dutch sentences had to be translated to English. While translating a sentence, it was possible for certain language-specific information to be lost. This meant that both the Dutch and English versions were not perfect, but Snorkel states that labelling functions can be noisy – they do not have perfect accuracy and do not have to label each data point. Labelling functions can overlap, which means that they can label the same data point, as well as possibly conflict, which means they can assign different labels to the same data point.

The different sets of labelling functions were tested on the test dataset, which will be described in more detail in Section 6.2. The results of the different sets of labelling functions are shown in Table 6.1. From the results, it could be concluded that only for the English versions of the labelling functions the results were insufficient. The Dutch labelling functions and the combination of the English and Dutch labelling functions performed nearly equally well. However, for the combination of the English and Dutch labelling functions, the F1-score and accuracy performed slightly better. For this reason, the dataset was labelled using both the English and Dutch versions of the labelling functions.

Table 6.1: Results Different Labelling Functions

Labelling Functions	**Precision**	**Recall**	**F1**	**Accuracy**
Dutch	0.313	0.615	0.412	0.553
English	0.3	0.355	0.325	0.62
Dutch & English	0.329	0.58	0.42	0.587

In addition to experimenting with different labelling versions, experiments were executed with degrees of precision of the rules. From this, a labelled training dataset was created which focused on high recall by using more broad labelling functions. Additionally, a labelled training set was created with the more precise labelling functions, which resulted in a higher precision-value but a lower recall-value. The results are shown in Table 6.2. Both versions of the labelled training dataset were used in further experiments.

Table 6.2: Results Labelling Functions Focusing on either Recall or Precision

Labelling Functions	**Precision**	**Recall**	**F1**	**Accuracy**	**Fraction of Personal Opinions**
High Recall Training Dataset	0.33	0.58	0.42	0.587	0.53
High Precision Training Dataset	0.65	0.28	0.39	0.67	0.26

6.1.3. Data Augmentation

Peter Norvig, Director of Research at Google, has said the following: *'We don't have better algorithms, we just have more data.'*. More data outperforms clever algorithms, but more ideal data outperforms more data. Obtaining ideal data is difficult; however, obtaining more data is possible. Data augmentation is a popular technique for increasing the size of labelled training sets by applying class-preserving transformations to create copies of labelled data points, as seen in Figure 6.4. For this thesis, multiple different transformation functions were used in order to increase the labelled training dataset. This was important because, for several of the advanced algorithms, excessively small amounts of data can lead to overfitting. In Table 6.3 the different transformation functions are shown. The sentence used for this table is in a famous Dutch book *De Helaasheid der Dingen* by Dimitri Verhulst:

> *'De vermeende terugkeer van tante Rosie naar Reetveerdegem werd als een aangename schok ervaren in de levens van onze volstrekt nutteloze mannen, waarvan ik er op dat ogenblik een in wording was.'*

In the first column, the name of the transformation function is shown. In the next column, the original text in Dutch is displayed. However, before the sentence could be transformed, the sentence had to be preprocessed. For this reason, the lemmatized version of the original sentences is shown. The words that were changed by the transformation functions are printed in italics. In the final column, the augmented Dutch texts are presented, for which the changed words are shown in italics as well.

Table 6.3: Overview Transformations Functions

Transformation Function	**Original Text Dutch (Lemmatized)**	**Augmented Text Dutch**
Swap Adjectives	de *vermeend* terugkeer van tante Rosie naar Reetveerdegem worden als een aangenaam schok ervaren in de leven van ons volstrekt *nutteloos* mannen waarvan ik er op dat moment een in wording zijn	de *nutteloos* terugkeer van tante Rosie naar Reetveerdegem worden als een aangenaam schok ervaren in de leven van ons volstrekt *vermeend* mannen waarvan ik er op dat moment een in wording zijn
Replace Verb with Synonymn	de vermeend terugkeer van tante Rosie naar Reetveerdegem worden als een aangenaam schok *ervaren* in de leven van ons volstrekt nutteloos mannen waarvan ik er op dat moment een in wording zijn	de vermeend terugkeer van tante Rosie naar Reetveerdegem worden als een aangenaam schok *ondervinden* in de leven van ons volstrekt nutteloos mannen waarvan ik er op dat moment een in wording zijn
Replace Noun with Synonymn	de vermeend terugkeer van tante Rosie naar Reetveerdegem worden als een aangenaam schok ervaren in de leven van ons volstrekt nutteloos mannen waarvan ik er op dat moment een in *wording* zijn	de vermeend terugkeer van tante Rosie naar Reetveerdegem worden als een aangenaam schok ervaren in de leven van ons volstrekt nutteloos mannen waarvan ik er op dat moment een in *ontstaan* zijn
Replace with Antonymn	de vermeend terugkeer van tante Rosie naar Reetveerdegem worden als een *aangenaam* schok ervaren in de leven van ons *volstrekt nutteloos* mannen waarvan ik er op dat moment een in wording zijn	de vermeende terugkeer van tante Rosie naar Reetveerdegem verschil ervaren als een *onaangename* schok in de levens van ons *deels nutteloze* mannen waarvan ik er op dat moment een in wording ben
Replace with Synonymn with BERT Multilingual	de vermeend terugkeer van tante *Rosie* naar Reetveerdegem worden als een aangenaam schok *ervaren in de leven* van ons volstrekt nutteloos mannen *waarvan ik er op dat moment* een in wording zijn	de vermeend terugkeer van tante *esther* naar reetveerdegem worden als een aangenaam schok er al *in de leven van ons volstrekt* nutteloos mannen *zien ik er op dat* moment een in wording

After applying the transformation functions, the training dataset increased from 8,212 different sentences to 37,458 sentences. For the remaining experiments, tests were executed with both the augmented and the unaugmented dataset.

6.1.4. Data Slicing

The final step in the Snorkel framework is called data slicing, as can be seen in Figure 6.4. Traditional machine learning applications optimise for overall quality. This can be overly simplified. It can be possible that the model performs well, overall; however, on certain critical parts of the data, it performs inadequately (Snorkel, 2019). However, currently, in our training data, there were no critical data slices identified for this dataset; therefore, this function of the Snorkel framework was not used for this research.

6.2. Test Dataset

The datasets that were described in Section 6.1 were used as training data. However, for testing, a different dataset had to be used. For the training data, the documents were converted to text files with help of OCR, from which sentences were extracted. The disadvantage of the way this training set was created was that the original redaction information of the Wob-specialists was lost. However, this was necessary in order to create a sufficiently large training dataset. In contrast, this amount of data was not necessary for the test dataset. Therefore, in order to create a test dataset, a different set of documents than the documents used for the training dataset was used to create a test dataset. This prevented sentences from appearing in both the training dataset and the test dataset, which would have caused bias in the results.

From all the documents that were present in the set of documents, only the documents which were tagged as 'partly public' were selected, similar to how the training dataset was created. The reason for this division in the test dataset was that, for those documents, it was certain that, for each sentence, an assessment was made of whether the sentence was a personal opinion or not. This was in contrast to the documents that were tagged 'public', where it was possible that there were personal opinions but possible that the documents did not fall under internal deliberation criteria. In such a case it is not allowed to redact personal opinions. This could occur when e-mail conversations among citizens and officiaries were relevant to the Wob-request because those were not a part of internal deliberations. This did not mean that those documents could not contain personal opinions. The documents that were tagged 'not public' contained a high number of personal opinions to the point that the remaining text would not provide a sufficient amount of information when it would be disclosed to the public and would only lead to confusion. From these documents it additionally was uncertain which sentences could be considered personal opinions and which sentences could not.

It is important to realise that officials were not under the obligation to redact a personal opinion, only in the event that the Wob-specialist considered that it would be harmful to the relevant officiary who wrote the sentences or the governmental institution for which the officiary worked if it were published. This meant that certain sentences that were not redacted could nonetheless be personal opinions but were left unredacted due to the

content. However, no background information was available on whether the redactions of the Wob-specialists were followed precisely for the test dataset. This meant that if a sentence was redacted by the Wob-specialist, it was labelled as a personal opinion in the test dataset, and if a sentence was not redacted, it was labelled as not being a personal opinion. This resulted in a test dataset containing 732 sentences, from which 192 sentences were classified as personal opinions, which was equal to 26.2%. In Figure 6.6 a histogram is shown where the distribution of the lengths of words of the different sentences is shown. In Table 6.4, a few statistics are shown concerning the sentences in the test dataset. These statistics could be used for parameter settings for the various algorithms.

Table 6.4: Statistics Sentences Total Test Dataset

Statistic	**Value**
Average Length	15.08
Standard Deviation	9.22
Maximum Length	62
Minimum Length	2

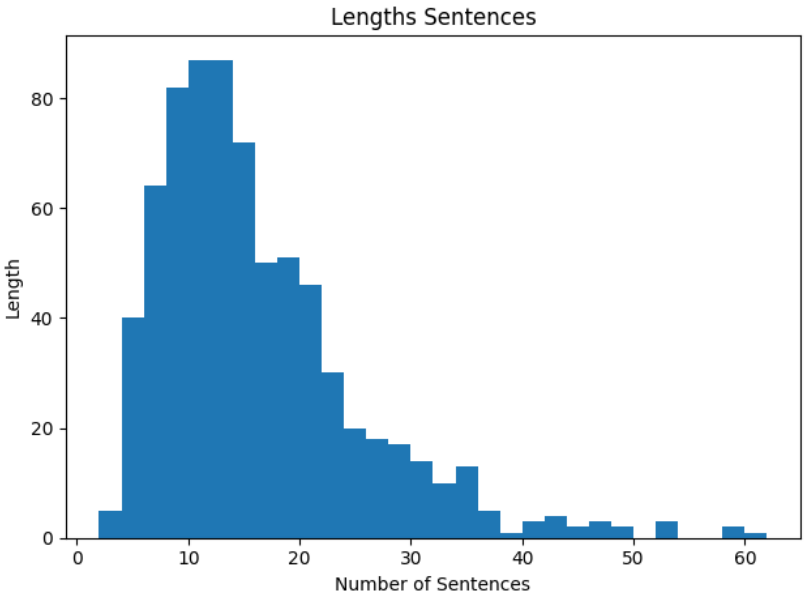

Figure 6.6: Lengths of Sentences Total Test Dataset

6.2.1. E-mail Subset of Total Test Dataset

During the interviews that were held with the Wob-specialists, it became apparent that, for them, redacting personal opinions written in e-mails was simpler than recognising personal opinions written in other types of documents, such as notes from meetings or reports. In the latter, personal opinions were frequently stated factually, which made them more difficult to recognise. This was not the case for personal opinions in e-mails. Here, individuals more frequently used combinations of words such as ‘`I expect ...`’, ‘`I`

`advise ...'`, `'we think...'`, `'he expects...'`. Experiments were performed to investigate whether this additionally held true for the different algorithms. As can be seen from Figure 6.7 and Table 6.5, the characteristics of the e-mail test dataset were similar to the characteristics of the total test dataset. In the remainder of this report, the e-mail subset of the test set will be referred to as the e-mail test dataset, and the entire test dataset will be referred to as the test dataset.

Table 6.5: Statistics Sentences Mail Test Dataset

Statistic	**Value**
Average Length	15.17
Standard Deviation	9.03
Maximum Length	62
Minimum Length	2

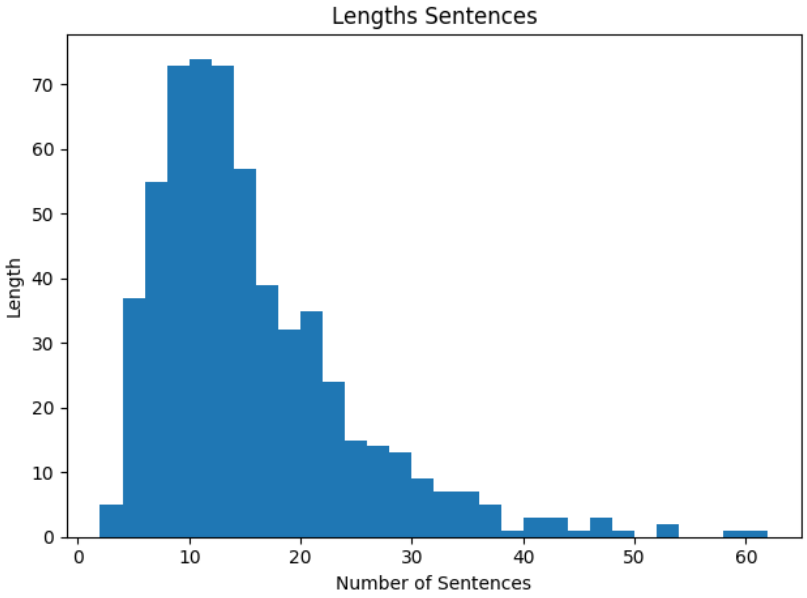

Figure 6.7: Lengths Sentences Mail Test Dataset

6.3. Conclusion Datasets

In this chapter, how the different datasets were developed was explained. For the training data, documents that were tagged as 'partly public' were extracted from the ZyLAB ONE platform, and, with help of OCR, these documents were transformed into text files. From those text files, the sentences were extracted and preprocessed, which functioned as the data points of the training data. While extracting the documents from the ZyLAB ONE platform, the redaction information was lost. This is why, with help of the Snorkel framework, the sentences were labelled with labelling functions, which were based on the rules created for the rule-based approach. Both precision-focused and recall-focused labelling functions were applied. To create more data, the Snorkel framework was used once more. However, this second time, transformation functions were used to augment

the data. Based on the previously mentioned steps, multiple versions of the training dataset were created, which can be seen in Table 6.6.

Table 6.6: Overview Training Datasets

Training Dataset	Recall/Precision	Lemmatization	Augmentation	Size of the Dataset	Number of Personal Opinions
1	Recall	Yes	Yes	37,457	52.6%
2	Recall	Yes	No	6,559	52.6%
3	Recall	No	Yes	44,606	49.8%
4	Recall	No	No	8,147	49.8%
5	Precision	Yes	Yes	44,613	12.8%
6	Precision	Yes	No	6,559	12.8%
7	Precision	No	Yes	47,076	8.4%
8	Precision	No	No	8,147	8.4%

Combining these different variables, which are stated in Table 6.7, led to eight different experiments. These outcomes were ranked per classification algorithm to research which combination showed ideal results for the recognition of personal opinions within internal deliberations. Per the classification technique, the most promising results will be discussed in the following sections. However, an extensive overview of the results for all the experiments are described in Appendix C, Appendix D and Appendix E, for the traditional machine learning, deep learning and BERT-based classification, respectively.

Table 6.7: Used Variables for Creating the Training Data

Variable	Value 1	Value 2	Explanation
Lemmatization	Yes	No	Section 6.1.1
Labelling Function	Focused on Recall	Focused on Precision	Section 6.1.2
Augmentation	Yes	No	Section 6.1.3

Similarly, for the test dataset, only documents tagged as ‘partly public’ were used but relevant to a different administrative matter. In this way, no overlap existed between the training and test datasets. In contrast to the training datasets, it was important to preserve the original redaction information. To achieve this, the sentences were manually rewritten. Sentences which were redacted by the Wob-specialist were used as the personal opinions in the test dataset. Sentences that were not redacted and therefore disclosed to the public were used as the non-personal opinions portion of the test dataset. An overview of the test datasets is shown in Table 6.8

Table 6.8: Overview Test Datasets

Training Dataset	Size of the Dataset	Percentage of Personal Opinions
Test	732	26%
Mail	591	29%

7

Traditional Machine Learning

In this chapter, for the traditional machine learning classification algorithm that was discussed in Chapter 4, the implementation and the results will be discussed. Multiple experiments were executed combining different training datasets that were created with certain combinations of variables and various parameter settings. This will provide the second part of the answer to the following sub-question that was stated in Chapter 1:

> *How should the chosen algorithms be implemented, and what combination of variables and parameter settings will lead to the most effective results for recognising personal opinions?*

7.1. Traditional Machine Learning Implementation

The first more advanced algorithm that was implemented was a traditional machine learning algorithm that has been described by Kamal (2014). The algorithm proposed by Kamal has been developed to recognise subjectivity and objectivity in sentences. The dataset to train and test the classifier was the SUBJ-dataset. Kamal has proposed splitting the sentences into unigrams. Per unigram, a feature vector was created which consisted of the TF-IDF values, PoS tags, opinion indicator seed words and negation features. These are described in more detail in Table 7.1. Additionally, all unigrams that were present in a subjective sentence were classified as subjective. The feature vectors for every unigram were used to train a NB classifier. Thereafter, the unigrams in the test dataset were classified. When at least one unigram in a sentence was classified as subjective, the entire sentence was classified as subjective. The algorithm proposed by Kamal (2014) is shown schematically in Figure 7.1.

Several adjustments were necessary to apply the same algorithm proposed by Kamal to the problem of automatically recognising personal opinions within internal deliberations. Two features could be implemented equally to how they were implemented by Kamal, these being the TF-IDF and position features because these features were not language-dependent. However, adjustments were necessary for the PoS, opinion indicator seed words and negations features because all three were language-dependent features and had to be implemented differently than was described by Kamal (2014). Kamal has not specified what kind of library was used to create the PoS-tags; however, it can be said

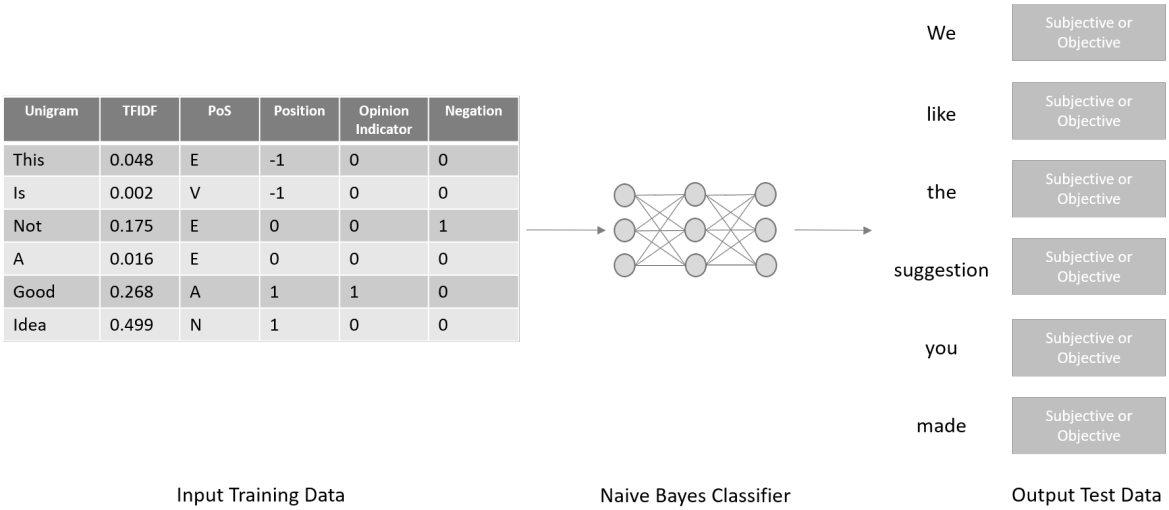

Figure 7.1: Traditional Machine Learning Algorithm proposed by Kamal (2014)

with certainty that an English version was used. This meant that it did not matter which library was used because a Dutch version had to be found. For the Dutch PoS tagging, the Stanza library was used, which has been created by the Stanford NLP Group. For the opinion indicator seed words and the negations, Kamal has not specified what dictionary he used, and, therefore, a new dictionary was created for the Dutch version.

Table 7.1: Features proposed by Kamal (2014)

Features	**Explanation**
TF-IDF	It combines the frequency of a unigram in a certain document with its occurrence in the whole corpus. It is calculated using the following equation: $TF-IDF = \frac{f}{s} * (-\log_2 \frac{N_f}{C})$, where s is the size of the document in terms of words, N_f is the number of documents in the corpus containing the unigram and C is the total number of documents in the corpus.
Position	It determines the position of the occurrence of the candidate unigram in a certain sentence. Occasionally, the position of the unigram plays an important role in deciding sentence subjectivity. The position attribute is set to -1, 0 and 1 in case the candidate unigram occurs at the beginning, in-between and end respectively of the enclosing sentence.
PoS	In this feature, the different types of parts of speech are used to perform the classification. It seems for example that adjectives are a ideal indicator of opinion and certain nouns and verbs can be used for subjectivity determination as well. The feature value is set to either A, D, N, V and E in case candidate is adjective, adverb, noun, verb or any other respectively.
Opinion Indicator Seed Word	Opinion indicator seed words are commonly used for expressing positive or negative sentiment regarding product features or services and can be used as a useful indicator for subjectivity determination, for example, a set of positive seed words (amazing, awesome, beautiful, decent, nice, excellent, good) and a set of negative seed words (bad, bulky, expensive, faulty, horrible, poor, stupid). The feature value was set to 1 in case a candidate unigram belonged to either one of the seed words.
Negation	The presence of negation was additionally treated as an important clue for subjectivity detection. In case the candidate unigram was a negation word, the feature value was set to 1 and otherwise set to 0.

After testing the aforementioned implementation, it was apparent that the current setup, which worked for subjectivity versus objectivity classification, did not perform well on the task of recognising personal opinions. After fitting both the SVM and NB model on the training data and evaluating it on the test dataset, all sentences in the test dataset were classified as personal opinions. For each sentence in the test dataset, at least one word was classified as subjective, and, therefore, all sentences were classified as personal opinions.

Table 7.2: Information Gain per Feature described by Kamal (2014)

Features	Information Gain
PoS	0.10364911
TF-IDF	0.02714459
Negation	0.02082773
Opinion Indicator Seed Word	0.00113212
Position	0.00017621

The experiments performed by Kamal (2014) showed that most information gains came from the TF-IDF and PoS features, as can be seen in Table 7.2. The adjustment to the original implementation was that, instead, not all unigrams in the sentence were classified separately. In the adjusted algorithm, the TF-IDF features were calculated for the entire sentence. In addition to the TF-IDF values of the unigrams, the TF-IDF values were calculated for multiple N-grams as well. The different possible N-grams are shown in Table 7.3 for the following example sentence:

```
The dog walks in the park
```

Table 7.3: Different N-grams

unigrams	bigrams	trigrams	four-grams	five-grams
the	the dog	the dog walks	the dog walks in	the dog walks in the
dog	dog walks	dog walks in	dog walks in the	dog walks in the park
walks	walks in	walks in the	walks in the park	
in	in the	in the park		
the	the park			
park				

In addition to the TF-IDF values for different N-grams, experiments were executed when, for the entire sentence, the PoS values were added in the feature vector. However, this had no influence on the results and was, therefore, not used in the final implementation.

7.2. Traditional Machine Learning Classification Results

In this section, the outcomes resulting from a traditional machine learning classification based on the approach proposed by Kamal (2014) will be discussed. For all combinations of variables, which are mentioned in Table 6.7, experiments were executed. In addition to the N-gram parameter setting, multiple settings were experimented with for the maximum document frequency parameter. The maximum document frequency parameter filtered out terms when building the vocabulary that had a document frequency strictly higher than the given threshold, additionally known as corpus-specific stop words. The following values for the parameters and the following classifiers were used in the experiments:

- **N-gram ranges** - 1 to 3, 1 to 5 and 1 to 7
- **Maximum Document Frequency thresholds** - 0.1, 0.3, 0.5 and 0.7
- **Classifiers** - NB and SVM, with kernels linear, polynomial with degree 2, polynomial with degree 3, radial basis function (**hereinafter** RBF) and sigmoid

For each combination of N-gram range, maximum document frequency threshold and classifier, the most ideal results are shown in Appendix C. From the experiments shown in Appendix C the most effective results for each combination of the variables are shown in Table 7.4 for the test dataset and in Table 7.5 for the e-mail test dataset. The results will be discussed separately in the following sections.

Table 7.4: Ranking Results Traditional Machine Learning - Test

Ranking	Variables	Precision	Recall	F1	Accuracy
1	High Recall - Not Lemmatized - Not Augmented	0.35	0.87	0.5	0.54
2	High Precision - Lemmatized - Augmented	0.39	0.68	0.5	0.64
3	High Precision - Lemmatized - Not Augmented	0.42	0.64	0.5	0.67
4	High Recall - Not Lemmatized - Augmented	0.31	0.83	0.46	0.48
5	High Recall - Lemmatized - Not Augmented	0.32	0.79	0.46	0.52
6	High Recall - Lemmatized - Augmented	0.32	0.73	0.45	0.53
7	High Precision - Not Lemmatized - Not Augmented	0.45	0.42	0.44	0.71
8	High Precision - Not Lemmatized - Augmented	0.43	0.35	0.38	0.7

7.2.1. Traditional Machine Learning Test Dataset Results

From Table 7.4, it can be noted that similar results regarding the F1 value could be achieved with labelling functions either focused on high recall or high precision. When considering the results for the recall metric, the labelling functions focused on recall performed more ideally than the labelling functions focused on precision. When focused on the precision metric, the high precision labelling functions performed more ideally. This resulted in similar F1 values for both high-recall labelling functions and high-precision labelling functions.

When high-recall labelling functions were used, the un-lemmatized versions of the training data achieved higher results in recall in comparison to when the training data was lemmatized, namely ranking 1 in comparison to ranking 5 and ranking 4 in comparison to ranking 6. In contrast, when labelling functions focused on high precision were used, the lemmatized training data achieved higher scores on recall. The precision-values were lower when the lemmatized version of the training data was used; however, the increase in recall was more than the decrease in precision. For this reason, the F1 scores were higher for the lemmatized version of the training data when high precision labelling functions were used, namely ranking 2 in comparison to ranking 8 and ranking 3 in comparison to ranking 7.

The final observation that could be made was that augmentation for the traditional machine learning algorithm did not improve the results. The results were similar or worse regarding the F1 values for the augmented training dataset compared to non-augmented training data. This was most evident for the combination of ‘high recall’ and ‘not lemmatized’, namely ranking 1 in comparison to ranking 4. Additionally, when regarding ‘high precision’ and ‘not lemmatized’, the results were worse for the augmented training dataset, namely ranking 7 in comparison to ranking 8. This can be attributed to the fact that traditional machine learning classification algorithms do not require a considerable amount of data, due to the its simple structure compared to the more complex structure of

a neural network.

Ranked at the first place and, therefore, able to be considered the ideal result was when high-recall labelling functions were used to label the training data and the training data was neither lemmatized nor augmented. The parameter settings for this result were as follows:

- **N-gram range** - 1 to 3
- **Maximum document frequency threshold** - 0.1
- **Classifier** - SVM, with a linear kernel

Compared to the baseline, the traditional machine learning classification performed more ideally regarding the F1-values, namely with an increase of 8%. The greatest difference could be seen in the increase in recall, namely with an increase of 29%. Unfortunately, the increase in precision was small, that being an increase of 2%, which meant that, for the traditional machine learning algorithm as well, the precision remained low. In Figure 7.2, the precision-recall curve (**hereinafter** PR-curve), which shows the trade-off between recall and precision for different decision boundaries, is displayed for the most effectively performing setting of the variables. It seemed that when the recall was decreased by increasing the threshold of the decision boundary, the precision did not increase. For this reason, no further changes were made to the decision boundary.

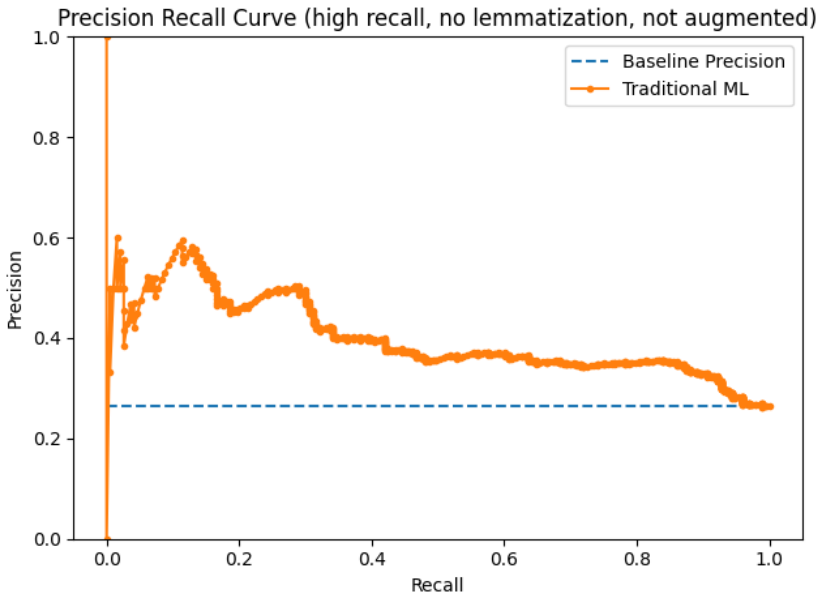

Figure 7.2: Precision-recall (PR) Curve Traditional ML - Test

7.2.2. Traditional Machine Learning E-mail Dataset Results

In Table 7.5, the results are shown for the e-mail test dataset. Similarly to the test dataset, the high-recall labelling functions generally performed more ideally than the high precision labelling functions, particularly for the recall results. Additionally similarly to

Table 7.5: Ranking Results Traditional Machine Learning - Mail

Ranking	Variables	Precision	Recall	F1	Accuracy
1	High Recall - Lemmatized - Not Augmented	0.44	0.84	0.57	0.55
2	High Recall - Lemmatized - Augmented	0.41	0.87	0.56	0.5
3	High Precision - Lemmatized - Not Augmented	0.5	0.65	0.56	0.64
4	High Recall - Not Lemmatized - Not Augmented	0.39	0.88	0.54	0.55
5	High Recall - Not Lemmatized - Augmented	0.35	0.84	0.5	0.49
6	High Precision - Lemmatized - Augmented	0.39	0.68	0.5	0.64
7	High Precision - Not Lemmatized - Not Augmented	0.47	0.44	0.45	0.68
8	High Precision - Not Lemmatized - Augmented	0.45	0.36	0.4	0.68

the test dataset, augmentation worsened the results for all combinations of labelling functions and either un-lemmatized and lemmatized versions of the training data. In contrast to the test dataset, lemmatization showed more ideal results than without lemmatization, particularly in the values for precision.

Ranked at the first place and, therefore, able to be considered the ideal result, was when high-recall labelling functions were used to label the training data and when the training data was lemmatized but not augmented. The parameter settings for this result were as follows:

- **N-gram range** - 1 to 5
- **Maximum document frequency threshold** - 0.3
- **Classifier** - SVM, with a RBF kernel

Compared to the baseline, the traditional machine learning classification performed more ideally when considering the F1-values, namely with an increase of 10%. The greatest difference could be seen in the increase in recall, that being an increase of 27%. Unfortunately, the increase in precision was small, that being an increase of 4%, which meant that, for the traditional machine learning algorithm as well, the precision remained low. The PR-curve in Figure 7.3 shows that increasing the threshold of the decision boundary showed a minimal increase in precision which was, however, insufficient in comparison to the decrease in recall. Therefore, similar to the test dataset, it was not beneficial to the result to change the threshold of the decision boundary.

In Table 7.6 the final results of the traditional machine learning classification of both the test and e-mail test datasets are shown. Regarding the F1-scores, the results for the e-mail test dataset were more ideal than the test dataset due to the 9% increase in precision of the model. However, the recall for the e-mail dataset was slightly lower than the test dataset.

Table 7.6: Comparison Final Results Test and Mail for Traditional Machine Learning

Test Dataset	Precision	Recall	F1	Accuracy
TEST	0.35	0.87	0.5	0.54
MAIL	0.44	0.84	0.57	0.55

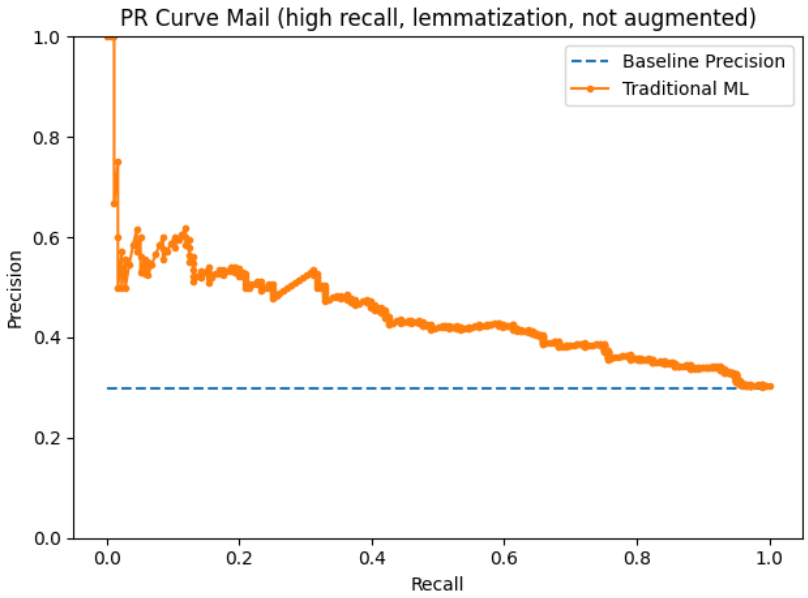

Figure 7.3: PR-Curve Traditional Machine Learning - Mail

8

Deep Learning Classification

In this chapter, for the deep learning classification algorithm that was presented in Chapter 4, the implementation and the results will be discussed. Multiple experiments were executed combining different training datasets that were created with certain combinations of variables and different parameter settings. This will provide the third part of the answer to the following sub-question that was stated in Chapter 1:

> *How should the chosen algorithms be implemented, and what combination of variables and parameter settings will lead to the most effective results for recognising personal opinions?*

8.1. Deep Learning Classification Implementation

The second more advanced algorithm that was implemented was a deep learning algorithm that has been proposed by Liu and Guo (2019). Liu and Guo have mentioned that, for text classification, a majority of the studies with deep learning methods can be divided into two parts. The first part is learning word vector representations through neural language models, and the second part is performing compositions over the learned word vectors for classifications. In a majority of cases, two versions of deep learning models are used in text classification: CNNs and RNNs. CNNs are able to learn local responses from the temporal or spatial data but lack the ability to learn sequential correlations. On the other hand, RNNs are specialised for sequential modelling but are unable to extract features in a parallel manner. For long data sequences, traditional RNNs cause an exploding and vanishing state against the RNN its gradient, which is solved by using LSTMs. Moreover, LSTMs can capture long-term dependencies. By using biLSTMs, the forward hidden layer and the backward hidden layer are combined, which can access both the preceding and succeeding contexts (Liu and Guo, 2019). The architecture that has been proposed by Liu and Guo (2019) is shown in Figure 8.1. All the different components of the network that have been proposed by Liu and Guo (2019), will be discussed in the following sections.

8.1.1. Word Embedding

The first decision that needs to be made is the choice for the word embedding. Traditional word representations face a problem, namely losing the word order. Distributed

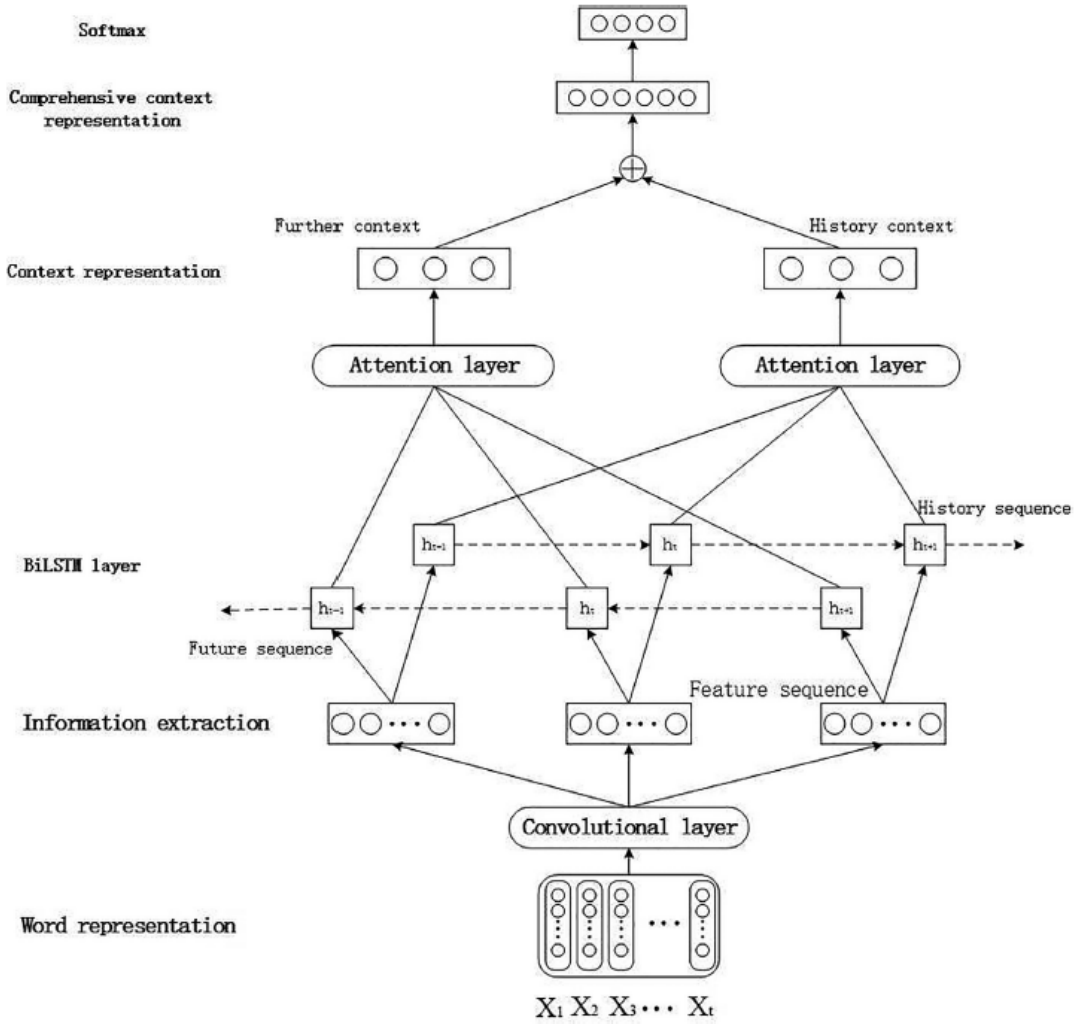

Figure 8.1: Neural Network Proposed by Liu and Guo (2019)

representations of word embeddings are more suitable and powerful than traditional word representations. There are multiple ready-to-use word embedding matrices that can be used. In the network proposed by Liu and Guo, the Word2Vec method proposed by Mikolov et al. (2013) is used with a dimensionality of 300 for each word. However, this version of Word2Vec is in English. Fortunately, a similar dictionary can be found for the Dutch language which has been created by Tulkens et al. (2016). The embeddings were trained on various Dutch datasets, such as Roularta, Wikipedia and Sonar500. There was a choice between 320- and 160-dimensional embeddings instead of the 300 proposed by Liu and Guo. From the experiments performed by Tulkens et al., it seemed that the 320-dimensional embedding outperformed the 160-dimensional embedding in nearly all cases (Tulkens et al., 2016). For this reason, in this research, the dimensionality of 320 was used, which was, additionally, closest to what has been proposed by Liu and Guo.

8.1.2. Convolutional Layer

The first layer is a convolutional layer. For text classification, vector representations are generally high-dimensional vectors. The high-dimensional vector as an input of LSTM

causes a sharp increase in the network parameters and makes the network difficult to optimise. The convolutional layer can extract the features while reducing the dimensionality of data. The convolutional operation is conducted in one dimension, with 100 filters and with window size three. As the filter moves forward, many sequences are generated that capture the syntactic and semantic features. The activation function used for this layer is a rectified linear unit, which can improve learning dynamics of the network and significantly reduce the number of iterations required for convergence (Liu and Guo, 2019).

8.1.3. Bidirectional Long Short-Term Memory and Attention Layers

Text classification is similar to the processing of sequential information. However, the feature sequences, which are obtained by the convolutional layer, no longer contain sequence information. BiLSTM are specialised in sequence modelling and can further extract the contextual information from the feature sequences. The effect of a BiLSTM is building the text-level word vector representation (Liu and Guo, 2019). Different words have different contributions to the sentiment of a sentence, and assigning different weights to words is a common way of solving this problem. An attention mechanism is designed to enhance the understanding of sentiment of the entire text by assigning those different weights to words. Attention mechanisms can focus on the features of the keywords to reduce the impact of non-keywords on the text sentiment, and it is considered as a fully-connected layer and a softmax function, as seen in Figure 8.1.

After performing experiments with the aforementioned implementation, it became apparent that the current setup did not perform as expected for the task of recognising personal opinions. When a network similar to what has been proposed by Liu and Guo (2019) was implemented to recognise personal opinions, the results shown in Table 8.1 were obtained. The results were not ideal compared to the results mentioned by Liu and Guo (2019), where the accuracy was equal to 94% on the SUBJ-dataset. For this reason, in addition to the network which contained a CNN, a biLSTM and an attention layer, networks consisting of only a CNN or biLSTM and a combination of CNN and biLSTM were experimented with as well. Apart from experimenting with the different the neural network, experiments were done with the parameters of the network in order to find the suitable parameter settings for this problem, such as amount of nodes per neural layer.

Table 8.1: Initial Results Original Setup Deep Learning

Neural Network	**Precision**	**Recall**	**F1**	**Accuracy**
CNN + biLSTM + Attention	0.29	0.72	0.42	0.43

Alongside experimenting with variations of the different layers and parameters, other adjustments were made. From the validation loss, it became apparent that the network was overfitting on the training data. To prevent the network from overfitting, extra dropout layers and regularizers were added in the LSTM layers. Furthermore, early stopping constraints were implemented when the validation loss increased after three sessions. The learning rate was set to its 0.001 and the number of epochs was set to 20.

8.2. Deep Learning Classification Results

In this section, the results of the deep learning classification, based on the approach proposed by Liu and Guo (2019), will be discussed. For all combinations of variables, which are mentioned in Table 6.7, experiments were performed. For each combination of variables, experiments were performed with different parameters. For all the different layers, experiments were executed with different numbers of hidden nodes. Another parameter that was experimented with was the maximum number of features that were to be used, which was the number of most frequently used terms that were used for the embedding of the sentences. Finally, the maximum sentence length was experimented with as well. The following values for the parameters were tested in the experiments:

- **CNN Layer** - 200, 100 and 50 nodes
- **biLSTM Layer** - 200, 100 and 50 nodes
- **Attention Layer** - 200, 100 and 50 nodes
- **Maximum Number of Features** - 10,000 and 5,000
- **Maximum Sentence Length** - 40 and 20

Not all combinations of all the different parameter settings are shown in Appendix D, as the number of possibilities was excessively large to report. Therefore, for each combination of neural layers, the most effective results are stated in Appendix D. From the experiments shown in Appendix D, the most effective results for each combination of the variables, which are mentioned in Table 6.7, are shown in Table 8.2 for the test dataset and in Table 8.3 for the e-mail test dataset. The results will be discussed separately in the following sections.

Table 8.2: Ranking Results Deep Learning - Test

Ranking	**Variables**	**Precision**	**Recall**	**F1**	**Accuracy**
1	High Recall - Not Lemmatized - Not Augmented	0.34	0.76	0.47	0.54
2	High Recall - Lemmatized - Augmented	0.31	0.74	0.44	0.51
3	High Recall - Not Lemmatized - Augmented	0.32	0.66	0.43	0.55
4	High Recall - Lemmatized - Not Augmented	0.33	0.62	0.43	0.58
5	High Precision - Lemmatized - Augmented	0.48	0.35	0.41	0.73
6	High Precision - Lemmatized - Not Augmented	0.46	0.35	0.39	0.72
7	High Precision - Not Lemmatized - Not Augmented	0.48	0.22	0.3	0.73
8	High Precision - Not Lemmatized - Augmented	0.43	0.17	0.24	0.72

8.2.1. Deep Learning Classification Test Dataset Results

From Table 8.2, it can be noted that, in all cases, the labelling functions focused on high recall outperformed the results achieved with the labelling functions focused on high precision. The differences varied by a few percent, namely ranking 3 compared to ranking 5, to a difference of over 20%, namely ranking 1 in comparison ranking 8.

It was notable that, for the high-recall labelling functions, the un-lemmatized version of

the training data outperformed the lemmatized version. However, for the high precision labelling functions, this effect was reversed, as the lemmatized version of the training data outperformed the un-lemmatized version.

In nearly all cases, the augmented version of the training data outperformed the non-augmented version of the training data. This could be observed when regarding ranking 1 in comparison to ranking 2, ranking 3 in comparison to ranking 4 and ranking 5 in comparison to ranking 6. Only for the high precision labelling functions and the un-lemmatized version of the training data did the unaugmented training data outperform the augmented training data.

Ranked in first place and, therefore, able to be considered the ideal result, was when high-recall labelling functions were used to label the training data and when the training data was neither lemmatized nor augmented. The ideal result was obtained with a neural network consisting of only a CNN layer. The parameter settings for this result were as follows:

- **CNN Layer** - 200 nodes
- **Maximum Number of Features** - 5,000
- **Maximum Sentence Length** - 25

Compared to the baseline, the deep learning classification for the test dataset performed more ideally regarding the F1-values, namely with an increase of 6%. The greatest difference could be seen in the increase in recall, namely an increase of 11%. Unfortunately, the increase in precision was small, namely an increase of 4%, which meant that, for the deep learning algorithm as well, the precision remained low.

The PR-curve in Figure 8.3 shows that increasing the threshold of the decision boundary showed little increase in precision, however not enough in comparison to the decrease in recall. Therefore, it was not beneficial to the result to change the threshold of the decision boundary.

Table 8.3: Ranking Results Deep Learning - Mail

Ranking	**Variables**	**Precision**	**Recall**	**F1**	**Accuracy**
1	High Recall - Lemmatized - Augmented	0.44	0.68	0.53	0.57
2	High Recall - Not Lemmatized - Not Augmented	0.39	0.73	0.51	0.58
3	High Recall - Lemmatized - Not Augmented	0.39	0.66	0.49	0.5
4	High Recall - Not Lemmatized - Augmented	0.33	0.75	0.46	0.53
5	High Precision - Lemmatized - Not Augmented	0.45	0.41	0.43	0.61
6	High Precision - Lemmatized - Augmented	0.46	0.4	0.43	0.62
7	High Precision - Not Lemmatized - Not Augmented	0.53	0.19	0.28	0.71
8	High Precision - Not Lemmatized - Augmented	0.48	0.17	0.25	0.7

8.2.2. Deep Learning Classification E-mail Dataset Results

From Table 8.3, it is additionally clear that, in all cases, the labelling functions focused on high recall outperformed the results achieved with the labelling functions focused on high

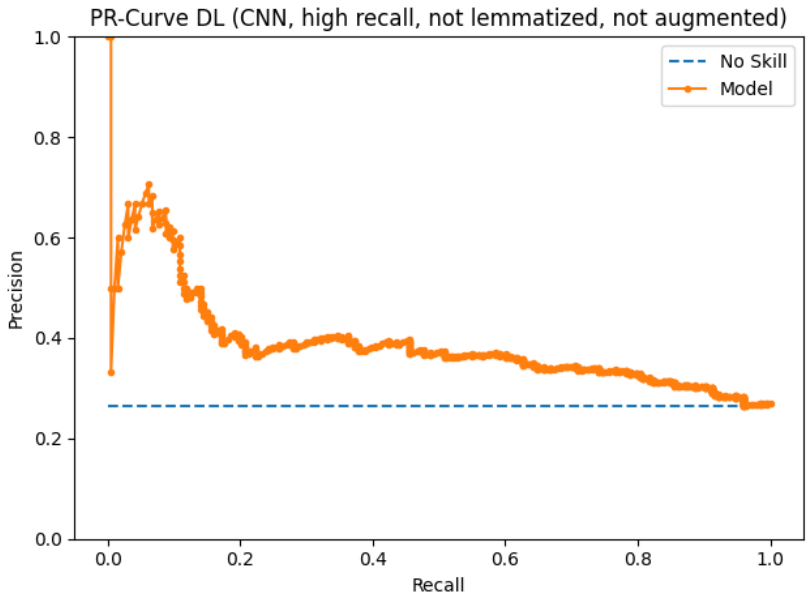

Figure 8.2: PR-Curve DL Classification -TEST

precision. The differences varied from a few per cent, namely ranking 3 compared to ranking 5, to a difference of over 20%, namely ranking 2 in comparison ranking 7.

For the labelling functions focused on high precision, the lemmatized versions of the training data outperformed the un-lemmatized versions, namely ranking 5 in comparison to ranking 7 and ranking 6 in comparison to ranking 8. However, for the labelling functions focused on high recall in combination with the unaugmented version of the training data, the un-lemmatized version of the training outperformed the lemmatized version, namely ranking 2 in comparison to ranking 3. Nonetheless, for the high-recall labelling functions in comparison with the augmented training data, the lemmatized version of the training data outperformed the un-lemmatized versions once more, namely ranking 1 in comparison to ranking 4.

Different from the test dataset, for the e-mail test dataset, in nearly all cases, the unaugmented version of the training data outperformed the augmented version of the training data. This could be observed when regarding ranking 2 in comparison to ranking 4, ranking 5 in comparison to ranking 6 and ranking 7 in comparison to ranking 8. Only for the high-recall labelling functions and the lemmatized version of the training data did the augmented training data outperform the augmented training data, namely for ranking 1 in comparison to ranking 3.

Ranked at first place and, therefore, able to be considered the ideal result was when high-recall labelling functions were used to label the training data and when the training data was lemmatized as well as augmented. The ideal result was obtained with a neural network consisting of only a CNN layer. The parameter settings for this result were as follows:

- **CNN Layer** - 50 nodes

- **Maximum Number of Features** - 5,000
- **Maximum Sentence Length** - 40

Compared to the baseline the deep learning classification for the e-mail dataset performs better regarding the F1-values, namely an increase of 6%. The greatest difference can be seen in the increase in recall, namely an increase of 11%. Unfortunately, the increase in precision is small, namely an increase of 4%, which means that also for the deep learning algorithm the precision remains low.

The PR-curve in Figure 8.3 shows that increasing the threshold of the decision boundary showed no increase in precision. Only when the recall was lower than 60% did the precision start to increase. A score lower than 60% for recall was insufficient, and, therefore, it was not beneficial to the result to change the threshold of the decision boundary.

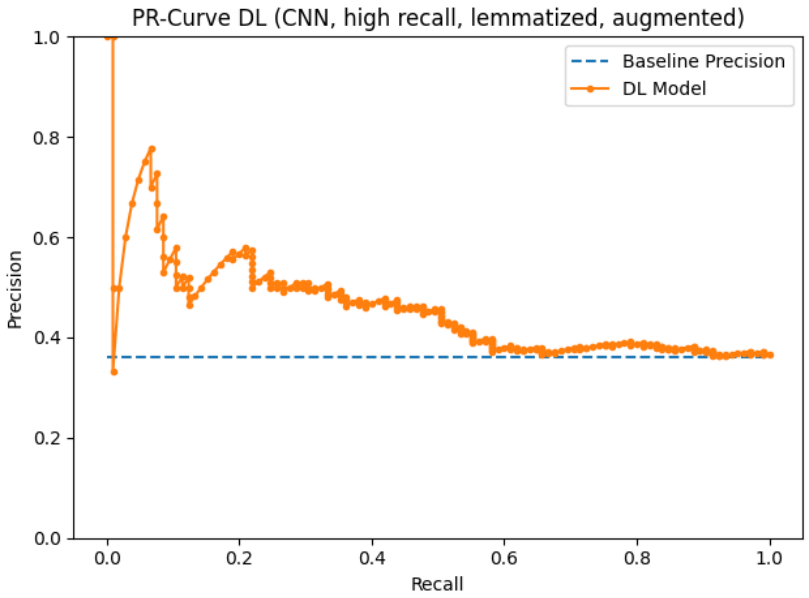

Figure 8.3: PR-Curve DL Classification - MAIL

In Table 8.4, the final results of the deep learning classification of both the test and e-mail test datasets are shown. Regarding the F1 scores, the results for the e-mail test dataset were more ideal than the test dataset, which was due to the 10% increase in precision. However, the recall for the e-mail dataset was lower than the test dataset, that being a decrease of 8%.

Table 8.4: Comparison Final Results Test and Mail for Deep Learning

Dataset	Precision	Recall	F1	Accuracy
TEST	0.34	0.76	0.47	0.54
MAIL	0.44	0.68	0.53	0.57

9

BERT-based Classification

In this chapter, for the BERT-based classification algorithm that was discussed in Chapter 4, the implementation and the results will be discussed. Multiple experiments were executed combining different training datasets that were created with certain combinations of variables and various parameter settings. This will provide the fourth part of the answer to the following sub-question that was stated in Chapter 1:

> *How should the chosen algorithms be implemented, and what combination of variables and parameter settings will lead to the most effective results for recognising personal opinions?*

9.1. BERT-based Classification Implementation

In this section, the way the BERT algorithm was implemented will be discussed. The algorithm that showed the most promise, as shown in Chapter 4, was the algorithm that has been described by Delobelle et al. (2020). The onset of neural networks in NLP have significantly improved SOTA results within the field. BERT has improved over previously used models by allowing the system to learn from input text in a bidirectional manner rather than only from left-to-right or the other way around (Delobelle et al., 2020). BERT was later re-implemented, critically evaluated and improved in the RoBERTa model (Liu et al., 2018). These models provide the advantage of being able to solve NLP tasks by having a common and expensive pre-training phase which can be downloaded off-the-shelf and that only needs to be fine-tuned. The pre-training happens in an unsupervised manner by providing large corpora of text in the desired language. The second phase only needs to be fine-tuned on a relatively small annotated dataset. While language models are generally trained on English data, several multilingual models exist which are trained on texts in different languages. Nonetheless, models trained on data from one specific language generally show more ideal results than the performance of multilingual models. For this reason, Delobelle et al. created a Dutch version of RoBERTa by pre-training on a Dutch dataset called RobBERT, a classic Dutch name.

RobBERT is pre-trained on the Dutch section of the OSCAR corpus, a large multilingual corpus, which was obtained by language classification in the Common Crawl corpus (Suarez et al., 2019). The corpus consists of 6.6 billion words, totalling 39 GB of text. It

contains 126,064,722 lines of text, where each line can have multiple sentences (Delobelle et al., 2020). In contrast to original BERT architecture, which uses WordPiece as subword embedding, RobBERT uses Byte Pair Encoding, which is used by generative pre-trained transformer 2 (GPT-2) (Radford et al., 2018) and RoBERTa (Liu et al., 2018).

The architecture that is used by RobBERT is equal to the architecture used in RoBERTa, which, in its turn, is an improvement over the original BERT. Therefore, the architecture consists of 12 self-attention layers with 12 heads (Devlin et al., 2019). The architectures of RobBERT and BERT are different due to the difference in pre-training, as RobBERT is only pre-trained on the MLM task, whereas BERT pre-trains both on the MLM and the NSP task. This means that RobBERT is only pre-trained on the word masking, where a prediction is made of which word was masked in a certain position of a given sentence.

The pre-trained model created by Delobelle et al. was downloaded and subsequently fine-tuned for the task of recognising personal opinions using the different training datasets. The code that implemented RobBERT, which prepared the data vectors, fine-tuned the pre-trained model, and, finally, tested were created by another data science intern of ZyLAB Youri van der Zee. The code was initially created for BERTje, which is the Dutch version of BERT, but this was changed to RobBERT for this research.

9.2. BERT-based Classification Results

In this section, the outcomes resulting from a BERT-based classification, called RobBERT, as proposed by Delobelle et al. (2020)), will be discussed. The learning rate was set to 5e-6 and the number of epochs was set to 3. For all combinations of variables, which are mentioned in Table 6.7, experiments were executed. For each combination of variables, experiments were performed for various maximum lengths of the sentences.

- **Maximum Sentence Length** - 15, 20, 30 and 50

Sentences that consisted of less words than the maximum sentences length were padded to the set maximum length. On the contrary, sentences that consisted of more words than the set maximum length were truncated to the maximum length. For each maximum length of the sentences, the ideal results are reported in Appendix D. From the experiments shown in Appendix E the most effective results for each combination of the variables, which are mentioned in Table 6.7, are shown in Table 9.1 for the test dataset and in Table 9.2 for the e-mail test dataset. The results will be discussed separately in the following sections.

9.2.1. Results BERT-based Classification Test Dataset

From Table 9.1, it can be seen that, in all cases, the labelling functions focused on high recall outperformed the results achieved with labelling functions focused on high precision. When considering the F1-values, the difference was, in nearly all combinations of the variables, over 20%.

The next observation that could be made was that, for all combinations of variables, augmenting the training data was beneficial for the results. The increase in the F1 score when augmenting the data was particularly due to an increase in precision, but an

Table 9.1: Ranking Results BERT-based Classification - Test

Ranking	Variables	Precision	Recall	F1	Accuracy
1	High Recall - Not Lemmatized - Augmented	0.4	0.6	0.48	0.66
2	High Recall - Not Lemmatized - Not Augmented	0.34	0.73	0.46	0.55
3	High Recall - Lemmatized - Augmented	0.37	0.59	0.46	0.64
4	High Recall - Lemmatized - Not Augmented	0.36	0.52	0.43	0.63
5	High Precision - Lemmatized - Augmented	0.45	0.25	0.32	0.72
6	High Precision - Not Lemmatized - Augmented	0.43	0.19	0.26	0.72
7	High Precision - Not Lemmatized - Not Augmented	0.57	0.13	0.21	0.74
8	High Precision - Lemmatized - Not Augmented	0	0	0	0.74

increase in recall could be observed in nearly all cases as well. This can be attributed to the fact that BERT-based algorithms require a considerable amount of data to fine-tune to the task of recognising personal opinions, which could explain for the augmented version of the training to outperform the unaugmented version.

From these results, it was difficult to assess what influence the lemmatization had on the results of the BERT-based classification. What could be noticed was that not lemmatizing the training data performed well for the high-recall labelling functions, namely ranking 1 in comparison to ranking 3 and ranking 2 in comparison to ranking 4. However, for the high precision labelling functions in combination with the augmented version of the training data, the lemmatization of the training data outperformed the un-lemmatized version.

Ranked in first place was the following combination of variables: high-recall labelling functions used to label the training data and the training data being un-lemmatized but augmented. The corresponding parameter setting of the maximum length was 20. Compared to the baseline, the BERT-based classification performed more ideally in terms of the F1-value, namely with an increase of 6%. The greatest difference could be seen in the increase in precision, namely an increase of 7%. Unfortunately, the increase in recall was smaller than all other classification algorithms, that being an increase of 2%.

In Figure 9.1, the PR-curve is displayed for the most effectively performing setting of variables. It seemed that when the recall was lowered by increasing the threshold of the decision boundary, the precision showed a slight increase, but recall showed a much larger decrease. Therefore, increasing the threshold for the decision boundary would not improve the result. For this reason, no further changes were made to the decision boundary.

9.2.2. Results BERT-based Classification E-mail Test Dataset

Similar to the results of the test dataset, in all cases, the labelling functions focused on high recall outperformed the results achieved with labelling functions focused on high precision. When reviewing the F1-values once more, the difference in all combinations of the variables was over 20%.

From these results, it could be concluded that, in nearly all cases, the lemmatized versions

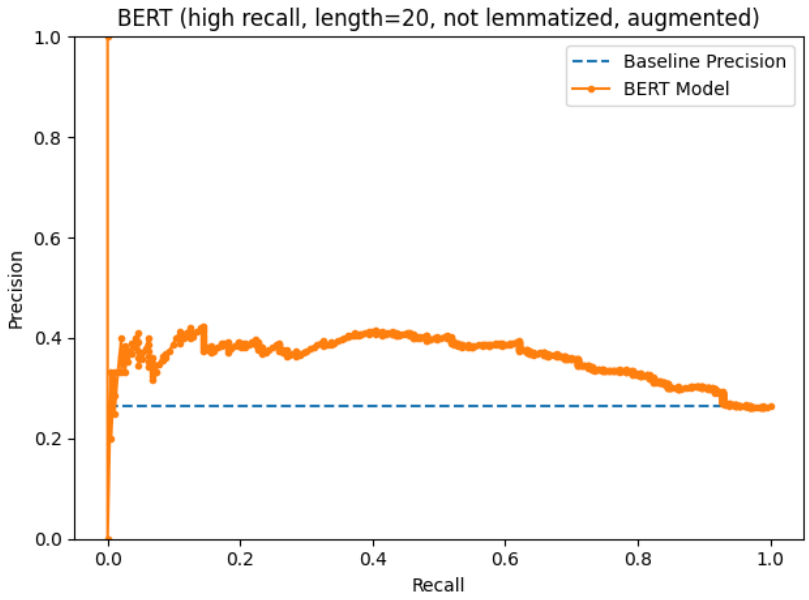

Figure 9.1: PR-Curve BERT-based Classification -TEST

Table 9.2: Ranking Results BERT-based Classification - Mail

Ranking	Variables	Precision	Recall	F1	Accuracy
1	High Recall - Lemmatized - Augmented	0.44	0.76	0.56	0.57
2	High Recall - Lemmatized - Not Augmented	0.39	0.76	0.51	0.53
3	High Recall - Not Lemmatized - Not Augmented	0.39	0.62	0.48	0.6
4	High Recall - Not Lemmatized - Augmented	0.42	0.57	0.48	0.6
5	High Precision - Lemmatized - Augmented	0.57	0.22	0.32	0.66
6	High Precision - Not Lemmatized - Augmented	0.55	0.17	0.26	0.71
7	High Precision - Not Lemmatized - Not Augmented	0.62	0.14	0.22	0.72
8	High Precision - Lemmatized - Not Augmented	0	0	0	0.64

of the training data outperformed the non-lemmatized versions of the training data, namely ranking 1 in comparison to ranking 4 and ranking 2 in comparison to ranking 3. Only for the high precision labelling functions and unaugmented data did the un-lemmatized version of the training data outperforms the lemmatized version.

In contrast to the results of the test dataset, for the e-mail test dataset, it was more difficult to assess what influence the augmentation had on the results. For the lemmatized versions of the training data, the augmented training data outperformed the non-augmented training data for both high precision and high-recall labelling functions. However, for the high recall labelling functions and the un-lemmatized version of the training data, the unaugmented training data performed more ideally than the augmented training data.

Ranked in first place and, therefore, able to be considered the ideal result was the combination of high-recall labelling functions to label the training data with the training data being lemmatized and augmented. The corresponding parameter setting of the maximum length was equal to the test dataset, namely a maximum length of 20.

Compared to the e-mail baseline, the BERT-based classification performed more ideally regarding the F1-value, namely with an increase of 9%. The greatest difference could be seen in the increase in recall, namely an increase of 19%. Unfortunately, the increase in precision was small, namely being an increase of 4%, which meant that, for the BERT-based algorithm as well, the precision remained low.

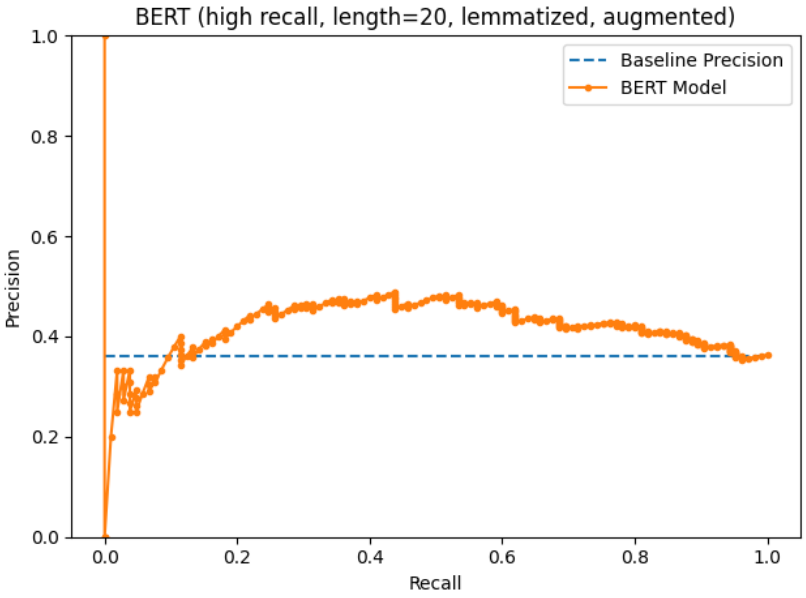

Figure 9.2: PR-Curve BERT-based Classification - Mail

In Figure 9.2, the PR-curve is displayed for the most effectively performing setting of variables. It seemed that when the recall was lowered by increasing the threshold of the decision boundary, the precision showed a slight increase in precision, but the recall showed a much larger decrease. Therefore, increasing the threshold for the decision boundary would not improve the result. For this reason, no further changes were made to the decision boundary.

In Table 9.3, the final results of the BERT-based classification for both the test and e-mail test datasets are shown. Regarding the F1 scores, the results for the e-mail test dataset were more ideal than for the test dataset, which was due to the 4% increase in precision and the 16% increase in precision.

Table 9.3: Comparison Final Results TEST and MAIL for BERT-based Classification

Dataset	**Precision**	**Recall**	**F1**	**Accuracy**
TEST	0.4	0.6	0.48	0.66
MAIL	0.44	0.76	0.56	0.57

10

Comparing the Results

In this chapter, a summary of the results will be provided of the different implemented classifications in order to provide a conclusion to the following sub-question that was stated in Chapter 1:

> *How should the chosen algorithms be implemented, and what combination of variables and parameter settings will lead to the most effective results for recognising personal opinions?*

Firstly, the rule-based classification baseline was set in Chapter 5. With help of the baseline, comparisons could be made with the more advanced classification algorithms to investigate whether or not the more advanced algorithms would outperform the rule-based classification. In Chapter 7, Chapter 8 and Chapter 9 for the traditional machine learning, deep learning and BERT-based classification algorithms, respectively, the various experiments performed were discussed, and the ideal results per combination of variables were stated in each chapter. Per classification algorithm, the following set of variables showed the most effective results for the test dataset:

- **Traditional Machine Learning Classification**: high-recall labelling functions and neither lemmatizing nor augmenting the training data
- **Deep Learning Classification**: high-recall labelling functions and neither lemmatizing nor augmenting the training data
- **BERT-based Classification**: high-recall labelling functions and not lemmatizing but augmenting the training data

From these results, it could be concluded that, in all cases, labelling functions focused on recall outperformed labelling functions focused on precision. Furthermore, not lemmatizing the training data additionally showed more ideal results for all classification algorithms. Only for augmenting the training data was a different setting necessary for the different algorithms because for the BERT-based classification augmentation improves the results when an insufficient amount of data is available. However, for traditional machine learning and for the deep learning classification, augmenting the data did not improve the results.

In Table 10.1, all the results of the different classification algorithms are shown for the test dataset. All the classification algorithms showed an increase in F1 values over the baseline due to both increases in recall and precision. It could be concluded that, for the test dataset, the traditional machine learning classification showed the ideal results due to the high score in recall and the second-highest score in precision. The BERT-based classification showed the highest result in precision, but the recall score was lowest compared to the other more advanced classification techniques.

Table 10.1: Overview Final Results - Test

Classification Algorithm	**Precision**	**Recall**	**F1**	**Accuracy**
Rule-based	0.33	0.58	0.42	0.59
Traditional Machine Learning	**0.35**	**0.87**	**0.5**	**0.54**
Deep Learning	0.34	0.76	0.47	0.54
BERT-Based	0.44	0.6	0.48	0.66

In Table 10.1, all the results of the different classification algorithms are shown for the e-mail test dataset. All the classification algorithms showed an increase in F1 values over the baseline due to both increases in recall and precision. It could be concluded that, for the e-mail dataset, once more, the traditional machine learning classification showed the ideal results due to the high recall score. Of note was that, for all classification techniques, precision was equal to 44%. When comparing the e-mail test dataset, all precision values increased or remained constant in contrast to the results for recall, as, for the e-mail dataset, except for the BERT-based and the rule-based classification, the recall values decrease.

Table 10.2: Overview Final Results - Mail

Classification Algorithm	**Precision**	**Recall**	**F1**	**Accuracy**
Rule-based	0.4	0.57	0.47	0.54
Traditional Machine Learning	**0.44**	**0.84**	**0.57**	**0.55**
Deep Learning	0.44	0.68	0.53	0.57
BERT-Based	0.44	0.76	0.56	0.57

In Table 10.3, all the results of the different classification algorithms are shown based on when the flexible constraints were used, which meant that only one sentence per redaction of a personal opinion had to be found. From the results, it could be concluded that the BERT-based classification showed the ideal results.

Table 10.3: Overview Final Results - Flexible Constraints

Classification Algorithm	**Precision**	**Recall**	**F1**	**Accuracy**
Rule-based	0.43	0.87	0.57	0.66
Traditional Machine Learning	0.39	1	0.56	0.57
Deep Learning	0.43	0.87	0.58	0.66
BERT-based	**0.49**	**0.83**	**0.61**	**0.72**

10.1. Another Method of Evaluating: Flexible Constraint Versus Strict Constraints

During the creation of the test dataset, it was decided that each data point was represented by a sentence from a document requested for a Wob-request. However, in many cases, multiple sentences were part of the same redaction of a personal opinion but were split into different sentences in the test dataset.

Nevertheless, the suggestions made by the algorithm would function as an area of interest for the Wob-specialists. Starting from the suggestion, the Wob-specialists would investigate whether or not the surrounding sentences were part of the same personal opinion described in the suggested sentence. If this was the case, the suggestion could be enlarged to the neighbouring sentences. Another observation that could be made from the redactions containing multiple sentences was that in many cases one sentence of the total redaction was indicating that the redaction indeed could be considered a personal opinion. However, the remaining sentences, which were also part of the redaction, were less likely to be recognised as a personal opinion. The following redaction was an example of this observation:

> ‘If the [GOVERNMENTAL BODY] deems it desirable that new [MEASUREMENT TECHNIQUE] is taken into account, we advise to implement [SET]. It is advised to make a new estimate after the [MODEL] has been calculated whether the calculation is deemed necessary. If this is the case, [SET] will come into effect. This results in an extra processing time of [TIME] compared to [SET].’

From the example it was apparent that in the first two sentences words were used which commonly express in opinions, namely ‘we advise...’ and ‘it is advised...’. However in the next two sentences such keywords were not used. When those sentences were to be assessed by the Wob-specialist, without the context information of the surrounding sentences, a different decision could have been made.

For this reason, it was important for each redaction made by a Wob-specialist and consisting of multiple sentences that at least one sentence be recognised by the classification algorithm. During the error analysis of the different classification algorithms, it was investigated whether the scores would improve when only one sentence of the redaction of a personal opinion had to be recognised. The constraint that all sentences had to be correctly classified was called ’strict’. When this constraint was changed to at least one sentence within a redaction, this was called ’flexible’.

During the error analysis, all redactions were manually checked to investigate what the results would be if not all but at least one sentence had to be recognised. The results are shown in Table 10.4. As expected, for all classification algorithms, an increase in F1 values could be observed. The largest increase that could be noted was for rule-based classification, where the F1 value increased by 15%. All values for precision increased by 10%, 4%, 9% and 5% for the rule-based, machine learning, deep learning and BERT-based classifications, respectively. Additionally, improvements could be seen for the recall,

namely increases of 19%, 13%, 11% and 5% for the machine learning, deep learning and BERT-based classification, respectively.

Table 10.4: Difference Between Strict and Flexible Constraints

Classification Algorithm	**Strict/Flexible**	**Precision**	**Recall**	**F1**	**Accuracy**
Rule-based	Strict	0.33	0.58	0.42	0.59
	Flexible	0.43	0.87	0.57	0.66
Traditional Machine Learning	Strict	0.35	0.87	0,5	0.54
	Flexible	0.39	1	0.56	0.57
Deep Learning	Strict	0.34	0.76	0.47	0.54
	Flexible	0.43	0.87	0.58	0.66
BERT-based	Strict	0.44	0.6	0.48	0.66
	Flexible	0.49	0.83	0.61	0.72

As the flexible constraints were considered sufficient by the Wob-specialist, the result from Table 10.4 were considered the final results. For this reason, the BERT-based classification algorithm proposed by Delobelle et al. (2020) will be considered as the most effective classification algorithm, due to the most promising F1-score.

11

Creating a Tool

As was presented in Chapter 10, it became apparent that the BERT-based approach proposed by Delobelle et al. (2020) had the most effective performance in terms of recognising the personal opinions within internal deliberations when flexible constraints were used. However, the F1 scores of the BERT-based classification were not and could not be expected to be without issue due to the complexity of the automatic redaction of personal opinions. Therefore, incorporating only the BERT-based classification into the ZyLAB ONE platform could lead to a situation where the recognition of personal opinions could make the redaction process more complicated and possibly slow the entire process. As a consequence, the automatic recognition of personal opinions would not be used by the Wob-specialists. In order to create a tool which would improve the current redaction process, multiple interviews were held with different Wob-specialists. This chapter provides an answer to the following sub-question that was stated in Chapter 1:

> *How should the automatic recognition of personal opinions be incorporated into the ZyLAB One platform in order to be a helpful tool for Wob-specialists?*

From the interviews held with Wob-specialists, it was clear that, in order for the recognition of personal opinions to become a helpful tool, a few different requirements had to be met. The requirements were divided into three different categories, namely the visual requirements, which will be discussed in Section 11.1; the amount of information that should be available, which was considered important, known as the information requirements and which will be discussed in Section 11.2; and finally, the various steps that needed to be taken to transform the suggestion for a personal opinion made by the algorithm to a final redaction. These requirements were called the processing requirements and will be discussed in more detail in Section 11.3.

11.1. Visual Requirements

The automatic redaction of personal opinions will be incorporated within the ZyLAB ONE platform. Decisions can be made on how the suggestions, which were made by the algorithm, will appear. A proposition was made to the Wob-specialists to present certain suggestions differently, for example, different colours for suggestions with a high probability of being a personal opinion than for suggestions with a probability closer to

the decision boundary, which could indicate uncertainty from the algorithm. However, it became apparent during the interviews that the most important visual requirements that were mentioned by all the Wob-specialists were consistency and simplicity. The reason these requirements were chosen was that each suggestion made by the algorithm would generally be humanly assessed no matter what the certainty score of a redaction will be. The only function the suggestions will have for the Wob-specialists is that the suggestions will lead the Wob-specialist's attention to that part of the text that could be a personal opinion. After the attention of the Wob-specialist is guided to the specific part of the text, the Wob-specialist will make the assessment about the suggested sentence and the surrounding sentences to confirm whether the suggestion can be considered a personal opinion.

The goal is to keep the overview of suggestions for personal opinions as simple and consistent as possible, which means that all the suggestions need to have the same colour. The suggestions additionally need to be transparent in order to keep the text behind the suggestion visible without moving the cursor over the text. In this way, the assessment can be swiftly made. Furthermore, the redaction of personal opinions is only one step of the entire redaction process. Consequently, to make it clear that the suggestions shown are part of the suggestions for personal opinions, a request from the Wob-specialists is to have different colours for the different steps of the redaction process.

11.2. Information Requirements

The following question was asked of all Wob-specialists: 'Would you blindly trust the decision of the classification algorithm?'. The reason this was asked was because it is difficult to imagine a person who would, in these high-risk situations, feel comfortable agreeing with the suggestions made by a classification algorithm without knowing how this decision was made. To reach trustworthiness, it is necessary to figure out the rationale of the machine behind the decisions. It would be interesting to not only provide the output of the machine but also provide explanations that can be understood by a human (Doran et al., 2017). This could be achieved by implementing Explainable AI techniques. Nonetheless, similar to what has been mentioned for the visual requirements, the answer of the Wob-specialists was that each suggestion has to be assessed by a human. This would additionally be the case in the event that an explanation of the decision was available. Therefore, it will not be necessary to provide extra information on, for example, how the suggestion was established because the suggestion will not be blindly trusted by the Wob-specialists.

Similar to the visual requirements, the Wob-specialists would prefer that the suggestions remain as simple as possible for the information requirements. In all the interviews, the only information that was requested by the Wob-specialists was the suggestion itself. However, this did not mean that the information was unwanted because it could be used for purposes other than the redaction of personal opinions. In certain governmental bodies, the set of documents requested would be divided into different categories, for example 'no risk', 'risk' and 'high risk'. The set of documents in the latter category could, for example, be assigned to a Wob-specialists with a legal background who would therefore be allowed to make the relevant high-risk decisions. In order to make that

division more specific, the information resulting from the classification could be used. One option could be to use the probabilities resulting from the classification to determine how certain it would be that a document would contain a personal opinion.

11.3. Processing Requirements

When considering the results that were mentioned in Chapter 10, it was apparent that, if the algorithm will be used in its present state of performance, there would be many false suggestions made by the algorithm due to low precision. In order to make the suggestions useful to the Wob-specialists, it will be important to consider the steps that would need to be taken to move from the suggestion to a final redaction.

The first processing requirement is that it should be simple to change suggestions made by the classification algorithm to a final redaction. The current precision score of the BERT-based classification is 49%. This means that slightly more than half of suggestions are personal opinions and should therefore be redacted. Consequently, it would take less time if suggestions have to be added to the final set of redactions. In a situation where the precision of the classification increases to more than 50%, the possibility should be available to remove the suggestions from the set of redactions instead of adding the suggestions to the final set of redactions. Either adding suggestions to or removing suggestions from the final set of redactions should be simple. This can be achieved by providing a button for each suggestion to either add or remove the suggestion from the set. In addition to either adding suggestions to or removing suggestions from the final set of redactions, it should be simple to add sentences that are not suggested by the BERT-based classification because the recall score has not reached 100%. For each sentence in the document, a button could be added to automatically add that sentence to the set of redactions.

The second requirement considered important by the Wob-specialists was that it needs to be simple to enlarge suggestions to more sentences or further include entire paragraphs or documents. The only reason the BERT-based classification performed more ideally than the traditional machine learning algorithm was because of the flexible constraints that were used. This meant that only one sentence of a redaction consisting of more than one sentence had to be found. Therefore, it is important that suggestions can be readily enlarged to multiple sentences, which can be achieved by adding movable corners to suggestions that can be dragged in order to make the area of the redaction larger. When the suggestion has to be enlarged to an entire paragraph or document, the possibility should exist of this occurring automatically.

The last processing requirement was that it needs to be simple to ask another person for a second opinion. When redacting personal opinions, the situation could arise where a Wob-specialist is uncertain about a certain suggestion. By giving the Wob-specialists the option to tag that specific sentences, it can improve the workflow of the Wob-specialist. The relevant document will then be stored together with other documents that should be reviewed by another person. By tagging that specific sentence, it is clear for which suggestion or multiple suggestions an uncertainty exists in the document.

12

General Discussion

This thesis consisted of various phases aimed at implementing a classification algorithm to automatically recognise personal opinions within internal deliberations. The current chapter will briefly summarise the main findings as well as elaborate on the implications of the research findings for further research.

12.1. Main Findings

This thesis aimed to find a classification algorithm which could recognise personal opinions within internal deliberations in order to accelerate the process of handling a Wob-request as well as decrease the risk of mistakes being made during the redaction process. The main research question was as follows:

> *How can the automatic recognition of personal opinions within internal deliberation in governmental documents be realised?*

This research question was divided into six different sub-questions which would ultimately lead to an answer to the central question of this thesis:

1. How is the redaction of personal opinions within internal deliberation currently addressed by Wob-specialists?

The redaction of personal opinions consists of two different phases. The first phase is to assess whether or not a document is part of internal deliberation. From conversations held with different Wob-specialists, it is apparent that all documents that are considered correspondence between two or more officiaries are to be considered part of internal deliberation. However, no predefined rules exist about what documents should fall under internal deliberation.

The second phase consists of finding sentences that contain a personal opinion. These sentences are searched for within the documents that are classified as internal deliberation by Wob-specialists. From the conversations and examining previously redacted documents, it can be concluded that the redaction of personal opinions is subjective to the person redacting the documents. It is possible that different

Wob-specialists can make different assessments about the same sentences because a ready-made answer on what should be considered a personal opinion does not exist.

Nonetheless, several similarities exist as to how Wob-specialists handle the redaction of personal opinions. In a majority of cases, it seems that the Wob-specialist searches for certain combinations of words. This method of redacting sentences can be compared to a rule-based approach. Therefore, a set of rules has been created through this research that function as a baseline for the automatic recognition of personal opinions, and this is most similar to how Wob-specialists currently approach the redaction of personal opinions.

2. What are the main challenges regarding the automatic recognition of personal opinions within deliberations?

The main challenges when automating the recognition of personal opinions can be divided into three different categories, namely challenges humans encounter, challenges due to the lack of understanding of language when automating the redaction of personal opinions and challenges during the implementation of a classification algorithm.

Multiple challenges are additionally encountered during human redaction. The first challenge is that the redaction of personal opinions falls under the relative grounds of refusal, which means a trade-off has to be made between the sensitivity of the information and the public interest. Another challenge humans additionally encounter is the need for background information on the administrative matter. When the background information is available, the assessment of the sensitivity of information is more reliable.

Several challenges that are encountered are due to the lack of understanding of language when NLP techniques are used. The first challenge is the detection of the boundaries of personal opinions. The data that is used to train and test the algorithm has a fixed size, but, in reality, personal opinions can spread over multiple sentences, paragraphs or entire documents. Furthermore, the recognition of ambiguity in language is a skill that humans perform well at but is a challenge for the automatic recognition of personal opinions.

The final set of challenges are encountered during the implementation of a classification algorithm that automatically recognises personal opinions. The first challenge is the absence of labelled training and test datasets because it is not possible to extract the redaction information available on the ZyLAB ONE platform. The second challenge is that a majority of SOTA algorithms are developed for the English language, but personal opinions can be written in Dutch. Substitutions have to be found for language-dependent dictionaries, word embeddings and certain libraries, amongst other tools.

3. Which algorithms can be used for recognising personal opinions within deliberations?

For this thesis, in order to find the appropriate algorithms for automatically recognising personal opinions within internal deliberations, relevant literature was researched. Relevant literature was identified based on the following criteria:

- Focused on similar tasks, such as subjectivity versus objectivity recognition,

sentiment analysis or polarity classification

- Focused on similar datasets, such as the 'SUBJ'-dataset, which contains objective and subjective sentences
- Focused on sentence-level classification
- A focus on Dutch was preferred over English

The algorithms were divided into three different categories, namely traditional machine learning algorithms, deep learning algorithms and BERT-based algorithms. The following algorithms can be considered to be the most promising to be implemented for the automatic redaction of personal opinions:

- A traditional machine learning approach proposed by Kamal (2014) - an approach using different features such as TF-IDF, PoS, position, opinion indicator seed words and negation. The classifier that was used in this study was a NB classifier; however, the SVM classifier was used as well.
- A deep learning approach proposed by Liu and Guo (2019) - an approach using a CNN layer in combination with a RNN layer with attention.
- A BERT-based approach proposed by Delobelle et al. (2020) - an approach using a Dutch pre-trained BERT model called RobBERT which is only trained on the MLM-task.

4. How should a dataset be obtained to train and subsequently test the chosen algorithms?

The following steps were taken to create training data:

1. Downloading the documents that were tagged as 'partly public', which meant that certain parts of the documents were disclosed to the public while certain parts contained sensitive information, such as personal opinions, and were therefore redacted.
2. Using OCR on the documents to obtain the text in the documents.
3. Extracting sentences from the text files and preprocessing data by removing non-lexical characters from and lemmatizing the sentences.
4. Labelling the training data with labelling functions within the Snorkel framework.
5. Augmenting the training with transformation functions within the Snorkel framework

The test dataset was created by manually rewriting sentences within documents that were tagged as 'partly public' but were part of a different Wob-request than the documents that were used for the training data. Sentences that were redacted by the Wob-specialist were labelled as a personal opinion in the test dataset, and the sentences that were not redacted functioned as the non-personal opinions.

5. How should the chosen algorithms be implemented, and what combination of variables and parameter settings will lead to the most effective results for recognising personal opinions?

For the rule-based classification, rules were created which could be divided into four different categories: advice, opinion, suggestion and expectation. For each category, a dictionary of relevant words was developed and, by using PoS-tag combinations, were made among the words in the dictionaries and personal pronouns, amongst others, that were commonly used personal opinions.

The traditional machine learning approach proposed by Kamal (2014) did not perform as expected when it was implemented similarly to how it was described in the literature. Based on the information stated by Kamal (2014), the decision was made to only include the feature with the most information gain, which was the TF-IDF feature. Additionally, multiple N-grams were added to the feature vector instead of only the unigrams as proposed by Kamal (2014).

The deep learning approach, as proposed by Liu and Guo (2019), did not perform as expected when it was implemented similarly to how it was described in the literature. The decision was made to implement multiple neural networks, where the layers proposed by Liu and Guo (2019) were combined. The implemented neural networks consisted of a CNN, a biLSTM, a CNN in combination with a biLSTM or, finally, a CNN in combination with a biLSTM with attention.

The BERT-based approach was implemented similarly to how it was proposed by Delobelle et al. (2020).

The most promising results were obtained with the BERT-based approach proposed by Delobelle et al. (2020) when focused on the flexible constraints, which meant that only one sentence had to be recognised from a redaction consisting of multiple sentences. The combination of variables that showed the most effective performance for this classification algorithm included high-recall labelling functions and non-lemmatization but the augmentation of the training data. The results are presented in Table 12.1.

Table 12.1: BERT-based Final Results - Flexible Constraints

Classification Algorithm	**Precision**	**Recall**	**F1**	**Accuracy**
BERT-based	0.49	0.83	0.61	0.72

6.How should the automatic recognition of personal opinions be incorporated into the ZyLAB One platform in order to be a helpful tool for Wob-specialists?

Three different categories of requirements were identified during interviews with Wob-specialists, namely visual requirements, information requirements and processing requirements:

- **The main visual requirements** are simplicity and consistency. These requirements can be achieved by showing the suggestions in the same colour and in a transparent

manner. Additionally, the suggestions of personal opinions should have a different colour than other redaction suggestions such as those regarding personal information.

- **The main information requirements** are simplicity and consistency. These requirements can be achieved by only showing the suggestions without any additional information.
- **The main processing requirements** are that it should be simple to change suggestions, enlarge suggestions and ask for a second opinion. These first requirements can be achieved by adding the suggestions to instead of removing suggestions from the final set of redactions because the precision of the BERT-based classification is slightly less than 50%. Additionally, a simple button should exist to add sentences that were not recognised by the BERT-based classification because recall continues to be at 83%. This means that 17% of the sentences containing a personal opinion will not be recognised. The second processing requirement can be achieved by giving the Wob-specialist the possibility of automatically enlarging the suggestion to the entire paragraph or document. The last requirement can be achieved by adding the possibility of tagging a certain suggestion when any confusion exists regarding whether the suggestion indeed contains a personal opinion.

12.2. Future Research

This thesis took the first steps towards the automatic recognition of personal opinions within internal deliberations. In comparison to the baseline set with the rule-based classification, improvements were realised with the BERT-based classification.Nonetheless, for the automatic recognition of personal opinions to be a helpful tool for Wob-specialists, several additional steps need to be taken. Currently, the scores for recall are considered sufficient, namely 83%. The main focus for further research should be on increasing precision, because currently the precision score is 49%. Four possible steps can be taken to increase precision.

1. Export sentences together with redaction information

For this research, it was not possible to export the documents from the ZyLAB ONE platform together with the redaction information. Therefore, the redaction information was lost when the documents were downloaded from the platform. In order to solve this problem, the sentences in the dataset were noisily labelled with the Snorkel framework, which meant that a portion of the data was incorrectly labelled. Subsequently, the more advanced algorithms were trained with the training data which contained incorrectly labelled data. As a consequence, it was more difficult for the classification algorithm to learn the difference between personal opinions and other texts. In contrast, when a tool is developed which can extract the sentences from the ZyLAB ONE platform together with the redaction information, the sentences could be labelled using that information.

2. More data is needed in order to avoid the need for augmentation

Due to the sensitivity of the information, it was difficult to gain access to a sufficient amount of data. As a result, it was necessary to augment the data to avoid problems for

several of the more advanced classification techniques, such as the BERT-based algorithm. However, creating a larger dataset from different governmental agencies could improve the results.

3. Create different training datasets and classifier for different types of documents
From the results, it can be concluded that the classification algorithms performed differently on the e-mail test dataset. This was due to the difference in language for the different types of documents. It could be beneficial to, for example. train a classifier on only e-mail data and subsequently validate the results on e-mail data.

4. More in-depth analysis of the most promising algorithm
In this thesis, the focus was on comparing multiple classification algorithms in order to improve the rule-based baseline. However, it could be beneficial to perform several additional in-depth analyses for the most promising classification algorithm, namely the BERT-based classification.

Bibliography

M. Anandarajan, C. Hill, and T. Nolan. *Practical Text Analysis*. Springer, 2019.

W. Chong, B. Selvaretnam, and L. Soon. Natural language processing for sentiment analysis. *4th International Conference on Artificial Intelligence with Applications in Engineering and Technology*, 2014. URL https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7351837&tag=.

Cornell. Movie review data. 2019. URL cs.cornell.edu/people/pabo/movie-review-data.

W. de Vries, A. van Cranenburgh, A. Bisazza, T. Caselli, G. van Noord, and M. Nissim. Bertje: A dutch bert model. 2019. URL https://arxiv.org/pdf/1912.09582.pdf.

P. Delobelle, T. Winters, and B. Berendt. Robbert: a dutch roberta-based language model. January 2020. URL https://github.com/iPieter/RobBERT.

J. Devlin, M. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171 – 4186, 2019.

D. Doran, S. Schulz, and T. Besold. What does explainable ai really mean? a new conceptualization of perspectives. 2017. URL https://arxiv.org/pdf/1710.00794.pdf.

Z. Gao, A. Feng, X. Song, and X. Wu. Target-dependent sentiment classification with bert. *IEEEAccess*, October 2019. URL https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8864964.

Google AI. Open sourcing bert: State-of-the-art pre-training for natural language processing. 2018. URL https://ai.googleblog.com/2018/11/open-sourcing-bert-state-of-art-pre.html.

A. Hassan and A. Mahmood. Efficient deep learning model for text classification based on recurrent and convolutional layers. *16th IEEE International Conference on Machine Learning and Applications*, 2017. URL https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8260793.

N. Jiang, F. Tian, J. Li, X. Yuan, and J. Zheng. Man: Mutual attention neural networks model for aspect-level sentiment classification in siot. *Internet of Things Journal, Vol 7. No. 4*, 2020. URL https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8949459.

A. Kamal. Subjectivity classification using machine learning techniques for mining feature-opinion pairs from web opinion sources. 2014.

Lexman Advocaten. Alles over de wob en wat u moet weten over de toepassing. 2020. URL https://wob.nl/alles-over-de-wob/.

G. Liu and J. Guo. Bidirectional lstm with attention mechanism and convolutional layer for text classification. *Neurocomputing 337*, pages 325 – 338, 2019. URL https://reader.elsevier.com/reader/sd/pii/S0925231219301067.

Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. Roberta: A robustly optimized bert pretraining approach. 2018.

Z. Liu, X. Bai, T. Cai, C. Chen, W. Zhang, and L. Jiang. Improving sentence representations with local and global attention for classification. *International Joint Conference on Neural Networks*, July 2019.

T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. 2013.

Ministerie van Algemene Zaken. Openbaarheid van overheidsinformatie. *Wet openbaarheid van bestuur (Wob) | Rijksoverheid.nl*, March 2019. URL https://www.rijksoverheid.nl/onderwerpen/wet-openbaarheid-van-bestuur-wob/openbaarheid-van-overheidsinformatie.

A. Montejo-Ráez, E. Martínez-Cámara, M. Martín-Valdivia, and L. Urena-López. Ranked wordnet graph for sentiment polarityclassification in twitter. *Computer Speech and Language 28*, pages 93 —- 107, April 2013. URL https://reader.elsevier.com/reader/sd/pii/S0885230813000284.

M. Munikar, S. Shakya, and A. Shrestha. Fine-grained sentiment classification using bert. *Artificial Intelligence for Transforming Business and Society (AITB)*, October 2019. URL https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8947435.

Nederlandse Grondwet. Artikel 110: Openbaarheid van bestuur. 2020. URL https://www.denederlandsegrondwet.nl/id/via0istdmgzu/artikel_110_openbaarheid_van_bestuur.

Overheid.nl. Wet openbaarheid van bestuur. 2020. URL https://wetten.overheid.nl/BWBR0005252/2018-07-28.

A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. Improving language understanding by generative pre-training. 2018. URL https://s3-us-west-2.amazonaw.com/openaiassets/researchcovers/languageunsupervised/languageunderstandingpaper.pdf.

A. Sharma and S. Dey. A comparative study of feature selection and machine learning techniques for sentiment analysis. *Proceedings of the 2012 ACM Research in Applied Computation Symposium*, October 2012. URL https://dl.acm.org/doi/pdf/10.1145/2401603.2401605.

Snorkel. Snorkel. 2019. URL https://github.com/snorkel-team/snorkel.

Y. Song. Mihnet: Combining n-gram, sequential and global information for text classification. *Journal of Physics: Conference Series*, 2020. URL https://iopscience.iop.org/Article/10.1088/1742-6596/1453/1/012156/pdf.

R. Srivastava, K. Greff, and J. Schmidhuber. Highway networks. 2015. URL https://arxiv.org/pdf/1505.00387.pdf.

P. Suarez, B. Sagot, and L. Romary. Asynchronous pipeline for processing huge corpora on medium to low resource infrastructures. *7th Workshop on the Challenges in the Management of Large Corpora (CMLC-7)*, 2019.

G. Tripathi and S. Naganna. Feature selection and classification approach for sentiment analysis. *Machine Learning and Applications: An International Journal (MLAIJ) Vol.2, No.2*, Juni 2015. URL https://pdfs.semanticscholar.org/2b6c/3ac7ba8fb563fb5d4b3b7e682f11c12d1560.pdf.

M. Trupthi, S. Pabboju, and G. Narasimha. Improved feature extraction and classification - sentiment analysis. *International Conference on Advances in Human Machine Interaction*, March 2016. URL https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7449189.

S. Tulkens, C. Emmery, and W. Daelemans. Evaluating unsupervised dutch word embeddings as a linguistic resource. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may 2016. European Language Resources Association (ELRA). ISBN 978-2-9517408-9-1.

United States Department of Justice. Foia.gov (freedom of information act) learn. *Freedom of Information Act: Learn*, 2020. URL https://www.foia.gov/about.html.

Wetten.nl. Circulaire handleiding wob en wbp. 2015. URL https://wetten.overheid.nl/BWBR0036845/2015-05-01.

Wetten.nl. Wet bescherming persoonsgegevens. 2018. URL https://wetten.overheid.nl/BWBR0011468/2018-05-01.

Y. Zhanga, Z. Zhanga, D. Miaoa, and J. Wang. Three-way enhanced convolutional neural networks for sentence-level sentiment classification. *Information Sciences 477*, pages 55 – 64, 2019.

M. Zulqarnain, S. Ishak, R. Ghazali, N. Nawi, M. Aamir, and Y. Hassim. An improved deep learning approach based on variant two-state gated recurrent unit and word embeddings for sentiment classification. *International Journal of Advanced Computer Science and Applications, Vol. 11, No. 1*, 2020. URL https://pdfs.semanticscholar.org/6502/ad8e3a06f65a44235f981a3fa0dc4a9e6e35.pdf.

A

Wet Openbaarheid van Bestuur: Legal Background

In this appendix the original description of the Wob is given, as stated in wetten.nl Wetten.nl (2015).

I. Definitions

Article 1

For the purposes of this Act and the provisions based thereon:

a document: a written document or other material held by an administrative body containing information;

b administrative matter: a matter relating to the policy of an administrative body, including its preparation and implementation;

c Internal deliberation: the deliberation on an administrative matter within an administrative body or within a circle of administrative bodies in the context of joint responsibility for an administrative matter;

d Non-official advisory committee: a body set up by the government with the task of advising one or more administrative bodies and of which no civil servants are members, which advise the administrative body to which they belong on the subjects submitted to the body. Officials who are secretary or advisory members of an advisory body shall not be regarded as members of it for the purposes of this provision;

e Official or mixed advisory body: a body which has the task of advising one or more administrative bodies, made up in whole or in part of officials, whose function is to advise the administrative body to which they belong on the matters referred to the body;

f personal policy opinion: an opinion, proposal, recommendation or conclusion of one or more persons on an administrative matter and the arguments put forward by them for that purpose;

g environmental information: that which is understood in Article 19.1a of the Environmental Management Act.

Article 1A

1 This Act shall apply to the following governing bodies:

a Our Ministers;

b The administrative bodies of provinces, municipalities, water boards and public-law business organisations;

c administrative bodies operating under the responsibility of the bodies referred to under a and b;

d other administrative bodies, insofar as not exempted by general order in council.

2 other administrative bodies, insofar as not exempted by general order in council.

II. Public Access

Article 2

1 In the performance of its duties, an administrative body shall, without prejudice to the provisions laid down elsewhere by law, provide information in accordance with this law and shall do so on the basis of the public interest in the disclosure of information.

2 The administrative body shall ensure as much as possible that the information it provides in accordance with this law is up to date, accurate and comparable.

III. Information on request

Article 3

1 Any person may address a request for information laid down in documents relating to an administrative matter to an administrative body or to an institution, service or company operating under the responsibility of an administrative body.

2 The applicant shall state in his application the administrative matter or the document relating to it on which he wishes to receive information.

3 The applicant shall not be required to declare an interest in the application.

4 If an application is formulated in too general a manner, the administrative authority shall as soon as possible ask the applicant to clarify the application and shall assist the applicant in doing so.

5 A request for information shall be granted in accordance with the provisions of Articles 10 and 11.

Article 4

If the application concerns information contained in documents held by an administrative body other than the one to which the application was made, the applicant shall, where appropriate, be referred to that body. If the application is made in writing, it shall be forwarded to the applicant, who shall be notified of the forwarding.

Article 5

1 A decision on a request for information shall be taken orally or in writing.

2 A total or partial refusal of a written request for information shall be in writing. In the case of an oral request, refusal shall be in writing if the applicant so requests. The applicant shall be informed of this possibility.

3 The decision shall also be in writing if the request for information concerns a third party and the third party has requested it. In that case, the information relating to the third party shall also be sent to the third party.

Article 6

1 The administrative body shall decide on the request for information as soon as possible, but at the latest within four weeks counting from the day after the day on which the request was received.

2 The administrative authority may postpone the decision for a maximum period of four weeks. The applicant shall be notified of the postponement in writing, stating the reasons for the postponement, before the expiry of the first period.

3 Without prejudice to article 4:15 of the General Administrative Law Act, the period for issuing a decision shall be suspended from the day following the day on which the administrative authority informs the applicant that article 4:8 of the General Administrative Law Act has been applied, until the day on which the interested party or parties have expressed an opinion or the period set for this purpose has expired unused.

4 If the suspension as referred to in subsection 3 ends, the administrative authority shall notify the applicant as soon as possible, stating the period within which the decision must still be taken.

5 If the administrative authority has decided to provide information, the information shall be provided at the same time as the decision is published, unless an interested party is expected to object, in which case the information shall not be provided until two weeks after the decision has been published.

6 To the extent that the request concerns the provision of environmental information:

a By way of derogation from the first paragraph, the deadline for taking a decision shall be two weeks if the administrative authority intends to supply the environmental information while an interested party is expected to object;

b the decision may only be postponed under subsection 2 if the volume or complexity of the environmental information justifies an extension;

c paragraphs 3 and 4 do not apply.

Article 7

1 The administrative body shall provide the information relating to the documents containing the requested information by:

a giving a copy of it or providing its literal content in another form,

b to allow access to the content,

c Provide an extract or summary of the content; or

d to provide information from it.

2 The administrative organ shall provide the information in the form requested by the applicant, unless otherwise stated:

a the provision of the information in that form cannot reasonably be required;

b the information is already publicly available in another form which is easily accessible to the applicant.

3 If the request concerns environmental information as referred to in Article 19.1a(1)(b) of the Environmental Management Act, the administrative body shall, if necessary, and if such information is available, also provide information on the methods used in compiling the first-mentioned information.

IV. Information of own accord

Article 8

1 The governing body that directly concerns it shall, on its own initiative, provide information about the policy, including its preparation and implementation, as soon as this is in the interest of good and democratic governance.

2 The administrative authority shall ensure that the information is provided in a comprehensible form, in such a way that interested and interested citizens are reached as much as possible and at such times that they can bring their views to the attention of the administrative authority in a timely manner.

Article 9

1 The administrative body directly concerned shall make public, where necessary and possible with an explanation, the opinions issued to the body by non-official advisory committees with a view to the policy to be formed, together with the requests for opinions and proposals submitted to the committees by the body.

2 Publication shall take place within four weeks after the advice has been received at the latest and shall be announced in the Government Gazette or another periodical made generally available by the government. Total or partial non-disclosure shall be notified in the same manner.

3 The documents referred to in the first paragraph may be made public by them:

a including them in a generally available publication,

b To be issued separately and made generally available; or

c to make it available for inspection, to provide a copy or to lend it out.

V. Grounds for exemptions and restrictions

Article 10

1 The provision of information by virtue of this law shall be omitted insofar as this is the case:

a could endanger the unity of the Crown;

b could harm the security of the State;

c concerns business and manufacturing data, which have been confidentially communicated to the government by natural persons or legal entities;

d personal data as referred to in Articles 9, 10 and 87 of the General Data Protection Regulation, unless the provision does not manifestly infringe on privacy.

2 The provision of information pursuant to this law shall also be omitted insofar as the interest thereof does not outweigh the following interests:

a the relations of the Netherlands with other states and with international organisations;

b the economic or financial interests of the State, the other bodies governed by public law or the administrative bodies referred to in article 1a, under c and d;

c the investigation and prosecution of criminal offences;

d inspection, control and supervision by administrative bodies;

e respect for privacy;

f the interest of the addressee in being the first to have access to the information;

g the prevention of any disproportionate advantage or disadvantage to natural or legal persons involved in the matter or to third parties.

3 The second paragraph, opening words and under e shall not apply insofar as the person concerned has consented to disclosure.

4 The first paragraph, opening words and under c and d, the second paragraph, opening words and under e, and the seventh paragraph, opening words and under a, are not applicable insofar as it concerns environmental information relating to emissions into the environment. Furthermore, by way of derogation from paragraph 1(c), the provision of environmental information shall be omitted only to the extent that the interest in disclosure outweighs the interest referred to therein.

5 The second paragraph, opening words and under b, shall apply to the provision of environmental information insofar as such information relates to acts of a confidential nature.

6 The second paragraph, opening words and under g, is not applicable to the provision of environmental information.

7 The supply of environmental information pursuant to this law shall also be omitted insofar as its importance does not outweigh the following interests:

a the protection of the environment to which this information relates;

b the security of companies and the prevention of sabotage.

8 Insofar as the first sentence of subsection 4 does not apply, when applying subsections 1, 2 and 7 to environmental information it shall be taken into account whether such information relates to emissions into the environment.

Article 11

1 In the event of a request for information from documents drawn up for the purposes of internal deliberation, no information shall be provided on personal policy views contained therein.

2 Information on personal policy opinions may be provided in a non-personally identifiable form for the purposes of good and democratic governance. If the person who has expressed these views or expressed his or her views has consented to this, the information may be provided in a form that can be traced back to individuals.

3 With respect to advice given by an official or mixed advisory committee, the provision of information about the personal policy views contained therein may take place if the intention to do so has been made known to the members of the advisory committee by the administrative body directly concerned before the start of their activities.

4 By way of derogation from subsection 1, in the case of environmental information the importance of protecting personal policy views shall be weighed against the importance of disclosure. Information on personal policy summaries may be provided in a non-personally identifiable form. The second paragraph, second sentence, shall apply accordingly.

B

Rule Based Implementation

Advice

```
!! ADVIES GEVEN

!1: "Ik adviseer dit."/"Iemand adviseert dit."
!2: "Mijn advies/mijns inziens is het volgende"
!3: "Intern is besproken..."

#subgroup ADVIES_WW1:{
<STEM:adviseren,POS:V>
|<STEM:opperen,POS:V>
|(<STEM:bevelen,POS:V><>*<aan>)
|(<STEM:dragen,POS:V><>*<aan>)
|(<STEM:leggen,POS:V><>*<voor>)
|(<STEM:brengen,POS:V><>*<in>)
|(<STEM:werpen,POS:V><>*<op>)
|(<STEM:stellen,POS:V><>*<voor>)
|(<STEM:prijzen,POS:V><>*<aan>)
|(<STEM:raden,POS:V><>*<aan>)
|<STEM:aanbevelen,POS:V>
|<STEM:aandragen,POS:V>
|<STEM:voorleggen,POS:V>
|<STEM:inbrengen,POS:V><>
|<STEM:opwerpen,POS:V><>
|<STEM:voorstellen,POS:V>
|<STEM:aanprijzen,POS:V>
|<STEM:aanraden,POS:V>
}

#group 1_ADVIES_PBO:{
[SN]<>*([UL](<POS:Pron-Pers>|<POS:Prop>),(%(ADVIES_WW1))[/UL])<>*[/SN]
}
```

```
#subgroup ADVIES_NW1:{
(<POS:Det-Poss><STEM:mening,POS:Nn>)
|(<POS:Det-Poss><STEM:inzicht,POS:Nn>)
|(<POS:Det-Poss><STEM:inziens,POS:Nn>)
|(<POS:Det-Poss><STEM:advies,POS:Nn>)
|(<POS:Det-Poss><STEM:idee,POS:Nn>)
|(<POS:Det-Poss><STEM:vraag,POS:Nn>)
|<STEM:idee, POS:Nn>
|(<STEM:wat><STEM:mij><STEM:betreft>)
|(<STEM:het><STEM:komen><STEM:mij><voor>)
|(<STEM:lijken, POS:V><STEM:het><POS:Pron-Pers>)
|(<STEM:goed><STEM:om><>*<POS:V-Inf>)
}

#group 2_ADVIES_PBO:{
[SN] <>* %(ADVIES_NW1)<>*[/SN]
}

#group 3_ADVIES_PBO:{
[SN] <>* <STEM:intern> <>*[/SN]
}
```

Opinion

```
!1: ik vind dit/iemand vindt dit
!2: Mijn mening is het volgende
!3: Het lijkt me een goed idee/Het lijkt iemand een goed idee
!4: ..., lijkt mij.
!5: Het baart mij zorgen
!6: Er is onduidelijkheid/onenigheid/... over
!7: Met de volgende strekking
!8: Eens


#subgroup MENING_WW1:{
<STEM:vinden,POS:V>
|<STEM:achten,POS:V>
|<STEM:beschouwen,POS:V>
|<STEM:menen,POS:V>
|<STEM:rekenen,POS:V>
|<STEM:oordelen,POS:V>
|<STEM:veronderstellen,POS:V>
|<STEM:aannemen,POS:V>
|(<STEM:neem,POS:V><>*<STEM:aan>)
|<STEM:pleiten,POS:V>
```

```
}

#group 1_MENING_PBO:{
[SN]<>*([UL](<POS:Pron-Pers>|<POS:Prop>),(%(MENING_WW1))[/UL]) <~(plaats){0}><>*[/SN]
}

#subgroup MENING_NW1:{
(<naar><POS:Det-Poss><STEM:mening,POS:Nn>)
|(<POS:Det-Poss><STEM:mening,POS:Nn>)
|(<volgens><POS:Pron-Pers>)
|(<POS:Det-Poss><STEM:inziens,POS:Nn>)
|(<naar><POS:Det-Poss><STEM:inschatting, POS:Nn>)
|(<POS:Det-Poss><STEM:standpunt, POS:Nn>)
|(<in><POS:Det-Poss><STEM:optiek, POS:Nn>)
|(<POS:Det-Poss><STEM:reactie, POS:Nn>)
|(<POS:Det-Poss><>*<STEM:analyse, POS:Nn>)
}

#group 2_MENING_PBO:{
[SN]<>*(%(MENING_NW1))<>*[/SN]
}

#subgroup MENING_WW2:{
<STEM:lijken,POS:V>
|<STEM:klinken,POS:V>
|<STEM:voelen,POS:V>
|<STEM:schijnen,POS:V>
}

#group 3_MENING_PBO:{
[SN]<POS:Pron-Indef>(%(MENING_WW2))<>*[/SN]
}

#group 4_MENING_PBO:{
[SN]<>*(%(MENING_WW2))<POS:Pron-Pers>[/SN]
}

#subgroup MENING_WW3:{
<STEM:baren,POS:V>
}

#group 5_MENING_PBO:{
[SN]<POS: Pron-Indef>(%(MENING_WW3))<POS:Pron-Pers><>*[/SN]
}

#subgroup MENING_NW2:{
```

```
<STEM:onduidelijkheid,POS:Nn>
|<STEM:chaos,POS:Nn>
|<STEM:onenigheid,POS:Nn>
|<STEM:paniek,POS:Nn>
|<STEM:ongenoegen,POS:Nn>
|<STEM:onvrede,POS:Nn>
|<STEM:meningsverschil,POS:Nn>
}

#group 6_MENING_PBO:{
[SN]<STEM:er><STEM:is, POS:V><>*(%(MENING_NW2))<STEM:over><>*[/SN]
}

#group 7_MENING_PBO:{
[SN]<>*<STEM:met><STEM:de><STEM:volgende><STEM:strekking><>*[/SN]
}

#group 8_MENING_PBO:{
[SN]<>*<~(nog){0}><>*<STEM: eens><>*[/SN]
}
```

Suggestion

```
!1: Ik opper om dat te doen
!2: Ben je het eens met:
!3: Is het niet verstandig om?
!4: het verzoek om:
!5: hij vraagt of...

#subgroup SUGGESTIE_WW1:{
<STEM:voorstellen,POS:V>
|<STEM:opperen,POS:V>
|<STEM:aandragen,POS:V>
|<STEM:inbrengen,POS:V>
|<STEM:voorleggen,POS:V>
|<STEM:suggereren,POS:V>
|<STEM:aanvoeren,POS:V>
|<STEM:noemen,POS:V>
|<STEM:probeer,POS:V>
}

#group 1_SUGGESTIE_PBO:{
[SN]<>*([UL](<STEM:ik>|<STEM:wij>),(%(SUGGESTIE_WW1))[/UL])<>*[/SN]
}

#group 2_SUGGESTIE_PBO:{
```

```
[SN]<>*<STEM:is, POS:V> (<STEM:jij>|<STEM:jullie>) <het> <eens> <met> <>* <\?>[/SN]
}

#subgroup SUGGESTIE_NW1:{
<STEM:verstandig,POS:Adj-Pred>
|<STEM:slim,POS:Adj-Pred>
|<STEM:intelligent,POS:Adj-Pred>
|<STEM:redelijk,POS:Adj-Pred>
|<STEM:wijs,POS:Adj-Pred>
|<STEM:bedachtzaam,POS:Adj-Pred>
|<STEM:zinnig,POS:Adj-Pred>
|<STEM:schappelijk,POS:Adj-Pred>
}

#group 3_SUGGESTIE_PBO:{
[SN]<>*<STEM:is, POS:V> <STEM:het> <>* (%(SUGGESTIE_NW1)) <STEM:om> <>* <\?>[/SN]
}

#subgroup SUGGESTIE_NW2:{
<STEM:verzoek,POS:Nn>
|<STEM:aanvraag,POS:Nn>
|<STEM:oproep,POS:Nn>
|<STEM:vraag,POS:Nn>
|<STEM:aanzoek,POS:Nn>
}

#group 4_SUGGESTIE_PBO:{
[SN]<>*(%(SUGGESTIE_NW2)) <STEM:om> <>*[/SN]
}

#group 5_SUGGESTIE_PBO:{
[SN]<>*(<POS:Pron-Pers>|<POS:Prop>)<STEM:of><>*[/SN]
}
```

Expectations

```
!!VERWACHTING
!1: "Ik verwacht het volgende"
!2: "Mijn idee is het volgende"
!3: "Naar verwachting kunnen we het volgende doen"
!4: "Verwachting is:..."
!5: "Het zou kunnen dat..."

#subgroup VERWACHTINGEN_WW1:{
<STEM:verwachten,POS:V>
|(<STEM:nemen,POS:V> <aan>)
```

```
|(<STEM:gaan,POS:V> <STEM:ervan> <STEM:uit>)
|(<STEM:rekenen,POS:V> <STEM:op>)
|<STEM:vertrouwen,POS:V>
|<STEM:hopen,POS:V>
|<STEM:vermoeden,POS:V>
|<STEM:anticiperen,POS:V>
|<STEM:wensen,POS:V>
|<STEM:denken,POS:V>
|<STEM:geloven,POS:V>
}

#group 1_VERWACHTING_PBO:{
[SN]<>*([UL](<STEM:ik>|<STEM:wij>),(%(VERWACHTINGEN_WW1))[/UL])<>*[/SN]}

#subgroup VERWACHTINGEN_NW1:{
(<POS:Det-Poss> <STEM:verwachting,POS:Nn>)
|(<POS:Det-Poss> <STEM:idee,POS:Nn>)
|(<POS:Det-Poss> <STEM:vermoeden,POS:Nn>)
|(<POS:Det-Poss> <STEM:hoop,POS:Nn>)
|(<POS:Det-Poss> <STEM:wens,POS:Nn>)
|(<POS:Det-Poss> <STEM:hoop,POS:Nn>)
}

#group 2_VERWACHTING_PBO:{
[SN](%(VERWACHTINGEN_NW1))<POS:Aux-Fin><>*[/SN]
}

#group 3_VERWACHTING_PBO:{
[SN]<STEM:naar><STEM:verwachting,POS:Nn><STEM:zijn, POS:V><>*[/SN]
}

#subgroup VERWACHTINGEN_NW2:{
(<STEM:verwachting,POS:Nn>)|
(<STEM:aanname, POS:Nn>)|
(<STEM:hoop, POS:Nn>)|
(<STEM:vooruitzicht, POS:Nn>)|
(<STEM:perspectief, POS:Nn>)|
(<STEM:vermoeden, POS:Nn>)|
(<STEM:voorspelling, POS:Nn>)
}


#group 4_VERWACHTING_PBO:{
[SN]<>*(%(VERWACHTINGEN_NW2))<POS:V><>*[/SN]
}
```

```
#group 5_VERWACHTING_PBO:{
[SN]<>*<STEM:zullen,POS:V><STEM:kunnen,POS:V-Inf><>*[/SN]
}
```

C

Overview Machine Learning Results

In this section an overview will be given of the results of the traditional machine Learning classification. Multiple experiments have been performed to find the best combination of variables to recognise personal opinions within internal deliberations:

- Augmentation: yes/no
- Lemmatization: yes/no
- High Recall Labelling Functions/High Precision Labelling Functions

Total Test Dataset

In this section the experiments are stated tested on the entire test dataset.

Table C.1: Traditional ML Results (High Recall - Test Dataset - Lemmatized)

	NOT AUG - TEST					AUG - TEST				
Parameter Settings	**Precision**	**Recall**	**F1**	**Accuracy**	**Classifier**	**Precision**	**Recall**	**F1**	**Accuracy**	**Classifier**
max_df = 0.1 & ngram = 1-3	0.34	0.68	0.45	0.573	SVM RBF	*0.32*	*0.73*	*0.45*	*0.53*	*SVM Sigmoid*
max_df = 0.1 & ngram = 1-5	0.34	0.7	0.46	0.573	SVM RBF	0.32	0.72	0.44	0.53	SVM Sigmoid
max_df = 0.1 & ngram = 1-7	0.33	0.71	0.45	0.561	SVM Linear	0.32	0.71	0.44	0.54	SVM RBF
max_df = 0.3 & ngram = 1-3	0.33	0.76	0.46	0.534	SVM Linear	0.31	0.75	0.44	0.505	SVM RBF
max_df = 0.3 & ngram = 1-5	0.31	0.78	0.45	0.508	SVM RBF	0.34	0.65	0.45	0.582	SVM Poly 2
max_df = 0.3 & ngram = 1-7	*0.32*	*0.79*	*0.46*	*0.517*	*SVM Linear*	0.36	0.6	0.45	0.616	SVM Poly 2
max_df = 0.5 & ngram = 1-3	0.32	0.74	0.45	0.527	SVM Poly 2	0.3	0.78	0.43	0.466	SVM RBF
max_df = 0.5 & ngram = 1-5	0.31	0.76	0.45	0.511	SVM Poly 2	0.29	0.85	0.44	0.431	SVM RBF
max_df = 0.5 & ngram = 1-7	0.29	0.86	0.44	0.427	NB	0.29	0.86	0.44	0.427	SVM RBF
max_df = 0.7 & ngram = 1-3	0.3	0.87	0.44	0.436	NB	0.3	0.76	0.43	0.474	SVM Linear
max_df = 0.7 & ngram = 1-5	0.29	0.87	0.44	0.422	NB	0.29	0.86	0.44	0.428	SVM Linear
max_df = 0.7 & ngram = 1-7	0.3	0.85	0.44	0.438	SVM RBF	0.29	0.87	0.44	0.428	SVM Sigmoid

bottom!

Table C.2: Traditional ML Results (High Recall - Test Dataset - Not Lemmatized)

	NOT AUG - TEST					AUG - TEST				
Parameter Settings	**Precision**	**Recall**	**F1**	**Accuracy**	**Classifier**	**Precision**	**Recall**	**F1**	**Accuracy**	**Classifier**
max_df = 0.1 & ngram = 1-3	*0.35*	*0.87*	*0.5*	*0.54*	*SVM Linear*	0.33	0.75	0.45	0.53	SVM RBF
max_df = 0.1 & ngram = 1-5	0.33	0.88	0.48	0.5	SVM Linear	0.29	0.95	0.45	0.38	NB
max_df = 0.1 & ngram = 1-7	0.34	0.87	0.49	0.52	NB	*0.31*	*0.83*	*0.46*	*0.48*	*SVM RBF*
max_df = 0.3 & ngram = 1-3	0.33	0.89	0.48	0.49	NB	0.3	0.8	0.45	0.48	SVM RBF
max_df = 0.3 & ngram = 1-5	0.32	0.89	0.47	0.48	*NB*	0.29	0.96	0.44	0.36	NB
max_df = 0.3 & ngram = 1-7	0.33	0.88	0.48	0.5	NB	0.31	0.82	0.45	0.47	SVM Linear
max_df = 0.5 & ngram = 1-3	0.32	0.89	0.47	0.48	NB	0.29	0.94	0.44	0.36	NB
max_df = 0.5 & ngram = 1-5	0.32	0.89	0.47	0.48	NB	0.29	0.96	0.44	0.36	NB
max_df = 0.5 & ngram = 1-7	0.33	0.88	0.48	0.49	NB	0.3	0.86	0.45	0.44	SVM Poly 2
max_df = 0.7 & ngram = 1-3	0.32	0.89	0.47	0.48	NB	0.28	0.94	0.44	0.36	NB
max_df = 0.7 & ngram = 1-5	0.32	0.89	0.47	0.48	NB	0.28	0.96	0.44	0.35	NB
max_df = 0.7 & ngram = 1-7	0.33	0.88	0.47	0.49	NB	0.3	0.86	0.45	0.43	SVM Poly 2

Table C.3: Traditional ML Results (High Precision - Test Dataset - Lemmatized)

	NOT AUG - TEST					AUG - TEST				
Parameter Settings	**Precision**	**Recall**	**F1**	**Accuracy**	**Classifier**	**Precision**	**Recall**	**F1**	**Accuracy**	**Classifier**
max_df = 0.1 & ngram = 1-3	0.41	0.28	0.33	0.71	SVM Linear	0.43	0.26	0.33	0.72	SVM RBF
max_df = 0.3 & ngram = 1-3	0.51	0.41	0.45	0.74	SVM RBF	0.48	0.34	0.4	0.73	SVM RBF
max_df = 0.3 & ngram = 1-5	0.43	0.54	0.48	0.7	SVM Linear	0.43	0.53	0.48	0.7	SVM Linear
max_df = 0.3 & ngram = 1-7	0.4	0.61	0.49	0.66	SVM RBF	0.42	0.61	0.5	0.68	SVM Linear
max_df = 0.5 & ngram = 1-3	0.49	0.44	0.47	0.74	SVM RBF	0.48	0.35	0.41	0.73	SVM RBF
max_df = 0.5 & ngram = 1-5	0.43	0.6	0.5	0.69	SVM RBF	0.44	0.58	0.5	0.7	SVM Linear
max_df = 0.5 & ngram = 1-7	0.39	0.65	0.49	0.65	SVM RBF	0.39	0.66	0.49	0.64	SVM RBF
max_df = 0.7 & ngram = 1-3	0.49	0.44	0.46	0.73	SVM RBF	0.48	0.36	0.41	0.73	SVM RBF
max_df = 0.7 & ngram = 1-5	0.43	0.59	0.5	0.69	SVM RBF	*0.39*	*0.68*	*0.5*	*0.64*	*SVM RBF*
max_df = 0.7 & ngram = 1-7	*0.42*	*0.64*	*0.5*	*0.67*	SVM Sigmoid	0.39	0.68	0.5	0.64	*SVM RBF*

Table C.4: Traditional ML Results (High Precision - Test Dataset - Not Lemmatized)

	NOT AUG - TEST					AUG - TEST				
Parameter Settings	**Precision**	**Recall**	**F1**	**Accuracy**	**Classifier**	**Precision**	**Recall**	**F1**	**Accuracy**	**Classifier**
max_df = 0.1 & ngram = 1-3	0.47	0.18	0.26	0.73	SVM Linear	0.48	0.11	0.18	0.73	SVM Linear
max_df = 0.1 & ngram = 1-5	0.46	0.3	0.36	0.72	SVM Linear	0.47	0.25	0.33	0.73	SVM Linear
max_df = 0.1 & ngram = 1-7	0.44	0.33	0.38	0.71	SVM Linear	0.43	0.27	0.33	0.71	SVM Linear
max_df = 0.3 & ngram = 1-3	0.53	0.23	0.32	0.74	SVM RBF	0.51	0.15	0.23	0.74	SVM Linear
max_df = 0.3 & ngram = 1-5	0.46	0.34	0.39	0.72	SVM RBF	0.47	0.28	0.35	0.73	SVM Linear
max_df = 0.3 & ngram = 1-7	0.44	0.4	0.42	0.71	SVM RBF	0.44	0.34	0.38	0.71	SVM Linear
max_df = 0.5 & ngram = 1-3	0.56	0.24	0.34	0.75	SVM RBF	0.53	0.16	0.24	0.74	SVM Linear
max_df = 0.5 & ngram = 1-5	0.44	0.33	0.38	0.71	SVM Linear	0.49	0.32	0.38	0.73	SVM Linear
max_df = 0.5 & ngram = 1-7	*0.45*	*0.42*	*0.44*	*0.71*	*SVM RBF*	*0.43*	*0.35*	*0.38*	*0.7*	*SVM Linear*
max_df = 0.7 & ngram = 1-3	0.55	0.24	0.33	0.75	SVM RBF	0.53	0.16	0.24	0.74	SVM Linear
max_df = 0.7 & ngram = 1-5	0.46	0.35	0.4	0.72	SVM RBF	0.49	0.32	0.38	0.73	SVM Linear
max_df = 0.7 & ngram = 1-7	0.45	0.42	0.44	0.71	SVM RBF	0.49	0.32	0.38	0.73	SVM Linear

E-mail Subset Test Dataset

In this section the experiments are stated tested on the e-mail subset of the test dataset.

Table C.5: Traditional ML Results (High Recall - Mail Dataset - Lemmatized)

	NOT AUG - MAIL					AUG - MAIL				
Parameter Settings	**Precision**	**Recall**	**F1**	**Accuracy**	**Classifier**	**Precision**	**Recall**	**F1**	**Accuracy**	**Classifier**
max_df = 0.1 & ngram = 1-3	0.41	0.81	0.54	0.512	NB	0.4	0.85	0.54	0.485	NB
max_df = 0.1 & ngram = 1-5	0.41	0.84	0.55	0.512	NB	0.39	0.86	0.54	0.467	NB
max_df = 0.1 & ngram = 1-7	0.42	0.84	0.56	0.515	NB	0.39	0.86	0.54	0.474	NB
max_df = 0.3 & ngram = 1-3	0.42	0.84	0.56	0.515	NB	0.39	0.86	0.54	0.471	NB
max_df = 0.3 & ngram = 1-5	*0.44*	*0.84*	*0.57*	*0.55*	*SVM RBF*	0.39	0.88	0.54	0.467	NB
max_df = 0.3 & ngram = 1-7	*0.44*	*0.84*	*0.57*	*0.55*	*SVM RBF*	0.4	0.88	0.55	0.47	NB
max_df = 0.5 & ngram = 1-3	0.41	0.87	0.56	0.505	NB	0.39	0.86	0.54	0.464	NB
max_df = 0.5 & ngram = 1-5	0.41	0.88	0.56	0.498	NB	0.4	0.85	0.55	0.55	SVM RBF
max_df = 0.5 & ngram = 1-7	0.42	0.87	0.56	0.512	SVM Sigmoid	0.39	0.88	0.54	0.46	NB
max_df = 0.7 & ngram = 1-3	0.41	0.88	0.56	0.509	NB	0.42	0.76	0.54	0.54	SVM Poly 2
max_df = 0.7 & ngram = 1-5	0.44	0.81	0.57	0.553	SVM Poly 2	0.41	0.85	0.55	0.5	SVM Linear
max_df = 0.7 & ngram = 1-7	0.41	0.86	0.55	0.502	SVM Linear	*0.41*	*0.87*	*0.56*	*0.5*	*SVM Linear*

Table C.6: Traditional ML Results (High Recall - Mail Dataset - Not Lemmatized)

	NOT AUG - MAIL					AUG - MAIL				
Parameter Settings	**Precision**	**Recall**	**F1**	**Accuracy**	**Classifier**	**Precision**	**Recall**	**F1**	**Accuracy**	**Classifier**
max_df = 0.1 & ngram = 1-3	*0.39*	*0.88*	*0.54*	*0.55*	*SVM Sigmoid*	0.33	0.94	0.49	0.41	NB
max_df = 0.1 & ngram = 1-5	0.37	0.89	0.52	0.51	SVM Sigmoid	0.33	0.95	0.49	0.41	NB
max_df = 0.1 & ngram = 1-7	0.37	0.86	0.52	0.53	NB	0.35	0.84	0.5	0.49	SVM RBF
max_df = 0.3 & ngram = 1-3	0.37	0.85	0.52	0.52	SVM Poly 2	0.36	0.81	0.5	0.51	SVM RBF
max_df = 0.3 & ngram = 1-5	0.37	0.85	0.52	0.52	SVM Poly 2	0.33	0.97	0.49	0.4	NB
max_df = 0.3 & ngram = 1-7	0.36	0.88	0.51	0.51	NB	*0.35*	*0.84*	*0.5*	*0.49*	*SVM Poly 2*
max_df = 0.5 & ngram = 1-3	0.37	0.86	0.51	0.52	SVM Poly 2	0.34	0.83	0.49	0.47	SVM RBF
max_df = 0.5 & ngram = 1-5	0.36	0.89	0.51	0.49	NB	0.33	0.97	0.49	0.39	NB
max_df = 0.5 & ngram = 1-7	0.36	0.88	0.51	0.5	NB	0.33	0.97	0.49	0.4	NB
max_df = 0.7 & ngram = 1-3	0.36	0.9	0.51	0.48	SVM Sigmoid	0.33	0.94	0.48	0.4	NB
max_df = 0.7 & ngram = 1-5	0.36	0.89	0.51	0.49	NB	0.33	0.97	0.49	0.39	NB
max_df = 0.7 & ngram = 1-7	0.36	0.88	0.51	0.49	SVM Linear	0.33	0.97	0.49	0.39	NB

Table C.7: Traditional ML Results (High Precision - Mail Dataset - Lemmatized)

	NOT AUG - MAIL					AUG - MAIL				
Parameter Settings	**Precision**	**Recall**	**F1**	**Accuracy**	**Classifier**	**Precision**	**Recall**	**F1**	**Accuracy**	**Classifier**
max_df = 0.1 & ngram = 1-3	0.46	0.3	0.37	0.62	SVM Linear	0.43	0.26	0.33	0.72	SVM RBF
max_df = 0.3 & ngram = 1-3	0.54	0.42	0.47	0.66	SVM RBF	0.48	0.34	0.4	0.73	SVM RBF
max_df = 0.3 & ngram = 1-5	0.49	0.57	0.53	0.63	SVM RBF	0.43	0.53	0.48	0.7	SVM Linear
max_df = 0.3 & ngram = 1-7	0.48	0.65	0.55	0.62	SVM RBF	0.42	0.61	0.5	0.68	SVM Linear
max_df = 0.5 & ngram = 1-3	0.55	0.48	0.51	0.67	SVM RBF	0.48	0.35	0.41	0.73	SVM RBF
max_df = 0.5 & ngram = 1-5	0.5	0.63	0.55	0.64	SVM RBF	0.44	0.58	0.5	0.7	SVM Linear
max_df = 0.5 & ngram = 1-7	0.47	0.66	0.55	0.6	SVM RBF	0.39	0.66	0.49	0.64	SVM RBF
max_df = 0.7 & ngram = 1-3	0.55	0.48	0.51	0.67	SVM RBF	0.48	0.36	0.41	0.73	SVM RBF
max_df = 0.7 & ngram = 1-5	0.49	0.62	0.55	0.63	SVM RBF	0.44	0.58	0.5	0.7	SVM Linear
max_df = 0.7 & ngram = 1-7	*0.5*	*0.65*	*0.56*	*0.64*	*SVM Sigmoid*	*0.39*	*0.68*	*0.5*	*0.64*	*SVM RBF*

bottom!

Table C.8: Traditional ML Results (High Precision - Mail Dataset - Not Lemmatized)

	NOT AUG - TEST					AUG - TEST				
Parameter Settings	**Precision**	**Recall**	**F1**	**Accuracy**	**Classifier**	**Precision**	**Recall**	**F1**	**Accuracy**	**Classifier**
max_df = 0.1 & ngram = 1-3	0.5	0.19	0.27	0.7	SVM Linear	0.51	0.12	0.19	0.7	SVM Linear
max_df = 0.1 & ngram = 1-5	0.47	0.31	0.38	0.69	SVM Linear	0.48	0.26	0.34	0.7	SVM Linear
max_df = 0.1 & ngram = 1-7	0.45	0.35	0.4	0.68	SVM Linear	0.43	0.28	0.34	0.68	SVM Linear
max_df = 0.3 & ngram = 1-3	0.54	0.24	0.33	0.71	SVM RBF	0.53	0.15	0.24	0.71	SVM Linear
max_df = 0.3 & ngram = 1-5	0.48	0.36	0.41	0.69	SVM RBF	0.49	0.29	0.36	0.7	SVM Linear
max_df = 0.3 & ngram = 1-7	0.46	0.41	0.44	0.68	SVM RBF	0.46	0.35	0.4	0.68	SVM Linear
max_df = 0.5 & ngram = 1-3	0.57	0.26	0.35	0.72	SVM RBF	0.56	0.16	0.25	0.71	SVM Linear
max_df = 0.5 & ngram = 1-5	0.48	0.36	0.41	0.69	SVM RBF	0.51	0.32	0.4	0.71	SVM Linear
max_df = 0.5 & ngram = 1-7	0.46	0.44	0.45	0.68	SVM RBF	0.45	0.36	0.4	0.68	SVM Linear
max_df = 0.7 & ngram = 1-3	0.56	0.25	0.35	0.72	SVM RBF	0.56	0.16	0.25	0.71	SVM Linear
max_df = 0.7 & ngram = 1-5	0.48	0.36	0.41	0.69	SVM RBF	0.51	0.32	0.4	0.71	SVM Linear
max_df = 0.7 & ngram = 1-7	*0.47*	*0.44*	*0.45*	*0.68*	*SVM RBF*	*0.45*	*0.36*	*0.4*	*0.68*	*SVM Linear*

D

Overview Deep Learning Results

In this section an overview will be given of the results of the deep learning classification. Multiple experiments have been performed to find the best combination of variables to recognise personal opinions within internal deliberations:

- Augmentation: yes/no
- Lemmatization: yes/no
- High Recall Labelling Functions/High Precision Labelling Functions

Total Test Dataset

In this section the experiments are stated tested on the entire test dataset.

Table D.1: DL Results (High Recall - Test Dataset - Lemmatized)

	NOT AUG - TEST				**AUG - TEST**			
	Precision	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**
CNN	**0.33**	**0.62**	**0.43**	**0.58**	**0.31**	**0.74**	**0.44**	**0.51**
biLSTM	0.32	0.52	0.4	0.59	0.33	0.61	0.43	0.58
CNN + biLSTM	0.32	0.66	0.43	0.55	0.34	0.55	0.42	0.61
CNN + biLSTM + Attention	0.29	0.7	0.41	0.49	0.33	0.62	0.43	0.58

Table D.2: DL Results (High Recall - Test Dataset - Not Lemmatized)

	NOT AUG - TEST				**AUG - TEST**			
	Precision	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**
CNN	**0.34**	**0.76**	**0.47**	**0.54**	**0.32**	**0.66**	**0.43**	**0.55**
biLSTM	0.35	0.63	0.45	0.59	0.34	0.47	0.4	0.63
CNN + biLSTM	0.33	0.73	0.46	0.54	0.34	0.51	0.41	0.61
CNN + biLSTM + Attention	0.33	0.75	0.46	0.53	0.34	0.5	0.41	0.61

E-mail Subset Test Dataset

In this section the experiments are stated tested on the e-mail subset of the test dataset.

Table D.3: DL Results (High Precision - Test Dataset - Lemmatized)

	NOT AUG - TEST				AUG - TEST			
	Precision	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**
CNN	0.48	0.32	0.39	0.73	0.41	0.38	0.4	0.7
biLSTM	0.49	0.28	0.36	0.74	**0.48**	**0.35**	**0.41**	**0.73**
CNN + biLSTM	**0.46**	**0.35**	**0.39**	**0.72**	0.49	0.19	0.28	0.74
CNN + biLSTM + Attention	0.44	0.31	0.36	0.72	0.4	0.26	0.32	0.71

Table D.4: DL Results (High Precision - Test Dataset - Not Lemmatized)

	NOT AUG - TEST				AUG - TEST			
	Precision	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**
CNN	**0.48**	**0.22**	**0.3**	0.73	**0.43**	**0.17**	**0.24**	**0.72**
biLSTM	0.49	0.19	0.28	0.73	0.44	0.11	0.18	0.73
CNN + biLSTM	0.57	0.16	0.25	0.75	0.57	0.12	0.2	0.74
CNN + biLSTM + Attention	0.4	0.11	0.18	0.72	0.46	0.12	0.19	0.73

Table D.5: DL Results (High Recall - Mail Dataset - Lemmatized)

	NOT AUG - MAIL				AUG - MAIL			
	Precision	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**
CNN	**0.39**	**0.66**	**0.49**	**0.5**	**0.44**	**0.68**	**0.53**	**0.57**
biLSTM	0.38	0.47	0.42	0.54	0.41	0.59	0.49	0.55
CNN + biLSTM	0.4	0.64	0.49	0.6	0.41	0.54	0.47	0.55
CNN + biLSTM + Attention	0.4	0.64	0.49	0.6	0.4	0.69	0.5	0.51

Table D.6: DL Results (High Recall - Mail Dataset - Not Lemmatized)

	NOT AUG - MAIL				AUG - MAIL			
	Precision	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**
CNN	**0.39**	**0.73**	**0.51**	**0.58**	0.35	0.58	0.44	0.56
biLSTM	0.37	0.69	0.48	0.56	0.37	0.44	0.4	0.61
CNN + biLSTM	0.38	0.49	0.43	0.61	**0.33**	**0.75**	**0.46**	**0.53**
CNN + biLSTM + Attention	0.35	0.68	0.46	0.53	0.35	0.53	0.43	0.57

Table D.7: DL Results (High Precision - Mail Dataset - Lemmatized)

	NOT AUG - TEST				AUG - TEST			
	Precision	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**
CNN	0.57	0.29	0.38	0.66	**0.46**	**0.4**	**0.43**	**0.62**
biLSTM	0.57	0.3	0.39	0.67	0.57	0.29	0.38	0.66
CNN + biLSTM	**0.45**	**0.41**	**0.43**	**0.61**	0.62	0.22	0.32	0.67
CNN + biLSTM + Attention	0.48	0.2	0.28	0.7	0.52	0.18	0.27	0.71

Table D.8: DL Results (High Precision - Mail Dataset - Not Lemmatized)

	NOT AUG - MAIL				AUG - MAIL			
	Precision	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**
CNN	0.47	0.18	0.26	0.69	0.42	0.14	0.21	0.68
biLSTM	0.53	0.14	0.22	0.71	0.53	0.11	0.19	0.71
CNN + biLSTM	**0.53**	**0.19**	**0.28**	**0.71**	0.53	0.11	0.18	0.71
CNN + biLSTM + Attention	0.51	0.11	0.18	0.7	**0.48**	**0.17**	**0.25**	**0.7**

E

Overview BERT-based Results

In this section an overview will be given of the results of the deep learning classification. Multiple experiments have been performed to find the best combination of variables to recognise personal opinions within internal deliberations:

- Augmentation: yes/no
- Lemmatization: yes/no
- High Recall Labelling Functions/High Precision Labelling Functions

Total Test Dataset

In this section the experiments are stated tested on the entire test dataset.

Table E.1: BERT Results (High Recall - Test Dataset - Lemmatized)

	NOT AUG - TEST				**AUG - TEST**			
	Precision	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**
max length = 15	0.32	0.58	0.41	0.57	0.37	0.59	0.46	0.64
max length = 20	**0.36**	**0.52**	**0.43**	**0.63**	0.3	0.62	0.41	0.39
max length = 30	0.33	0.41	0.36	0.63	**0.39**	**0.76**	**0.51**	**0.53**
max length = 50	0.32	0.51	0.39	0.6	0.33	0.48	0.39	0.62

Table E.2: BERT Results (High Recall - Test Dataset - Not Lemmatized)

	NOT AUG - TEST				**AUG - TEST**			
	Precision	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**
max length = 15	0.34	0.67	0.45	0.56	0.4	0.48	0.43	0.67
max length = 20	**0.34**	**0.73**	**0.46**	**0.55**	**0.4**	**0.6**	**0.48**	**0.66**
max length = 30	0.38	0.54	0.44	0.64	0.37	0.56	0.45	0.64
max length = 50	0.51	0.15	0.23	0.74	0.35	0.51	0.42	0.63

E-mail Subset Test Dataset

In this section the experiments are stated tested on the e-mail subset of the test dataset.

Table E.3: BERT Results (High Precision - Test Dataset - Lemmatized)

	NOT AUG - TEST				AUG - TEST			
	Precision	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**
max length = 15	0	0	0	0.74	0.41	0.24	0.3	0.71
max length = 20	0	0	0	0.74	**0.45**	**0.25**	**0.32**	**0.72**
max length = 30	0	0	0	0.74	0.55	0.18	0.27	0.74
max length = 50	0	0	0	0.74	0.8	0.02	0.05	0.75

Table E.4: BERT Results (High Precision - Test Dataset - Not Lemmatized)

	NOT AUG - TEST				AUG - TEST			
	Precision	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**
max length = 15	0.4	0.01	0.02	0.73	0.51	0.1	0.17	0.74
max length = 20	0.6	0.03	0.06	0.74	0.53	0.1	0.17	0.74
max length = 30	0.64	0.09	0.16	0.75	0.53	0.14	0.22	0.74
max length = 50	**0.57**	**0.13**	**0.21**	**0.74**	**0.43**	**0.19**	**0.26**	**0.72**

Table E.5: BERT Results (High Recall - Mail Dataset - Lemmatized)

	NOT AUG - MAIL				AUG - MAIL			
	Precision	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**
max length = 15	0.48	0.54	0.51	0.63	0.5	0.53	0.52	0.64
max length = 20	**0.39**	**0.76**	**0.51**	**0.53**	**0.44**	**0.76**	**0.56**	**0.57**
max length = 30	0.42	0.6	0.49	0.55	0.41	0.45	0.43	0.57
max length = 50	0.47	0.43	0.45	0.62	0.45	0.43	0.44	0.6

Table E.6: BERT Results (High Recall - Mail Dataset - Not Lemmatized)

	NOT AUG - MAIL				AUG - MAIL			
	Precision	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**
max length = 15	0.37	0.73	0.49	0.54	**0.42**	**0.57**	**0.48**	**0.63**
max length = 20	0.4	0.61	0.48	0.61	0.41	0.57	0.48	0.63
max length = 30	0.39	0.53	0.45	0.62	0.38	0.52	0.44	0.61
max length = 50	**0.39**	**0.62**	**0.48**	**0.6**	0.37	0.52	0.43	0.6

Table E.7: BERT Results (High Precision - Mail Dataset - Lemmatized)

	NOT AUG - MAIL				AUG - MAIL			
	Precision	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**
max length = 15	0	0	0	0.63	0.43	0.1	0.16	0.63
max length = 20	0	0	0	0.64	**0.57**	**0.22**	**0.32**	**0.66**
max length = 30	0	0	0	0.64	0.51	0.21	0.3	0.64
max length = 50	0	0	0	0.64	0.47	0.2	0.28	0.63

Table E.8: BERT Results (High Precision - Mail Dataset - Not Lemmatized)

	NOT AUG - MAIL				AUG - MAIL			
	Precision	**Recall**	**F1**	**Accuracy**	**Precision**	**Recall**	**F1**	**Accuracy**
max length = 15	0.5	0.02	0.03	0.7	0.54	0.09	0.15	0.71
max length = 20	0.55	0.03	0.06	0.7	0.46	0.06	0.11	0.7
max length = 30	0.55	0.07	0.12	0.071	0.51	0.16	0.24	0.7
max length = 50	**0.62**	**0.14**	**0.22**	**0.72**	**0.55**	**0.17**	**0.26**	**0.71**