Haldis Borgen and Oline Zachariassen

Detecting Private-Sensitive Content in Norwegian Social Media

Master's thesis in Computer Sience
Supervisor: Özlem Özgöbek
July 2023

Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Computer Science

NTNU
Norwegian University of
Science and Technology

Haldis Borgen and Oline Zachariassen

Detecting Private-Sensitive Content in Norwegian Social Media

Master's thesis in Computer Sience
Supervisor: Özlem Özgöbek
July 2023

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Computer Science

NTNU
Norwegian University of
Science and Technology

Abstract

Social media platforms have transformed the landscape of information sharing and online connectivity, but concerns have arisen regarding potential violations of users' privacy. With the rapid advancement of technology and globalization, protecting personal data has become increasingly challenging. This thesis addresses the problem of detecting and classifying private-sensitive content in Norwegian social media, with the aim of enhancing privacy protection for Norwegian social media users. A labelled data set specifically for private-sensitive content in Norwegian social media is created to achieve this. The data set is used to train the transformer-based NB BERT model and conventional classifiers for content detection. Our comparative analysis reveals that the fine-tuned NB BERT model achieves an F1 score of 0.82, surpassing the best-performing conventional classifier, which scores 0.74. The contributions of this thesis include a definition of private-sensitive content aligned with the General Data Protection Regulation (GDPR), a labelled data set of Norwegian social media content, a fine-tuned NB BERT-base model for Norwegian social media text, and a fine-tuned multi-class NB BERT-base model for classifying private-sensitive content. The research methods employed in this thesis involve a literature review, data annotation, and two experiments: training and fine-tuning the NB BERT-base model, as well as training conventional classifiers.

Sammendrag

Sosiale medieplattformer har endret landskapet for hvordan informasjon deles og hvordan folk forbindes over nett, men det har oppstått bekymringer angående potensielle brudd på brukernes personvern. Med den raske teknologiske utviklingen og globaliseringen har det blitt stadig vanskeligere å beskytte persondata. Denne masteroppgaven adresserer problemet med å oppdage og klassifisere privat-sensitivt innhold på norske sosiale medier, der målet er å bidra til bedre personvernbeskyttelse for norske sosiale mediebrukere. For å oppnå dette opprettes et annotert datasett som inneholder privat-sensitiv tekst fra sosiale medier på norsk. Dette datasettet brukes til å trene NB BERT-modellen, som er basert på transformers, samt konvensjonelle klassifikatorer for deteksjon. Den finjusterte NB BERT-modellen oppnår en F1-score på 0.82, noe som overgår den konvensjonelle klassifikatoren som fikk den nest beste F1-scoren på 0.74. Bidragene fra denne masteroppgaven inkluderer en definisjon av privat-sensitivt innhold basert på EUs personvernforordning (GDPR), et annotert datasett med norsk tekst fra sosiale medier, en finjustert NB BERT-base-modell for norsk tekst i sosiale medier, og en fler-klasse NB BERT-base-modell for klassifisering av privat-sensitivt innhold. Forskningsmetodene som er brukt i denne oppgaven inkluderer en litteraturgjennomgang, annotering av data og to eksperimenter: trening og finjustering av NB BERT-base-modellen, samt trening av konvensjonelle klassifikatorer.

Preface

This thesis was submitted to the Norwegian University of Science and Technology (NTNU) for the final course *TDT4900 Computer Science, Master's Thesis*. Our research was conducted at NTNU and supervised by Özlem Özgöbek, associate professor at the Department of Computer Science at NTNU.

We want to thank Özlem Özgöbek for providing guidance during this research and for her valuable feedback. We would also like to thank the annotators who generously volunteered their time to label the dataset used in our experimental research. Additionally, we would like to thank our family and friends for their support and assistance in proofreading our thesis.

Haldis Borgen and Oline Zachariassen

Trondheim, 13th July 2023

Contents

List of Figures

List of Tables

Acronyms

API Application Programming Interface.

BERT Bidirectional Encoder Representations from Transformers.

BoW Bag Of Words.

CV Cross-Validation.

FN False Negative.

FP False Positive.

GDPR General Data Protection Regulation.

IDF Inverse Document Frequency.

LR Logistic Regression.

ML Machine Learning.

MLM Masked Language Model.

NB Naïve Bayes.

NLP Natural Language Processing.

PSI Personal Sensitive Information.

RNN Recurrent Neural Network.

SVM Support Vector Machine.

TF Term Frequency.

TF-IDF Term Frequency-Inverse Document.

TN True Negative.

TP True Positive.

1. Introduction

The objective of this section is to provide an introduction to the thesis by presenting the problem domain and its contextual background. We begin by emphasizing the significance and relevance of the chosen topic by providing background and motivation. Subsequently, we provide an outline of the problem statement, research goal, research questions and methods used. Additionally, we outline the contributions made by this thesis and provide a summary of its structure.

1.1. Background and motivation

The use of social media has revolutionized the way people connect online. It provides an easy and inexpensive means of sharing information and expressing opinions. However, this connectivity through social media can lead to potential violations of users' privacy (Baruah, 2012). With the rapid advancement of technology and globalization, protecting personal data has become challenging. Individuals are increasingly sharing personal information publicly and globally, leading to an observed increase in the sharing and collection of personal data (European Parliament and Council of the European Union). During a keynote speech in 2009, the European Consumer Commissioner at the time emphasized the importance of personal data and stated:

> "Personal data is the new oil of the Internet and the currency of the digital world."
>
> — Meglena Kuneva [1]

Personal data has become a valuable resource for targeted marketing, data analytics, and potentially intrusive purposes. However, regulations have been implemented to strike a balance between the interests of businesses and consumers. The General Data Protection Regulation (GDPR) is enforced in European Union law to empower consumers to control

[1] Meglena Kuneva. Meglena Kuneva - European Consumer Commissioner - Keynote Speech - Roundtable on Online Data Collection, Targeting and Profiling. https://ec.europa.eu/commission/presscorner/detail/en/SPEECH_09_156. (Accessed 20.06.2023).

their data and protect their privacy. Penalties, including administrative fines, are a consequence of violating the regulations, which aim to strengthen the enforcement of the rules of the Regulation (European Parliament and Council of the European Union).

While regulations exist to limit data usage by companies, it is essential to recognize the distinction between these imposed restrictions and the information voluntarily shared by users. As stated by the authors in the paper Acquisti et al. (2015), people tend to be unaware of the information they are sharing and how it can be used, and even when they are fully aware of the consequences of sharing, they are uncertain about their preferences. To help consumers navigate the complex task of privacy, a tool that notifies the users can help with this otherwise difficult task. Users often regret what they post on social media, partly due to oversharing or reaching an unintended audience. The findings of the study Sleeper et al. (2013), show that people may be unaware that they are posting something they will later regret, and that the reactions of others to the content contribute to the regret. The study Murmann and Karegar (2021) evaluated a prototype for providing users with privacy notifications before posting on Facebook. The findings suggest that these notifications can enhance users' situational awareness and enable them to make more informed decisions about their posts. Murmann and Karegar suggest that the likelihood of post-related regret and potential repercussions can be minimised by implementing a system that warns or notifies users before sharing something private or sensitive on social media. This study shows how incorporating privacy notifications as a preventive measure to safeguard users' personal information can promote responsible sharing practices.

Furthermore, in the field of Natural Language Processing (NLP), a lot of research is done focusing primarily on widely spoken languages like English. NLP entails developing models and algorithms that allow computers to comprehend and process human language. It has proven useful in various applications such as machine translation, sentiment analysis, and information extraction. The availability of NLP resources in languages like Norwegian compared to English is somewhat more limited. Initiatives like the NB BERT-base model presented in Kummervold et al. (2021) have been introduced to enhance NLP for the Norwegian language catering to the unique linguistics and context of the Norwegian language. By focusing on Norwegian, we aim to address the inequitable treatment of diverse demographic groups and factors in language models (Ramesh et al., 2023) and contribute to Norwegian NLP resources.

1.2. Problem outline

Despite a growing interest in detecting personal or sensitive data, different definitions are used for detecting it. As there are different ways to measure the subjective matter of the sensitive or private data different approaches have been considered to use in this thesis. These approaches include using the visibility or anonymity of the user posting (Correa et al., 2015), utilizing privacy dictionaries to search for sensitive words or terms (Correa, 2015), considering sensitive topics independent of personal identification (Petrolini et al., 2022), or examining the context of a text comprehensively to determine sensitivity (Bioglio and Pensa, 2022). A problem with existing definitions, and the field in itself, is that different definitions can result in obtaining results that do not consider the same problem even when using the same vocabulary (Barker et al., 2009). Utilizing a definition aiming to align with GDPR can therefore be a means to obtaining more aligned findings.

The current state of privacy detection and classification, as far as we could research predominantly focuses on English or other widely spoken languages, potentially leading to an oversight of the unique linguistic characteristics of the Norwegian language. Additionally, social media text is usually noisy and informal, making it challenging for NLP tools that are usually trained on more formal language from newspapers from conventional data sources like news articles, websites, books and papers (Jiang et al., 2022). To the extent of our research, we have not encountered any publicly available labelled Norwegian social media data set specifically designated for private or sensitive content. This further complicates the development of tailored solutions for accurately detecting and classifying private-sensitive content specific to Norwegian social media users.

Consequently, there is a research gap in the field for detecting and classifying private-sensitive content in Norwegian social media aligned with GDPR. Addressing this research gap could enhance privacy protection for Norwegian social media users by providing warnings and guidance to prevent the sharing of private or sensitive content, ultimately contributing to a safer digital environment.

1.3. Goals and research questions

Goal *Enhance privacy protection for Norwegian social media users by developing an automated detection system for private information disclosure.*

The goal of this thesis is to align the definition of private-sensitive content with the General Data Protection Regulation (GDPR) and apply it to investigate techniques for detecting private-sensitive user-generated content on social media platforms written in the Norwegian language. The long-term goal is a tool that would notify a user before sharing information online.

Research question 1 *How can we define private-sensitive content and optimize the labels for classification?*

This research question aims to address two key aspects: defining private-sensitive content and optimizing labels for effective classification in detecting such content in Norwegian user-generated content on social media platforms. By establishing subcategories aligned with the definition of private-sensitive content presented by the General Data Protection Regulation (GDPR), the goal is to enhance the accuracy and performance of classifiers specifically designed to detect private-sensitive content. This research seeks to contribute to the development of effective strategies for privacy protection and awareness.

Research question 2 *How do different classifiers perform on detecting private-sensitive content in Norwegian social media?*

To explore the most effective approach for detecting privacy-sensitive information, this study aims to compare the performance of various classifiers on the task of detecting private-sensitive content in Norwegian social media. We want to consider both conventional classifiers and NB BERT-Base, a transformer based model. To achieve this, two sub-research questions have been formulated:

- **Research question 2.1** *How do conventional classifiers perform in detecting private-sensitive content using the collected data set?*

  For content classification, conventional classifiers, which typically employ conventional machine learning techniques, have been widely used. Important insights can be gained by evaluating how well these tried-and-true methods perform at finding private-sensitive content in Norwegian social media. This analysis enables us to assess the performance of standard classifiers, pinpoint potential drawbacks, and investigate areas for development in the detection of private-sensitive content with conventional classifiers.

- **Research question 2.2** *How does the NB BERT-based model, which is a transformer-based model, perform in detecting private-sensitive content using the collected data set?*

  Transformer-based classifiers, such as those based on state-of-the-art models like NB BERT-Base, have gained significant attention in natural language processing tasks, including content classification. Investigating their performance in detecting private-sensitive content in Norwegian social media can shed light on the capabilities and explore their potential advantages or limitations compared to conventional classifiers.

1.4. Research Method

Our methodology involves a literature review to establish a definition of private-sensitive content. We then annotate scraped data from Norwegian social media based on this definition. We conduct two experiments for detecting private-sensitive content in Norwegian social media. The first experiment involves fine-tuning the transformer-based NB BERT-base model on Norwegian social media content and then training it to detect private-sensitive content using the annotated data. The second experiment involves training conventional classifiers on the annotated data and applying traditional machine learning techniques, including feature engineering and TF-IDF vectorization, to detect private-sensitive content. We compare the performance of these different classification models to determine which approach is more effective for detecting private-sensitive content in Norwegian social media.

Overall, our methodology includes a literature review, data annotation, and two experiments: training and fine-tuning the NB BERT-base model and training conventional classifiers. We then compare their performance to address our research questions.

1.5. Contributions

A detailed description of the contributions will appear in chapter 8 after the presentation and discussion of the results. This section provides a brief summary of the paper's contributions. In summary, the contributions of this study include:

- With the aim of aligning with the definition provided by GDPR, we have developed guidelines for annotation purposes to define what can be considered private-sensitive content.

- We created a labelled data set containing social media posts in Norwegian, addressing the existing gap in resources available for training and evaluating private-sensitive detection models specifically in this language domain. The data set can be acquired upon request.

- We provide an NB BERT-base model fine-tuned on 12 000 unlabeled data entries from two Norwegian channels on Reddit. This model can be useful when dealing with social media text in Norwegian.

- We present a fine-tuned multi-class NB BERT-base model trained on Norwegian language from social media, capable of determining whether the content is private-sensitive, non-private-sensitive or if the content is of unknown sensitivity within the data set created in this thesis.

1.6. Thesis Structure

The thesis is divided into eight chapters. The respective chapters and their principal purpose are listed below.

1. **Introduction**
   In the introduction, the background and motivation for the thesis are presented followed by the problem outline and the goal and research questions. In addition, the methods used, contributions and the thesis structure is outlined.

2. **Background Theory**
   This chapter presents background theory linked to the topics and methodologies covered in the thesis.

3. **Related Work**
   This chapter reviews the literature on defining private and private-sensitive content, including the GDPR's definition. It also explores methods for detecting private-sensitive content and the challenges involved in detecting it in Norwegian social media.

4. **Defining private-sensitive content**
   In this section, we outline the process of creating the definition of private-sensitive content for the annotation guidelines. Additionally, we present the annotation guidelines themselves.

5. **Data set Creation**
   This chapter walks through the process of creating the labelled data set for use in the experiments, as well as an inspection and analysis of the data.

6. **Methods and experiments**
   This chapter presents the methods utilized in the experiments and presents the experimental plan and setup.

7. **Results and Discussion**
   The chapter starts by presenting the experiment results with a discussion of these results. This is followed by a general discussion that includes limitations, a comparison of results from different experiments, a comparison to related work, and a discussion of the definition and labelled data set created.

8. **Conclusion and Future Work**
   In this final chapter, the work in the thesis is concluded, the contributions will be summarized and suggestions for future work are proposed.

2. Background Theory

This chapter provides the necessary background theory for the research conducted in this thesis. First, we present an overview of the fundamental principles of machine learning, with a particular emphasis on the supervised machine learning method known as classification. After this, we present the concept of hyperparameter tuning and evaluation of classification models. Finally, we present statistical measures for describing the level of agreement between raters which in this thesis are used to evaluate the agreement between the annotation of our labelled data set. Certain portions have been directly quoted from the specialization project Borgen and Zachariassen (2022) preceding this thesis, and will be explicitly stated as such.

2.1. Natural Language Processing

Natural Language Processing (NLP) is a field that encompasses a range of computational techniques aimed at analyzing and representing naturally occurring texts to achieve human-like language processing for various tasks and applications (Nadkarni et al., 2011).

2.2. Machine Learning

Machine learning addresses how to build computers that improve automatically through experience. A Learning problem in Machine learning can be defined as the problem of improving some type of performance for executing a task through some training experience (Jordan and Mitchell, 2015). Depending on the available data and the given problem, we distinguish between three types of Machine Learning: Supervised, unsupervised, and reinforcement learning (Janiesch et al., 2021). Supervised learning requires an input/output pair containing both the input data and labelled answers for the output data. Unsupervised learning is supposed to detect input data patterns without known labels. Reinforcement learning does not use input/output pairs like supervised learning. Instead, the system's current state is described, and a goal is provided along with a list of allowable actions and the corresponding environmental constraints for their outcomes. The process of achieving the goal is based on the principle of trial and error (Janiesch et al., 2021). In this thesis, we focus on supervised machine learning.

2.2.1. Supervised Machine Learning

Supervised learning is a widely used training paradigm in machine learning. In contrast to unsupervised learning, where instances are unlabeled, supervised learning involves instances with known labels (Kotsiantis et al., 2007). The objective is to build a model that maps input data to their corresponding expected outputs. When training a model with supervised learning, the training set is a collection of pairs (x, y) where x is the input data, and y is the labelled answer for the output data. The training set is used to train the ML model. If the ML model is successfully trained, it can predict the target y based on unseen input data x (Janiesch et al., 2021). In supervised learning, we can further distinguish between classification, where the prediction is a categorical class, and regression, where the prediction is a numeric value (Janiesch et al., 2021). An example of a classification problem can be to predict if a review is negative or positive, and an example of a regression problem can be to predict a person's height based on different features.

2.2.2. Classification

In this thesis, we will focus on classification techniques. Classification is a supervised learning method where the input data have a corresponding label serving as the expected output. Classification can be divided into two categories: single-label and multi-label classification. Within single-label classification, there are subcategories known as binary classification and multi-class classification. In binary classification, each instance is classified into one out of two classes. Each instance is allocated to one class from a set of mutually incompatible classes in multi-class classification. As a result, in single-label classification, each instance is assigned to a single correct class. In multi-label categorization, however, each instance can be assigned to multiple classes (Er et al., 2016).

2.3. Classifiers

Classifiers are algorithms trained to determine the class to which an input belongs. We now present the classifiers we will use in our experiments.

2.3.1. Decision Tree Classifier

The decision tree classifier is an algorithm commonly used for solving classification tasks. The general idea is to divide a complex decision into smaller and simpler ones, creating multiple yes-or-no decisions.

Each decision node represents a feature, and each branch represents a value the node can assume (Kotsiantis et al., 2007). The feature causing the most significant separation between the observations in the left and right nodes is chosen, resulting in the tree structure depicted in figure 2.1. The decision tree is directed and consists of one root node with no incoming edges. Nodes with no outgoing edges are referred to as leaves Rokach and Maimon (2005).

The instances begin at the root node and are sorted down to the leaf nodes based on their feature values (Kotsiantis et al., 2007). The leaves represent the class to which the instance belongs, representing the final classification. For an in-depth explanation of decision trees, see the work of Murthy.

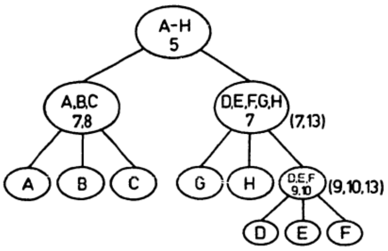


Figure 2.1.: A simple decision tree
Swain and Hauska (1977)

2.3.2. Random Forest Classifier

The Random Forest is an ensemble learning method known to reduce a classifier's variance and improve the decision system's accuracy and robustness (Cutler et al., 2012). During training, it generates a large number of de-correlated decision trees. It obtains a class vote from each decision tree and then classifies based on the majority vote (Hastie et al., 2009a).

Random Forest utilizes bootstrap aggregation, also called bagging, which is a technique for reducing the variance of the decision tree classifier (Hastie et al., 2009a). Bagging enables the different trees to be randomly sampled from the training set. Assuming a training set of size n, each decision tree will sample a training set of size n. The training sets are sampled with replacement, meaning elements can be selected multiple times in the same sample. This can result in some elements missing while others appear several times in the sample. The goal is to create different training sets for each decision tree to provide distinct tree structures.

To obtain uncorrelated trees, each decision tree is randomly assigned distinct subsets of the features, a process known as feature bagging. This means that each tree consists of different features, with no feature overlap between the trees. The idea is that different features result in different splitting and less correlated trees. For more information about Random Forest, see the work of Hastie et al. (2009a).

2.3.3. Naive Bayes Classifier

Naive Bayes is a family of Bayes theorem-based probabilistic classifiers. It calculates the posterior probability of an event using Bayes theorem combined with a naive simplification assumption (Russell and Norvig, 2010). The naive assumption is that it presumes independence between the features/evidence (Murphy et al., 2006).

The Bayes theorem equation is represented in equation 2.1 and defines the posterior probability of an event. Given that B is true, the equation shows the likelihood of event A occurring.

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)} \tag{2.1}$$

Naive Bayes can predict the class variable using the observed features in the training set. It is particularly useful when the feature space size is large, making density estimation unappealing. Hastie et al. (2009b). A Multinomial Naive Bayes classifier is a specific instance of a naive Bayes classifier that uses a multinomial distribution for each feature. The naive assumption: that the features are conditionally independent of each other, allows us to use equation 2.2 (Russell and Norvig, 2010).

$$P(C|f_1, ..., f_n) = \alpha P(C) \prod_i P(f_i|C) \tag{2.2}$$

The naive assumption allows us to take the product of the conditional probabilities of each feature f_i given the class C.

2.3.4. Support Vector Machine Classifier

A Support Vector Machine (SVM) classifier's objective is to find the optimal hyperplane for classifying data points. A hyperplane is a distinguishing boundary that separates data points, indicating the separation of two categories. The ideal hyperplane is the one with the greatest margin between the hyperplane and the data points in either category. It is called linear SVM if the data can be separated by a straight line or a hyperplane (Suthaharan and Suthaharan, 2016). Figure 2.2 show two different hyperplanes, where the margin to the right (b) is greater than the margin to the left (a). As a result, the ideal hyperplane is the one to the right (b). For a more detailed description of SVM, see the work of Yue et al..

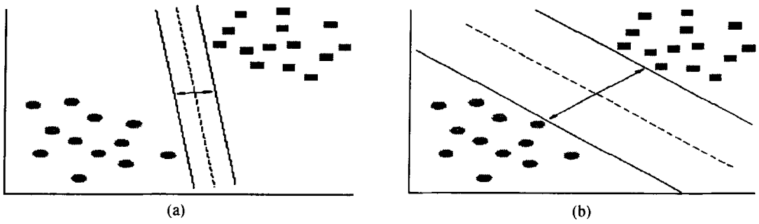


Figure 2.2.: The optimal margin between two planes.
Yue et al. (2003)

2.3.5. Multinomial Logistic Regression Classifier

Logistic Regression (LR) describes the relationship between several independent variables (x1, x2, .., xn) and a dependent variable Y. LR uses the logistic function shown in equation 2.3. The logistic function transforms a linear combination of the independent variables into a value between 0 and 1 to represent the estimated probability of Y (Richardson, 2011).

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{2.3}$$

Multinomial LR is an extension of regular LR that allows for more than two categories or labels. It calculates the probability for each class and assigns it to the class with the highest probability. Same as for LR, Multinomial LR uses maximum likelihood estimation to find the probability of categorical membership (Kwak and Clayton-Matthews, 2002).

2.4. TF-IDF Vectorizing

The following paragraph introducing the principle of TF-IDF Vectorizing has been directly quoted from the specialisation project (Borgen and Zachariassen, 2022).

A term frequency-inverse document is a text vectorizer that turns text into numerically encoded vectors. The term "term frequency" (abbreviated "TF") refers to the frequency a certain term appears in a text or document. The distinct terms in the rows and the documents in the columns make up the matrix that represents the term frequency. Document frequency, or DF, is a measure of a word's frequency. If the word "and" occurs frequently in all of the papers, the DF will classify it as a common phrase and won't let it be used to distinguish between the documents. The IDF steps in at this point. IDF is the term's weight. If the term appears frequently in several documents, its significance is reduced. The weight would be zero if the word "and" appeared in every single one of the n documents. A word will have more weight if it is uncommon. Getting meaningful numerical features that represent text in natural language is the goal of TF-IDF vectorizing.

2.5. Transformers and BERT

2.5.1. Transformers

The Transformer model, introduced by Vaswani et al. (2017), is a groundbreaking sequence transduction model that relies solely on self-attention. The original Transformer model was designed for machine translation with an encoder-decoder architecture but has achieved state-of-the-art performance in a range of NLP tasks.

The Transformer model uses an attention mechanism called self-attention. The self-attention mechanism captures contextual information and dependencies between words in a sequence. It allows each word/token in the input sequence to attend to other words/tokens in the same sequence. Consider the following example to demonstrate the simplified principle of self-attention: "She is going to the restaurant to eat". Assuming that we focus on the word 'eat', self-attention would be used to calculate the relevance or importance of each word in relation to the word 'eat'. The other words in the sentence would be assigned scores based on their contextual significance in the same way. The Transformer model can capture these relationships and dependencies through self-attention. A benefit of attention mechanisms is that it allows modelling the dependencies regardless of the distance in the input or output sequences. This attention mechanism enables the Transformer model to draw global dependencies between the input and output (Vaswani et al., 2017).

Another benefit of the Transformer model is its ability to parallelize computations and reduce training time significantly. It replaced the recurrent layers, commonly used in encoder-decoder architectures, with multi-head self-attention (Vaswani et al., 2017). This replacement not only removes the sequential computation of recurrent layers but also enables parallelization through factors such as attention mechanisms.

Figure 2.3 illustrates the architecture of the Transformer model. The architecture consists of encoders and decoders. The encoder consists of N identical layers. Figure 2.3 displays that each layer consists of two sub-layers: multi-head self-attention and a feed-forward network. By working together, the encoder and decoder can be used to perform tasks such as language translation. More comprehensive coverage of the Transformer model can be found in the work of Vaswani et al. (2017).

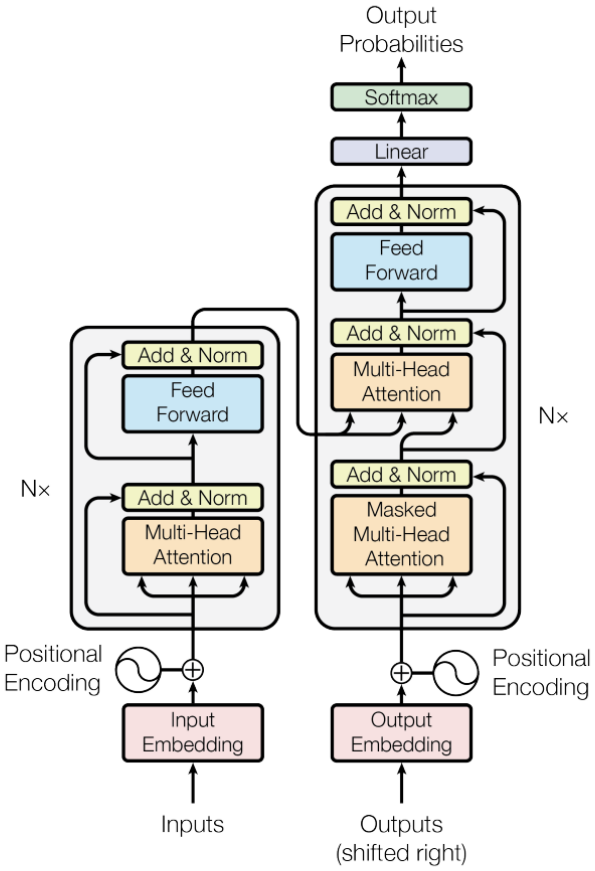


Figure 2.3.: logistic rerchitecture
(Vaswani et al. (2017))

2.5.2. BERT

BERT stands for Bidirectional Encoder Representations from Transformers, introduced in Devlin et al. (2018). It uses bidirectional representations to consider both left and right contexts. It uses the encoder part of the transformer architecture, including the self-attention mechanism described in subsection 2.5.1.

BERT follows a two-step process: pre-training and fine-tuning. In the pre-training phase, BERT is trained on unlabeled text using Next Sentence Prediction or Masked Language Modeling. This training involves predicting missing words by considering the left and right contexts of each word in the sequence. During fine-tuning, the pre-trained parameters are fine-tuned by using labelled data from the downstream task. A notable advantage lies in the ability of task-specific models to derive benefits from the richer pre-trained representations, even when working with limited amounts of downstream task data. The original BERT model was pre-trained using a large corpus consisting of 16GB of uncompressed text (3,300 million words) where the majority contained content from English Wikipedia (Devlin et al., 2018). Fine-tuning is relatively inexpensive compared to pre-training. Fine-tuning a BERT model can therefore be a cost-effective approach because it leverages an already pre-trained BERT model.

After the introduction of BERT, several other BERT models have been built. One of them is the Robustly Optimized BERT (RoBERTa) model which is pre-trained using five English-language corporas. This results in a pre-trained language model trained on a total of over 160 GB of uncompressed text (Liu et al., 2019).

2.5.3. NB BERT

The NB BERT model is a BERT-based model that is built using a large Norwegian corpus. To build the pre-trained language model, the authors use the original BERT architecture and pre-train it using MLM. They pre-train the model using approximately 109 GB of uncompressed text. The corpus consists of Norwegian Books, Parliament Documents, Newspaper Scans, Online Bokmål Newspapers, Wikipedia articles, and more (Kummervold et al., 2021).

2.5.4. WordPiece Tokenizer

The WordPiece tokenizer is often used in BERT models. It plays a crucial role in tokenizing text by splitting words into smaller subword tokens. One of the key advantages of the WordPiece tokenizer is its ability to handle unseen or out-of-vocabulary (OOV) words. To illustrate this concept, let us consider a straightforward example. Assuming the word "by" is known, but the words "byfolk" and "folk" are not included in the vocabulary. Rather than treating the OOV word "byfolk" as an unknown entity, the tokenizer can divide it into two subtokens: "by" and "##folk". This way, the model can still leverage the known subword "by" to understand the word's context and meaning. This approach can improve the handling of complex, unseen and rare words.

2.5.5. Fine-Tuning

Pre-training language models on large-scale unlabeled text data, and then fine-tuning the model to a downstream task, has made a breakthrough in many NLP tasks (Xu et al., 2020). Fine-tuning involves taking a pre-trained language model trained on a large corpus of text data and then fine-tuning it to make it task-specific (Liu et al., 2023). Overall, fine-tuning pre-trained language models such as BERT is an effective way that gives state-of-art results in various downstream tasks (Xu et al., 2020). Compared to training a model from scratch, it can save time and computational resources.

2.5.6. Masked Language Modeling

In Masked Language Modeling (MLM), some tokens in a sequence are masked. The objective is to predict the original vocabulary for the masked tokens based on its context (Devlin et al., 2018). By training the model to fill in the masked tokens, the model learns to understand and capture the relationships between words in the sequence. For example, if we consider the sentence: "Jeg har alltid på meg [MASK] på hodet" During the pre-training phase, the model might be trained to predict the missing word, which is masked as "[MASK]", based on the surrounding words. In this case, the model could predict "lue" or "hår" as the masked word to complete the sentence correctly.

2.5.7. Domain Adaption

Domain adaptation addresses the challenge of leveraging labelled data from a similar domain as the target domain to build classifiers for unseen or unlabeled data in a target domain (Csurka, 2017). It tackles the high resource costs associated with data annotation by utilizing large volumes of available unlabeled data (Csurka, 2017). One way to achieve domain adaption for Bert models is to fine-tune a model using Next Sentence Prediction or Masked Language Modeling before making it task-specific (Devlin et al., 2018).

2.6. Hyperparameter Tuning

An important phase in developing machine learning models is hyperparameter tuning. It involves finding the optimal values for the hyperparameters of a model, which are parameters that are not learned directly from the data but are set before the learning process begins. Some hyperparameter techniques are manual search, random search and grid search. Manual search is the most basic method and relies on guessing and domain knowledge to find satisfactory hyperparameters (Zahedi et al., 2021). In Grid search, we set a set of values for the hyperparameters and test every combination. Random search chooses the parameter combinations from the set values random until its resources are used (Zahedi et al., 2021). It requires fewer resources than grid search but does not necessarily find the optimal hyperparameter combination within the value range. In this thesis, we use grid search.

2.7. Grid Search

Grid search is a systematic approach to hyperparameter tuning that does a complete search over a given subset of the hyperparameters within a predefined search space (Liashchynskyi and Liashchynskyi, 2019). For example, if we have two hyperparameters, A and B, and we specify three values for A (A1, A2, A3) and two values for B (B1, B2), the grid search will evaluate the model performance for all possible combinations: (A1, B1), (A1, B2), (A2, B1), (A2, B2), (A3, B1), and (A3, B2). For each combination of hyperparameter values, the model is trained and evaluated. The optimal collection of hyperparameter values is then chosen as the combination that produces the best performance. Grid search is complete brute-force and can be time-consuming (Liashchynskyi and Liashchynskyi, 2019), especially when the search space is large or training a model is resource-intensive. However, it guarantees to explore the entire search space and find the best hyperparameter within this space.

2.8. Evaluation of Classification Models

This section provides an overview of metrics used to evaluate classification models, followed by approaches to improve the model's performance on unseen data.

2.8.1. Evaluation Metrics

When evaluating the model's performance on classification tasks, the predicted label is compared to the actual label of the instance. Four experimental outcomes can occur:

- True Positive (TP): The model correctly predicts the label.
- False Positive (FP): The model predicts the label but should have been assigned to a different category.
- True Negative (TN): The model correctly does not predict the label, as it is not the correct category.
- False Negative (FN): The model fails to predict the label, even though it should have been assigned to that category.

A confusion table is displayed in Table 2.1 to illustrate these experimental outcomes further. The two columns show whether the label is true or not for the specific instance. The two rows indicate whether the model predicted that the instance belonged to the label or not. True positives indicate a relevant instance, and the row "predict label" shows what the model believes is relevant.

Table 2.1.: Confusion Matrix

		Actual Class	
		True	False
Predicted Class	Predict label	True Positive (TP)	False Positive (FP)
	Not Predict	False Negative (FN)	True Negative (TN)

Precision

Precision can be defined as the probability that an object is relevant, given that it has been returned by the system (Goutte and Gaussier, 2005). It is calculated as the ratio of true positive (TP) predictions to the sum of true positive and false positive

(FP) predictions, shown in equation 2.4. The precision decreases as the number of false positives increases and is especially useful when the cost of false positives is high.

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2.4}$$

Recall

Recall may be defined as the probability that a relevant object is returned by the system (Goutte and Gaussier, 2005). Recall is a performance metric that reflects the proportion of true positives to everything that should have been predicted as positive. It is calculated as the ratio of true positive (TP) predictions to the sum of true positives and false negatives (FN) predictions, shown in equation 2.5. The recall decreases as the number of false negatives increases and is especially useful when it is essential to identify all positive instances.

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{2.5}$$

F1-Score

A model may predict few instances and obtain a low recall, but a 100 % accuracy. In contrast, it may predict all instances and get a 100 % recall, but a low precision. The F1 score combines both measures (Russell and Norvig, 2010). The score is calculated as the harmonic mean of precision and recall, shown in equation 2.6. The score ranges between 0 and 100 % where a high F1 score indicates the better model. This score can be used to compare classification models.

$$F_1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{2.6}$$

Macro-Average F1-Score

The macro average F1-score is a modified metric for multi-labelled classification. The macro-average is computed by first computing the f1 score of each class and then calculating the average over all classes. Each class is weighted equally, capturing effectiveness while not favouring larger classes. To better capture the smaller classes in an unbalanced data set, the macro-average F1-score is preferable (Christopher D. Manning, 2008).

Accuracy

Equation 2.7 shows the computation of accuracy. The sum of true positives and true negatives is divided by all instances in the data set. The average accuracy can be used to evaluate multi-classification models.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.7}$$

2.8.2. Overfitting

Overfitting poses a fundamental challenge in supervised machine learning, hindering the ability to obtain perfectly generalized models that fit well on both observed data in the training set and unseen data. The presence of noise, limited training set size, and classifier complexity are all factors contributing to the occurrence of overfitting (Ying, 2019). Despite performing well on the training data, the model's generalization to new unseen data is ineffective. The accuracy of algorithms eventually stops improving and can even decline after a certain point due to the model picking up arbitrary noise and considering it as a meaningful concept. A strategy to address this issue is called early stopping. This strategy aims to determine the ideal number of epochs by identifying the point just before the performance declines, striking a balance between underfitting and overfitting. Early-stopping and other strategies to mitigate overfitting are covered in depth by Ying.

2.8.3. K-Fold Cross-Validation

K-fold cross-validation is a technique used to estimate the performance of a model. The technique involves dividing the data set into K subsets or folds and then iteratively training and testing the model K times. In each iteration, one fold is designated as the test set, while the remaining K-1 folds are used for training (James et al., 2013). The evaluation metrics obtained from each iteration are combined to provide an overall performance estimate. K-fold cross-validation helps monitor overfitting/underfitting and facilitates hyperparameter tuning by providing a representative evaluation of the model's generalized performance (Hastie et al., 2009c).

2.9. Statistical Measures for Agreement Among Annotators

We now present two statistical measures used in our thesis, which can be used to evaluate agreement between annotators.

2.9.1. Cohen's Kappa

Cohen's kappa provides a measure of the agreement between two raters. The equation for Cohen's kappa is shown in Equation 2.8.

$$\kappa = \frac{P_o - P_e}{1 - P_e} \tag{2.8}$$

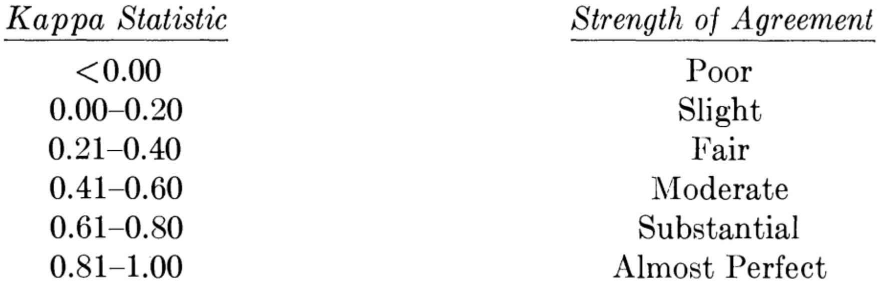
P_o denotes the observed proportion of pairwise agreement between the raters, while P_e is the proportion agreement expected by chance (Kvålseth, 1989). If both annotators agree perfectly, the P_o would equal 1, and we would obtain a Cohen's kappa of 1.

2.9.2. Fleiss' Kappa

Fleiss' kappa is a generalisation to Cohen's kappa and allows for measuring the agreement between more than two raters (Falotico and Quatto, 2015).

2.9.3. Benchmarks for Describing the Level of Agreement

Figure 2.4 is derived from the work of Landis and Koch and displays labels assigned to various kappa ranges. As these divisions are arbitrary, the authors note that this is primarily useful as a benchmark for discussion.



Kappa Statistic	*Strength of Agreement*
<0.00	Poor
0.00–0.20	Slight
0.21–0.40	Fair
0.41–0.60	Moderate
0.61–0.80	Substantial
0.81–1.00	Almost Perfect

Figure 2.4.: Benchmarks for describing the level of agreement. Landis and Koch (1977)

3. Related Work

This chapter provides an overview of the related work for defining private-sensitive content. We introduce previous approaches and methods utilized for the detection of private-sensitive content. Lastly, we present relevant research on the challenges associated with detecting private-sensitive content, particularly within the unique context of Norwegian social media.

3.1. Definition of Private-Sensitive Content

This section begins by presenting the related work concerning the definition of privacy. We also examine how the General Data Protection Regulation (GDPR) defines personal and sensitive data. Furthermore, we explore how other related works, specifically those focused on detecting private-sensitive data, have defined this concept. Most of this section, which deals with the definition, is directly taken from the specialization project that preceded this master's thesis (Borgen and Zachariassen, 2022).

Research on privacy has increased as technology made it easier than ever to monitor and harvest information. According to Westin, the increased privacy threats to society that emerged after the end of the Second World War are due in part to better technology and the generally low cost and convenience with which electronic devices may be bought (Westin, 1968). In later years, regulations protecting individuals' privacy have been put into effect, demonstrating the issue's attention. However, although there are laws limiting how businesses can use consumers' data, the question of what qualifies as privacy is still up for debate.

A classic definition of privacy is Westin's definition: 'Privacy is the claim of individuals, groups, or institutions to determine for themselves when, how, and to what extent information about them is communicated to others' (Westin, 1968). This definition relates to the control aspects of privacy but also reflects how individuals are perceived by society. Warren and Brandeis defined privacy as the right to be left alone. This aspect of privacy reflects a distinction between the individual and the rest of society. The component of anonymity and the choice of what to share and not is therefore an important part of privacy. What makes this complex is that people have different perceptions of what they consider sensitive information, and how much they wish to share or keep to themselves. A general definition covering all aspects can therefore be challenging to obtain.

According to GDPR in Article 4, paragraph 1, personal data is defined as any information related to an identified or identifiable natural person; where an identifiable person is one that can be directly or indirectly identified. In particular, this includes identifiers such as the name, location data, identification number and factors specific to the physical, genetic, mental, economic, cultural, physiological, or social identity of that natural person (European Parliament and Council of the European Union). Each of these bits of information may not be sufficient to identify a specific individual. However, when aggregated, they may be enough to identify an individual and compromise personal information. Because "any information" is included in the definition, GDPR requires that the term "personal data" should be interpreted as broadly as feasible. This is also suggested in European Court of Justice case law when more subtle personal data issues may arise, such as work time recorders that include details about an employee's start and end timings, as well as breaks and non-work time periods (European Parliament and Council of the European Union). In the work Tesfay et al. (2016), the authors criticize the notion of "any information" as being overly broad for analyzing user privacy. Instead, they advocate for concentrating on sensitive data.

According to GDPR recital 51, some personal data is particularly sensitive in relation to an individual's fundamental rights and freedom. Sensitive data is described as a special category of personal data. In Article 9, Paragraph 1 of the GDPR, it is stated that special categories of personal data should be protected. This includes personal data that reveal political opinions, religious or philosophical beliefs, trade union membership, as well as the processing of genetic data, biometric data for uniquely identifying individuals, data concerning health, racial or ethnic origin, and data concerning a person's sex life or sexual orientation. In addition, article 10 of the GDPR highlights that the processing of personal data relating to criminal convictions and crimes must be protected (European Parliament and Council of the European Union). This strikes a balance between the legitimate interests of authorities and the fundamental rights of the individuals involved.

In the work Correa et al. (2015), the authors introduce the notion of anonymity sensitivity to measure the sensitivity of content posted on social media sites. Anonymity sensitivity captures whether the annotators prefer to post the message anonymously. Hence, this is a binary question centred on the identity of the message's writer. To measure anonymity sensitivity, they use an anonymity sensitivity score which captures the probability that an annotator would consider the text anonymous. They observe that the annotators have different perceptions of sensitivity while annotating the messages. The results show that messages posted on anonymous sites have a higher anonymity score than content posted on non-anonymous sites such as Twitter. There were considerable linguistic differences between text written on public and anonymous sites, with negative emotionally loaded phrases being used more frequently on anonymous sites. They suggest that it may be possible to develop classifiers that automatically differentiate between anonymity sensitive and non-anonymity sensitive data. According to the authors, such classifiers can be useful in protecting users' privacy. As a result, their understanding of privacy is inextricably related to the writer's awareness of anonymity as well as the decision to conceal or reveal their identity.

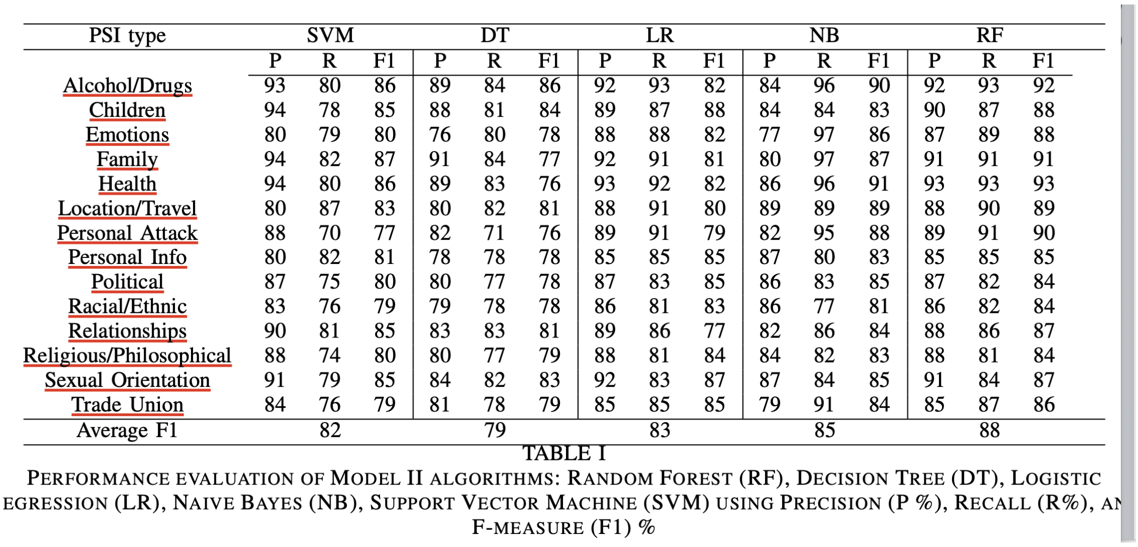
The authors argue in their work Bioglio and Pensa (2022), that the technique presented by Correa et al. is too simplistic since it underestimates the quantity of sensitive personal information people share. They emphasize that anonymity and sensitivity are not inextricably linked, and they point out that a message might employ emotionally charged phrases without exposing private information, while messages using less emotionally loaded words can reveal a lot of private information. This makes the task of detecting private-sensitive text even more complex.

3.2. Previous Approaches and Methods for Private-Sensitive Content Detection

This section first introduces previous approaches for detecting private-sensitive content. Finally, we present some commercial tools launched for the purpose of detecting personal data and private sensitive data. We note that the first paragraph in this section is directly taken from the specialization project that preceded this master's thesis Borgen and Zachariassen (2022).

This paragraph is taken from the specialization project Borgen and Zachariassen (2022) preceding this thesis. In the work, Tesfay et al. (2019), the authors offer a privacy bot that can detect and categorize private, sensitive information into 14 different categories, providing a more in-depth examination of the type of private information disclosed. The categories are based on Art. §9 in GDPR. They divide the classification problem into two categories: a binary classification problem and a multi-label classification problem. To determine if a text contains private-sensitive information or not, they first apply binary classification. The nature of the private-sensitive information is then further classified using multi-label classification. Both the binary classification problem and the multi-label classification problem are tackled using Random Forest, Decision Tree, LR, Naive Bayes, and SVM. To extract features for the classification tasks the authors employ TF-IDF. As shown in Figure 3.1, the decision tree had the best performance metrics. Since just one social media platform, namely Twitter, provided the data used, PrivacyBot may not work as well on other platforms, such as Facebook and Instagram.

In the work Petrolini et al. (2022), the authors use a transformer deep-learning model, to develop classifiers, capable of detecting whether a document contains sensitive data. In



PSI type	SVM			DT			LR			NB			RF		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Alcohol/Drugs	93	80	86	89	84	86	92	93	82	84	96	90	92	93	92
Children	94	78	85	88	81	84	89	87	88	84	84	83	90	87	88
Emotions	80	79	80	76	80	78	88	88	82	77	97	86	87	89	88
Family	94	82	87	91	84	77	92	91	81	80	97	87	91	91	91
Health	94	80	86	89	83	76	93	92	82	86	96	91	93	93	93
Location/Travel	80	87	83	80	82	81	88	91	80	89	89	89	88	90	89
Personal Attack	88	70	77	82	71	76	89	91	79	82	95	88	89	91	90
Personal Info	80	82	81	78	78	78	85	85	85	87	80	83	85	85	85
Political	87	75	80	80	77	78	87	83	85	86	83	85	87	82	84
Racial/Ethnic	83	76	79	79	78	78	86	81	83	86	77	81	86	82	84
Relationships	90	81	85	83	83	81	89	86	77	82	86	84	88	86	87
Religious/Philosophical	88	74	80	80	77	79	88	81	84	84	82	83	88	81	84
Sexual Orientation	91	79	85	84	82	83	92	83	87	87	84	85	91	84	87
Trade Union	84	76	79	81	78	79	85	85	85	79	91	84	85	87	86
Average F1		82			79			83			85			88	

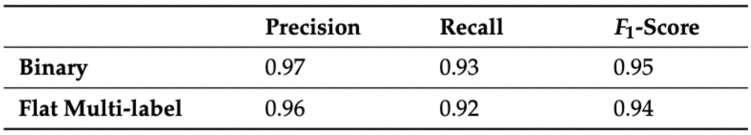
TABLE I

PERFORMANCE EVALUATION OF MODEL II ALGORITHMS: RANDOM FOREST (RF), DECISION TREE (DT), LOGISTIC EGRESSION (LR), NAIVE BAYES (NB), SUPPORT VECTOR MACHINE (SVM) USING PRECISION (P %), RECALL (R%), AN F-MEASURE (F1) %

Figure 3.1.: 14 PSI categories and the performance of each model. Tesfay et al. (2019)

contrast to our work, their motivation for detecting sensitive content is for companies to have better knowledge of their data. To simplify the problem of detecting sensitive data, they consider four main classes of sensitive topics as their definition of private-sensitive data. The sensitive topics are derived from GDPR's definition of sensitive data and constitute politics, religion, health and sexual habits. They reduce the problem of detecting sensitive data to the problem of detecting sensitive topics. However, simplifying the task of detecting sensitive data to detecting sensitive topics alone may overlook the requirement of establishing a link to a natural person, which is necessary under the GDPR for data to be classified as sensitive. The authors argue that this simplification can potentially reduce false negatives since the link to the natural person could be written in other parts of a document or may be represented by the sender/recipient of an email.

Petrolini et al. compare two BERT-based model architectures: a flat multi-label model and a hierarchical model. The flat multi-label model utilizes a single-classifier multi-label architecture where the output indicates the likelihood of the input belonging to each of the four topics. The input is considered sensitive if at least one of the topics obtains a likelihood exceeding 50%. The hierarchical model includes two classifiers: a binary classifier and a multi-class classifier. The binary classifier determines whether the input contains a sensitive topic or not. The multi-class classifier is activated only if the binary classifier determines that the input includes a sensitive topic. It then determines which of the sensitive topics the sentence belongs to. They compare the binary classifier of the hierarchical model and the flat multi-label model on the task of sensitive-topic detection by treating the union of the four topics as a sensitive class. The models were trained on a data set consisting of 47,539 samples and evaluated on a test set containing 2,400



	Precision	Recall	F_1-Score
Binary	0.97	0.93	0.95
Flat Multi-label	0.96	0.92	0.94

Figure 3.2.: The results of the sensitive-topic detection for the flat-multi-label model and the binary classifier in the hierarchical model.
Petrolini et al. (2022)

samples. Figure 3.2 presents the results, including precision, recall, and F1 score. The binary classifier of the hierarchical model achieves the highest F1 score of 0.95, and the flat multi-label achieves an F1 score of 0.94.

In the work Bioglio and Pensa (2022), the authors adopt a definition of privacy-sensitive content that strongly emphasises self-disclosure and the disclosure of others. As a result, they argue that simply addressing sensitive topics or using emotionally charged language is insufficient to classify something as privacy-sensitive.

To assess the sensitivity of user-generated posts on social media, Bioglio and Pensa utilize a data set of 9917 Facebook posts. Due to restrictions set by the Facebook API, they utilize a data set collected by Facebook for research purposes over ten years ago. Their annotation process includes 12 annotators, each labelled approximately 2480 entries into one of the four labels: sensitive, non-sensitive, unknown and unintelligible. Four groups consisting of three annotators labelled the data set, and the agreement between the groups labelling the same entries obtain a Fleiss' kappa varying between 0.22 and 0.42. They introduce a SENS2 and SENS3 data set containing entries where at least two (SENS2) and three (SENS3) annotators agree. SENS2 contains 8765 entries, where 3336 are labelled as sensitive. The SENS3 data set contains 4046 entries, where 1444 are labelled as sensitive. The authors noted that this shows that the task of labelling can be challenging.

Bioglio and Pensa utilize four distinct text classifiers to analyze the embedded text. These classifiers consisted of a Convolutional Neural Network (CNN), a Recurrent Neural Network (RNN) with gated recurrent unit (GRU) nodes, an RNN with long short-term memory (LSTM) nodes, and BERT. Additionally, they employ bag-of-words (BOW) models with feature extraction based on TF-IDF. Among the classification models they use, both the BERT and the RNN models demonstrate a considerable performance improvement compared to the other models. The authors propose that this performance disparity could be attributed to the ability of transformer-based language models, like BERT, to capture the context of words and sentences effectively. Furthermore, the authors demonstrate that dictionary-based or bag-of-words approaches are less effective in directly detecting privacy sensitivity. The BERT model achieves a macro F1 score of 0.89 on the SENS3 data set and a macro F1 score of 0.78 on the SENS2 data set. In addition, they achieve an F1 score of 0.85 on the SENS3 data set and an F1 score of 0.73 on the SENS2 data set specifically for the sensitive class.

Multiple commercial tools have been launched as awareness of the importance of detecting sensitive data has grown. These tools include Microsoft Purview Information Protection, IBM Security Discover and Classify, and Concentric AI's Data Categorization and Classification Platform. Microsoft Purview Information Protection provides detection of sensitive and business critical data [1]. Concentric AI offers a Data Categorization and Classification Platform [2]. IBM Security Discover and Classify is a product provided to detect sensitive data among different platforms, at the enterprise level [3]. In their documentation, IBM explicitly states that their product Natural Language Understanding (NLU) has limited support for Norwegian compared to English [4]. Unfortunately, because these tools are commercial products, conducting an in-depth study of their approach and performance in the Norwegian language and on social media content can be difficult.

3.3. Challenges in Detecting Private-Sensitive Content in Norwegian Social Media

The detection of private-sensitive content in Norwegian social media poses several challenges. In this section, we explore the specific challenges associated with detecting private sensitive content in Norwegian social media and examine the related work conducted to tackle these challenges.

As described in Section 3.1, scholars struggle to reach a consensus on the definition of private-sensitive content, which may lead to variations in detection models. Different definitions can result in the detection of different sensitive information, making the task of accurately detecting such information challenging. Another challenge in detecting private sensitive content, as highlighted by Bioglio and Pensa, is identifying sensitivity solely based on the presence or absence of specific terms or topics because context can play a fundamental role. Bioglio and Pensa describe how even if a text does not contain sensitive terms or topics, it can disclose sensitive information, and the other way around,

[1] Microsoft. Microsoft purview information protection. https://www.microsoft.com/en-us/security/business/information-protection/microsoft-purview-information-protection. (Accessed 20.06.2023).

[2] Concentric AI. Data categorization and classification platform. https://concentric.ai/use-cases/data-discovery-and-classification/ (Accessed 20.06.2023)

[3] IBM. IBM security discover and classify. https://www.ibm.com/products/ibm-security-discover-and-classify. (Accessed 20.06.2023)

[4] IBM. Language support. https://cloud.ibm.com/docs/natural-language-understanding?topic=natural-language-understanding-language-support&mhsrc=ibmsearch_a&mhq=language+support#norwegian-bokmal. (Accessed 20.06.2023)

even if the text does contain a sensitive topic or term it doesn't necessarily disclose sensitive information.

To train a model for detecting private-sensitive content, obtaining a data set of sensitive data is necessary. As stated by Petrolini et al., finding such a data set can be challenging as they did not find any widely accepted public data set of sensitive data. Our research did not uncover any public data sets for private-sensitive content in Norwegian Social Media. The lack of available data poses a challenge for training and building the models. Furthermore, the nature of social media introduces additional complexity as social media text is usually noisy and informal, making it challenging for NLP models that are usually trained on more formal language (Jiang et al., 2022).

Researchers have explored domain adaptation techniques to address these challenges of limited labelled data and working with domain-specific data. In the work Rietzler et al. (2019), the authors explore the two-step process of fine-tuning the model to the target domain before making it task specific to improve the model's understanding of the language patterns specific to the target domain. They refer to this process as domain adaption. The results Rietzler et al. obtain, show that fine-tuning significantly improves the performance of the model's accuracy. They even observe that when the domain used for fine-tuning did not match the target domain, the model's performance is improved. The authors of the study propose that the reason for this performance improvement could be due to the fact that the BERT-base model was initially trained on corpora with a strong knowledge-based focus, such as Wikipedia, and that simply fine-tuning the BERT model on less knowledge-based corpora improves its performance (Rietzler et al., 2019). In the work Li et al. (2015), the authors explore the potential of domain adaptation to leverage unlabeled data in a target domain when working with limited labelled data. Their research focuses on utilizing microblogging data, specifically tweets from Twitter, to improve disaster response efforts' speed, quality, and efficiency. However, obtaining labelled data for a new disaster is often challenging, while unlabeled data accumulates rapidly (Li et al., 2015). To address this issue, the authors propose employing labelled data from a previous disaster (source disaster) and unlabeled data from the current disaster (target disaster) to train domain adaptation classifiers for the target disaster. The results obtained by Li et al. demonstrate the effectiveness of domain adaptation with unlabeled data from the target disaster, combined with labelled data from previous disasters, in enhancing the classification performance for the target disaster.

In summary, the challenges we found in related work of detecting private sensitive content encompass the absence of labelled data sets, variations in the definition of sensitive content, and the necessity of considering contextual information. Additionally, the unique domain of Norwegian social media, characterized by informal language, poses additional complexities.

4. Defining Private-Sensitive Content

The variation in definitions used for detecting private sensitive content, as highlighted in Chapter 3, can lead to differences in the detected content, as they may not be grounded in the same principles. Given the subjective aspect of privacy, it is crucial to establish a shared understanding of what constitutes private-sensitive content with clear guidelines for the purpose of annotating the data set in our research. In this thesis, our objective is to develop a comprehensive definition of what constitutes private-sensitive content, drawing upon the principles and guidelines outlined in GDPR. While we acknowledge that other privacy regulations exist in different regions, we focus on GDPR as it serves as a prominent and authoritative framework within the EU for governing individuals' privacy. Consequently, we believe that aligning the definition of privacy detection with the GDPR can contribute to a more standardized approach.

The process of annotation is conducted to label entries in our data set to provide the ground truth for training the classifiers. The final annotation process is presented in Chapter 5 which results in our labelled data set. This process involves the utilization of annotators who manually label the data set according to our established guidelines. In this chapter, we introduce the concept of test annotation, which involves annotating data using different annotation guidelines in an iterative manner to progressively improve the definition and annotation guidelines. The data set for labelling will be introduced in Chapter 5 and the data set used for test annotation is a subset of the data set used in the final annotation process. This method is employed to prepare for the final annotation process, aiming to facilitate the development of a high-quality labelled data set.

This chapter, presents the process of establishing our annotation guidelines, specifically focusing on defining private-sensitive content. Following that, the established guidelines are presented, including a precise definition of private sensitive content based on GDPR's definition of personal and sensitive data, as well as previous research discussed in Chapter 3. This chapter is critical in addressing Research Question 1 as we describe the decisions made to define private sensitive content. These decisions reflect how we chose to optimize the labels for the classification task of detecting private-sensitive content.

4.1. Optimizing the Categories for Labelling

This section outlines the process of establishing clear guidelines for annotation. We begin by adopting the annotation guidelines presented in Bioglio and Pensa (2022). To assess the suitability of these categories for our task, we conduct a test annotation process. We annotate 2000 entries from our data set presented in Chapter 5 with the participation of the two authors of this thesis and one external annotator. Based on the observations and feedback received during this process, we make modifications and improvements. The resulting definition and guidelines are presented in the following section.

We draw inspiration from similar research when developing the categories for what constitutes private-sensitive information. The guidelines presented in Bioglio and Pensa (2022) is used as a baseline for annotating text as sensitive, non-sensitive, unknown, and unintelligible. In Bioglio and Pensa (2022), the authors consider a post to be sensitive if the text is written in clear English and if the annotator is certain that the information provided violates an individual's privacy. In order to be labelled as sensitive, the information would reveal at least one of the following :

- Events in the private sphere
- Health or mental status
- One's habit
- Sentimental status
- Considerations that may hint at the political orientation or religious belief of a mentioned person
- Current or upcoming moves
- Information that can help geolocate the author or other person mentioned

This list of subcategories is not exhaustive, as the authors acknowledge that text can be labelled as sensitive if the annotator feels discomfort specifically due to the private content it contains. They further clarify that this discomfort should not arise from moral considerations (Bioglio and Pensa, 2022).

We now introduce the modifications made to the guidelines originally presented by Bioglio and Pensa before conducting the test annotation. To ensure consistency and objectivity in the annotation process, we narrow down the categories to be exhaustive, excluding instances where annotators feel discomfort outside the predefined categories. Additionally, we adapt the guidelines to the context of Norwegian social media by using the Norwegian language and including slang and abbreviations. We replace "geolocate" with "locate" to broaden the scope of determining someone's whereabouts. Furthermore, we include additional categories aligned with the data considered important for protection under the GDPR, such as an individual's financial situation, personal identifying information (PII), and non-public information related to criminal activity. In our context, PII includes details such as name, address, phone number, social security number, email address, driver's license number, passport number, and other personal identification numbers or codes. The categories used for the test annotation process are: Sensitive, Non-sensitive, Unknown, and Unintelligible. In order to be labelled as sensitive, the information would reveal at least one of the following:

- PII
- Events in the private sphere
- Health or mental status
- One's habit
- Sentimental status
- Considerations that may hint at the political orientation or religious belief of a mentioned person
- Current or upcoming moves
- Information that can help locate the author or other person mentioned
- Reveal information about one's financial situation.
- Reveal information about criminal activity that is not public information

During the test annotation process, we annotate 2000 entries using the modified definition of Bioglio and Pensa presented above. To prevent hidden assumptions in our annotation guidelines, we engage an external annotator to provide feedback. By including an external annotator, who is not directly involved in the development of the guidelines, we aim to introduce fresh perspectives and minimize potential biases that can arise from our own subjective preferences or prior knowledge. As a result, we identify certain subcategories within the private-sensitive category that are overly broad and frequently overlapped. To address this, we make changes in the list presented above in an effort to arrive at more essential and specific subcategories. We now go over these changes and the justifications for each change.

The subcategory "current or upcoming moves" is removed as it duplicates the location-related aspect covered by another category. When annotating, we observed that the subcategory often overlapped with the category “can help localize the author of the post or other people mentioned”. "Current or upcoming moves" would in essence include text that revealed the current or future location of an individual. As a result, we made the decision to remove this subcategory to create more fundamental subcategories.

The decision to remove “information about one’s habits” was not as straightforward, but we ultimately came to the conclusion that the most private-sensitive factors are not the habits themselves but rather the aspects they reveal, such as location and other fundamental elements. The variation in interpretation during the test annotation process, which resulted in different labelling, is a significant factor behind this decision. The annotation was approached in diverse ways, leading to potential discrepancies in how the subcategory was understood and applied. We perceive this as a potential risk to the overall quality of the annotated data set, as the subjective opinions, lacking a consistent foundation for evaluating sensitivity, could introduce an inconsistency in the labelling. By emphasizing more fundamental private-sensitive elements in our definition, we aim to reduce room for subjective interpretation and facilitate a more consistent labelling process for annotators.

The subcategory "information on an event in the private sphere" is replaced with a new subcategory called "intimate details on family or romantic relationships." We observed that the term "private sphere" was interpreted differently among us, and this subcategory often revealed an individual’s location or other fundamental elements covered by other subcategories. By introducing a subcategory that is clearer and more narrowly focused, we may reduce overlap with the location subcategory, and clear up an ambiguity regarding how broadly the term "private psher" is defined.

The subcategory "information on sentimental status" is excluded because of its potential for subjective interpretations and overlap with the ’health/mental status’ category. ’Sentimental status’ is a broad expression, and during the annotation, we experienced that there were different interpretations of what would classify as information on sentimental status. We observed that in the most severe cases, this subcategory overlapped with the "health/mental status" category. As a result, the most intimate examples of sentimental status would already be covered by another subcategory. We chose to exclude the sentimental state from our definition because it allows for a wide range of subjective interpretations and the most basic examples were already covered by another subcategory.

The feedback we received from the external annotator provided us with valuable insights. Specifically, it emphasized the significance of clearly articulating the definitions for the four labels in our annotation guidelines. It highlighted the need to remove any hidden assumptions which is essential to help provide a common understanding of the problem so that annotators can apply the labels appropriately. The labelling conducted during the test annotation process of optimizing the categories is not considered in the final results of the annotation process. Instead, it serves as an iterative approach to establish appropriate guidelines for the annotation process. The resulting definition and guidelines for the annotation process are presented in the next section.

4.2. Annotation Guidelines

In this section, we present the guidelines for the annotation process, which present the criteria for labelling information as private-sensitive, non-sensitive, unknown and unintelligible. In our definition of private sensitive information, we have incorporated the definition of personal data as stated by GDPR, along with the definition of privacy-related information presented by Bioglio and Pensa in their research on detection. Our definition is structured as a categorical list presented in Table 4.1, and is designed to avoid any overlap between categories. This has been achieved by carefully delineating the categories at a fundamental level.

The informal nature of chat forums often leads to the frequent use of slang, abbreviations, and acronyms. Consequently, we try to incorporate these linguistic elements into our research. To this end, our annotators are instructed to conduct thorough research in order to understand the meanings of any words that are unclear or ambiguous.

With regard to the labelling of data in our research, we have established four categories: sensitive, non-sensitive, unknown, and unintelligible. The sensitive category is further divided into seven subcategories. The next subsections go in-depth into each of the four categories and the seven subcategories that together constitute our definition of the category private-sensitive.

4.2.1. Sensitive

In line with GDPR's definition of private sensitive data, we define address, phone number, account information, credit card number, employee ID, license plate information, customer number, and address as private-sensitive information. This means the text should be labelled as private-sensitive if the text contains any of the PII's listed above. It is important to clarify that the PII's doesn't necessarily need to belong to the writer. The reason this information is considered sensitive is that it can potentially be used to identify an individual if it is aggregated with other data.

Information that can locate a person at a certain time is another category which should be labelled as private-sensitive. This is the case if the text contains information about where a person is located at a specific time. There are different degrees of how subtle this information is. For instance, a straightforward example would be "Every Wednesday at 11 am I go to watch Nidarosdomen and enjoy the church bells ring." This information provides a specific location, Nidarosdomen in Trondheim, as well as a specific time, every Wednesday at 11 am. In contrast, a more subtle example is "Every morning I take the 20 bus from Carl Berner, and the roundabout is not efficient enough". From this text, we can gather that the individual is likely waiting at a bus stop at Carl Berner during a specific timeframe. The information provided in the latter example is less precise than the first one but still provides some degree of insight into the individual's location. It is worth noting that seemingly innocent details like this can be used to locate the whereabouts of an individual at a specific timeframe and potentially pose a threat to the individual's privacy.

Any information revealing an individual's health or mental state should be classified as private sensitive. This category includes information such as an individual's diagnosis, symptoms, treatment, medications, addiction, and therapy. Some examples of text that should be classified in this category are as follows: "My mother is constantly intoxicated on multiple occasions throughout the week. It irritates me that she can't control her consumption." and "I can't handle large crowds; they give me social anxiety." The first example should be labelled as private sensitive because it indicates that the mother may have an addiction problem, even though this is not explicitly stated. The author of the second example claims to have mental health issues when the individual is in large crowds and should also be labelled as sensitive. On the other hand, the text "All of these new updates are irritating me. I won't have time to learn it until the next update." should not be labelled as private sensitive. The writer claims to be irritated, but being irritated or having a feeling about a situation does not qualify as giving information about the writer's mental or health state.

Intimate details about someone's personal life should be labelled as private-sensitive. This category can be challenging because the definition of intimate details can vary depending on the individual or cultural context. We have worked to make the category as concrete as possible to minimize subjective differences between annotators. Examples of intimate details about someone's personal life include sexual orientation or gender identity, issues within a relationship such as cheating or divorce, details about an individual's romantic and sexual history or preferences, and more significant family decisions such as whether to have a child or not. We note that these examples are not exhaustive. This category should not include everyday descriptions of family or relationships, such as stating that you have children or that you read to your children every night. However, stating that you are female and have a boyfriend should be labelled as intimate details because it may reveal information about the individual's sexual and romantic orientation. As a general rule, it should be labelled as "intimate detail about someone's personal life" if it potentially could harm someone's reputation or pose a threat to the writer or others.

Information about one's financial status is defined as private-sensitive information. While it is acceptable to discuss the economy in general terms, revealing an individual's personal financial situation is considered sensitive. The distinction is important because it is only the disclosure of an individual's own financial situation that is considered to be private-sensitive. For instance, the text "The interest rate has risen two per cent since last year" is talking about the economy in general terms which should not be labelled as private-sensitive. However, if the text contains information about the writer's financial situation in relation to this increase it is considered as private-sensitive. An example of this would be "The interest rate has risen two per cent since last year, and if this continues, we might have to sell the house."

Information that may reveal one's political orientation or religious beliefs is also considered to be private-sensitive. This includes opinions regarding political topics, or if the text reveals information about the writer's political stances. An example of political stances includes socialism, liberalism, conservatism, environmentalism, and others. Examples of political topics may include civil rights, healthcare policy, immigration, climate change, education policy, social welfare, and more. An example of a text that would fall within this category is "Shall we just sit on our asses and do nothing while the rich get richer?". In this example, we get the information that the person writing probably has socialistic views and beliefs that wealth should be more fairly distributed. An example of a text that would not fall within this category is "I saw a news article about the new health minister in Norway. What are your opinions on them?". In this example, the user talks about politics but does not reveal their political stance or opinion on the matter.

Information related to criminal activity refers to any personal information or data that can be used to identify or incriminate an individual or group involved in illegal activities. This includes individuals' criminal records, history of arrests and charges, information about ongoing criminal investigations, an individual's involvement in organized crime and information about an individual's involvement in illegal activities like drugs. An example of a text that would fall within this category is "When I was 19 I drove while drunk and hit a fence. Luckily I know better now". Text discussing crimes described in a news article, as well as general discussions about crime or criminal activity that are openly accessible, are not considered private sensitive information.

In order for a text to be defined as private sensitive it needs to be understood in Norwegian and fall within at least one of the categories listed below.

- Personal identification information, such as full name, address, phone number, or email address.
- Information that can help locate the post's author or other people mentioned
- Information on health or mental status

- Intimate details on family or romantic relationships
- Information about one's economic status
- Information that may reveal one's political orientation or religious beliefs.
- Information about one's or others' criminal activity that is not already public information.

4.2.2. Unknown

It is important to note that when annotating text, the annotator should make a concerted effort to label the text as private-sensitive or non-sensitive, and not use the unknown label as a "maybe" label. The unknown label should be used if the annotator does not have enough context, information or domain knowledge to evaluate if it is private-sensitive.

The text may lack sufficient context for the annotator to determine whether it contains sensitive information. If the text being evaluated could fit into the other sensitivity categories presented above, such as opinions on political topics, but could also be non-sensitive in other contexts, it should be labelled as unknown. For instance, the phrase "Yes I agree", could be labelled as private-sensitive if it was part of a political debate, but could also be labelled non-sensitive if it was part of a conversation about cake preferences. Hence, the phrase should be labelled as unknown due to the lack of context necessary to evaluate the sensitivity of the post.

Incomplete text can also be a significant obstacle as annotators may not have access to additional information necessary to evaluate. This can occur if determining the sensitivity heavily depends on information that is linked to an external site. For instance, the sensitivity of the statement "I recommend this article, follow this link to read it:" is heavily based on the external article it links to. Access to the link is therefore necessary to determine whether it is private-sensitive or not. It is not clear if the article provides a political message or falls within other private-sensitive categories, hence the statement should be labelled as "unknown". However, the mere presence of a link is not problematic if the rest of the text provides enough information for the annotator to determine its sensitivity. For instance, the statement "I will be at Nidarosdomen tomorrow at 12, follow this link to see information about the event:" is private-sensitive due to the author's disclosure of their future geolocation, regardless of whether the annotators have access to the following link or not. It is important to underline that the annotators should not explore links to help determine the sensitivity of a text. This is because links can refer to external sites that might not be valid in the future. Links should solely be viewed as a "black box" for the purpose of referring to external sites.

A lack of domain knowledge can also be problematic when determining the sensitivity. Annotators can encounter texts that require a higher level of expertise than they possess. An example of this is when the text contains unfamiliar terminology and the annotator cannot determine the meaning even after conducting a simple research, such as a Google search. For instance, it is necessary for the annotator to know that fibromyalgia is a medical disorder in order to understand that the text "Last year I got fibromyalgia" reveals medical information. If the annotator searches for the meaning, but still does not understand that this is a medical disorder, the text should be labelled as unknown. It is important to emphasize that the annotator does not need to have a deep understanding of the domain in order to determine the sensitivity. An example of this is the text "Last year I was in the hospital because I got fibromyalgia" which should be labelled as private-sensitive because the author explains that fibromyalgia was the reason why the person was hospitalized. In the latter example, it is not necessary to have a thorough understanding of what the medical disorder implies in order to determine that the text reveals parts of the individual's medical record.

The following list summarizes the different criteria for labelling text as having unknown sensitivity:

- Lack of context to determine the sensitivity of the text.
- Information that is linked to an external site is necessary to determine the sensitivity of the text.
- The text contains unfamiliar terminology necessary to determine the sensitivity, and they cannot determine the meaning even after conducting simple research, such as a Google search.

4.2.3. Unintelligible

A text can be considered unintelligible when it is not understandable from a lexical point of view. This can happen due to poor language use, where the words and sentence structure do not make sense to the annotator. The text will also be unintelligible if a substantial part of the text is written in another language than Norwegian. While it is acceptable to include English slang words used together with Norwegian, whole sentences that cannot be categorized as slang should be labelled as unintelligible. The sentence "How are you" should be labelled as unintelligible, but "Håper det går bra på forestillingen! Break a leg!" will not be labelled as unintelligible.

The following list displays the criteria for labelling text as unintelligible:

- It is written with a syntax that prevents it from being comprehensible from a lexical point of view.
- The text only includes links to external sites.
- The text is written in another language than Norwegian. The exception is loanwords and slang from other languages.
- The text contains whole sentences written in another language than Norwegian that cannot be categorized as slang.

4.2.4. Table overview

Table 4.1 highlights the definitions for the categories "Non-Sensitive", "Unknown" and "Unintelligible", followed by the subcategories within the "private-sensitive" category.

Table 4.1.: Overview of our definitions of the different labelling categories

Category	**Definition**
Non-sensitive	Non-sensitive information refers to text that does not fall within any of the following entries in this table.
Unknown	The "unknown" category is used when the sensitivity of a text cannot be determined due to a lack of context, necessary external information, or unfamiliar terminology. It should not be used as a "maybe" label, but rather when the annotator is unable to confidently determine the sensitivity of the text.
Unintelligible	Unintelligible refers to text that cannot be understood due to poor language use or a substantial amount being written in a language other than Norwegian, unless it's slang or loanwords. It also includes text with a syntax that makes it incomprehensible and text that only includes links to external sites.
Personal identifying information (PII)	Information such as name, address, phone number, social security number, email address, driver's license number, passport number, and other personal identification numbers or codes.
Location information	Information that reveals the location of an individual at a specific time. This includes any details that can be used to pinpoint a person's whereabouts, such as the name of a specific location, time of day, or mode of transportation. This information can be used to track or monitor an individual's movements.
Health and Mental State Information	Information that refers to any information related to an individual's physical and mental health. This includes details about an individual's medical history, diagnosis, treatment, medication, therapy, and any other information related to their physical or mental well-being.
Intimate details on family or romantic relationships	Intimate details refer to any information that reveals sexual orientation, gender identity, romantic history, or family planning, and has the potential to harm someone's reputation or pose a threat to the writer or others if disclosed.
Financial Status Information	Financial status information refers to any information related to an individual's personal financial situation, such as income, expenses, debts, investments, and credit history. It is considered private-sensitive because it may be used to discriminate against or exploit individuals based on their financial situation.
Political and Religious Stances	Political and religious stances refer to any information that may reveal an individual's political orientation or membership in a religious group, including opinions on political topics and specific political stances. This information is considered private-sensitive because it can lead to discrimination or prejudice against an individual based on their stance.
Criminal activity	Personal information or data that can identify or incriminate individuals or groups involved in illegal activities, including criminal records, arrests, charges, and involvement in organized crime or illegal activities like drugs.

5. Data Set Creation

This Chapter presents how we create the labelled data set for the classification experiments in Chapter 6. To address Research Question 1, we will describe our approach for improving the labels used for classification and the amount of data associated with each label. Avoiding hidden assumptions, the precision of the annotators and appropriate data sources are all essential factors to consider during this process.

A significant part of this research has been dedicated to data collection and annotation processes. In this chapter, we will explain our decision to create a new data set, how we collected and preprocessed the data, the annotation process, and the data preprocessing after the annotation, as well as a data analysis of the resulting labelled data set. Finally, we outline the additional processing steps performed on the data set to prepare it for utilization in the experiments conducted in Chapter 6.

5.1. Existing Data Sets

Our search did not discover any data sets used for detecting private-sensitive content consisting of Norwegian entries from social media. The search was extended to other fields, and we discovered that the master thesis "Detecting and Grading Hateful Messages in the Norwegian Language" by Svanes and Gunstad presents a data set collected from several social media platforms in Norwegian. The authors generously provided us with the data set, which had not been preprocessed other than anonymizing names and tagged users. The data set included Norwegian tweets, discussions from the website of the newspaper Resett, and Facebook posts from public Facebook pages such as Norwegian newspapers and public figures. The authors explicitly expressed their intention to collect data from perceived controversial and heavily debated posts. Notably, the most heavily debated posts from Facebook, Twitter and Resett focused on immigration, the environment, and politics (Svanes and Gunstad, 2020). After examining the data set, we discovered a significant imbalance because a great majority contained political opinions that we would label as private-sensitive. This was expected because the most debated posts from their data sources were political topics.

Due to the imbalance caused by the high percentage of political discussions, we agreed that the data set created to detect hateful content was not optimally suited for private-sensitive detection. Furthermore, we wanted to contribute to the research field with a labelled Norwegian data set from social media. Consequently, instead of using more resources to search for existing Norwegian data sets, we agreed to begin collecting our own data set going forward.

5.2. Data Sources from Reddit

When deciding which social media platform to extract data from, we considered Facebook, Twitter and Reddit. We explored private and official groups on Facebook. We found some promising candidates but decided to disregard the platform due to uncertainty regarding legal regulation and ethical data extraction issues. When exploring the possibilities on Twitter, we discovered that posts often referred to a picture, which is beyond our scope. Furthermore, we struggled to find relevant users writing in Norwegian apart from public newsletters or organizations. On the other hand, Reddit contained discussions in Norwegian about diverse topics.

Ultimately, we chose to extract data from Reddit. Reddit is a social media platform where users can share and discuss content. The platform is divided into communities known as subreddits, each of which focuses on a specific topic. Users on this platform commonly create usernames unrelated to their identities.

One of the primary reasons we chose Reddit as our data source was the presence of publicly available data and the use of informal language by its users. Moreover, the platform offered an API for data extraction. To extract Norwegian content, we investigated two subreddits, r/Norge and r/Oslo. With 211,000 and 48,000 members, respectively, these subreddits were selected based on the strong presence of discussions in non-standard Norwegian.

5.3. Scraping Using the Reddit API

We used the Reddit API and the PRAW Python library to collect data from the subreddits r/Norge and r/Oslo. We scraped the most recent posts and related comments from each subreddit using the subreddit/new endpoint with a limit of 1000 posts. We chose this endpoint to analyze the most recent community discussions. We used the PRAW library to access the Reddit API because it provided a convenient wrapper around the API, allowing us to send requests and receive responses in Python easily. We collected data in a Jupyter Notebook environment, which allowed us to store and manipulate the data efficiently.

During our data collection process, we encountered a limitation enforced by the Reddit API that restricted the number of requests we could make within a specific time frame. We implemented a restriction of 1000 requests and made multiple requests to handle this limitation.

After the scraping, we obtained a data set of 21,069 entries, including posts and comments. Among these entries, 13,427 rows were from the r/norge subreddit, while 7,642 rows were from the r/oslo subreddit. Each entry in the data set contains the selftext, representing the content of the post or comment, along with the associated title, subreddit, and postid. Each post possesses a unique postid, whereas comments have the postid of the post they are responding to. Within the data set, the post marked as the first in a sequence of posts with identical postids signifies the initiation of a new post thread.

5.4. Data Cleaning for Preparing the Data Set for Annotation

The raw data was cleaned to remove inconsistencies, errors, and missing values that could jeopardize the analysis's integrity. To begin cleaning the data, we removed any rows or entries that contained media files or were marked as deleted, as these entries were irrelevant to our analysis. After the cleaning process, the data set consisted of 20852 rows.

To make it easier to navigate which rows the annotators would label, we created a new data set that only consisted of 4000 rows from the merged and cleaned data set. We combined the first 2000 entries from the r/norge and r/oslo subreddits to create the final data set. Hence, the data set each annotator received consisted of 4000 rows and they were given instruction for a subset of rows they were assigned to label ranging from 1000 rows to two 2000 rows. An example of a complete row from the data set provided to the annotators is displayed in Appendix A. Table 5.1 displays the amount of entries obtained from the scraping, after merging and cleaning the two data sets and the final data set.

Table 5.1.: Overview of the amount of data obtained from scraping the subreddits r/norge and r/oslo, the merged and cleaned data set consisting of the entries from the two subreddit and the final data set used for labelling.

Data sets	
Subreddit	**Number of entries**
r/norge	13427
r/oslo	7642
Merged and cleaned data set	20852
Annotation data set	4000

5.5. Annotation Process

A total of eight volunteer annotators participated in the annotation of the Annotation data set. Each annotator was given an Excel spreadsheet for the annotation, a tool that all annotators were familiar with and could use. The annotators were introduced to the task through the communication platform that suited them best. Depending on the preferences of each annotator, a combination of digital video meetings, phone calls, or in-person meetings was used as the communication medium. Given the high number of categories involved, it was vital to explain the nuanced distinctions between them. We illustrated each annotation category with examples where especially edge cases were presented. Additionally, all annotators were given a written document containing comprehensive definitions accompanied by additional examples.

To improve the reliability of the labelling, we have adopted an annotation approach inspired by Bioglio and Pensa (2022), in which multiple annotators annotate each entry. In our annotation process, we ensure that each entry is annotated by at least two annotators, with a target of involving three annotators whenever possible.

5.6. Improving the Amount and Quality of Labelled Data

After reviewing the result during the annotation process, we observed a shortage of private-sensitive labelled entries. Hence, we created an extra data set consisting of another 2000 rows from the merged and cleaned data set. All entries in the extra data set were part of the subreddit r/norge. To obtain a more balanced data set, we needed more private-sensitive data. Hence, we targeted the post threads with titles that could suggest private-sensitive discussions. An example is a thread titled "gjeldstrøbbel", where we expect sensitive information about finances to be disclosed. This resulted in 844 of the 2000 entries in the extra data set being labelled.

Table 5.2 displays the subreddit source, total entries and the number of annotators labelling each data set. The data set created to supplement the shortage of private-sensitive content is referred to as Dataset 5. The remaining data sets include the entries from the original Annotation data set of 4000 rows. Hence, the total number of entries labelled was 4842 rows. Ideally, each data set would be labelled by three different annotators, but due to limited resources, some of the data sets were labelled by two annotators.

Table 5.2.: Overview of the labelled data sets used in our research, including the number of annotators for each data set.

Labeled Data sets			
Data set	**Subreddit**	**Total Entries**	**Annotators**
Dataset 1	norge	1000	3
Dataset 2	norge	999	2
Dataset 3	oslo	1000	3
Dataset 4	oslo	999	2
Dataset 5 (extra data set)	norge	844	2

5.7. Agreement Among Annotators

The level of agreement among annotators can provide insights into various aspects, such as the annotation guidelines' effectiveness, the annotators' precision, or the annotation task's difficulty. We will first display details from the annotation results for data sets annotated by three annotators, followed by the same approach for data sets annotated by two. Following, we present statistical measures of annotator agreement for each of the five annotated data sets.

Table 5.3 presents the annotation results from the data sets involving three annotators. The three middle columns illustrate the instances where each class label reached precisely one, two, or three annotations. Additionally, the last column illustrates the sum of these columns, which implies the number of entries that received at least one annotation. For the private-sensitive class, the majority of its annotations were assigned when only one of the annotators labelled it as private-sensitive. This count decreases from 228 to 145 when exactly two annotators agree on the label, and further decreases to 88 when all three annotators agree. A similar trend is observed for the "Unknown" label. In contrast, the non-sensitive and Unintelligible classes have a majority when all annotators agree.

Table 5.3.: The table provides details of the annotation for Dataset 1 and Dataset 3, both of which involved three annotators

Dataset 1 and 3				
Class	**1 annotation**	**2 annotations**	**3 annotations**	**Sum**
Private-sensitive	228	145	88	461
Non-sensitive	320	416	514	1250
Unknown	361	106	41	508
Unintelligible	63	20	575	658
Total	972	687	1218	2877

Table 5.4 presents similar results for the data sets involving two annotators. Interestingly, all classes demonstrate a higher level of agreement between both annotators, indicated

Table 5.4.: The table provides details of the annotation for Dataset 2, Dataset 4, and Dataset 5, each of which involved two annotators.

Dataset 2, 4 and 5			
Class	**1 annotations**	**2 annotations**	**Sum**
Private-sensitive	96	760	856
Non-sensitive	195	778	973
Unknown	223	504	727
Unintelligible	16	535	551
Total	530	2577	3107

by the increasing number of classes receiving exactly two annotations.

To provide insight into the agreement among annotators, we have calculated statistics for annotation agreement using various metrics based on the number of annotators involved. Fleiss' kappa is used to assess agreement between data sets labelled by three annotators, whereas Cohen's kappa is used to assess agreement between data sets labelled by two annotators. Additionally, we have computed the percentage of the annotation results where at least two or all three annotators agree.

Table 5.5 presents the calculated statistics of agreement between annotators. The annotation results for Dataset 1 achieve a Fleiss' kappa of 0.2454, which is considered to be a fair agreement. In contrast, Dataset 3 obtains a Fleiss' kappa of 0.7648, which is considered to be substantial agreement. However, it is worth noting that 576 entries in Dataset 3 were labelled as "Unintelligible" by at least two annotators, as shown in Table 5.6. The data set contained several threads including the English language, so this high number of unintelligible annotations is unsurprising. In Dataset 1, which exhibits a lower level of agreement, only 19 entries are labelled unintelligible, as shown in Table 5.6. This may suggest that the task of deciding whether an entry is private-sensitive, non-sensitive or of unknown sensitivity is a challenging task.

The data sets labelled by two annotators obtain similar values of Cohen's kappa and the percentage of agreement between the two annotators. Table 5.5 shows that the data sets obtain a Cohen's kappa between 0.8345 and 0.8659, which is considered an almost perfect agreement. The high level of agreement may be related to the annotators' involvement in annotating multiple data sets. This experience may have contributed to their improved annotation consistency and agreement.

Table 5.5.: Overview of agreement between annotators. Fleiss' kappa for three annotators, Cohen's kappa for two annotators. Additionally, the table presents the percentage of agreement.

Agreement between annotators				
Group	**Cohen's kappa**	**Fleiss' kappa**	**At least 2 agree**	**All 3 agree**
Dataset 1	-	0.245416	92%	41.1%
Dataset 2	0.834513	-	88%	-
Dataset 3	-	0.764783	98.5%	80.7%
Dataset 4	0.865853	-	93.2%	-
Dataset 5	0.846013	-	89.8%	-

5.8. Data Cleaning of the Annotated Data Sets

It was vital to correct any potential mistakes that may have occurred during labelling because the annotators labelled the data set using a spreadsheet, making it possible for them to accidentally modify cells. A common mistake was classifying the text as one of the private-sensitive subcategories while also classifying the text as unknown, unintelligible or non-sensitive. Hence, it was crucial to ensure that each entry marked as private-sensitive was also marked as belonging to at least one of the subcategories and vice versa. Additionally, we verified that each labelled data set only contained entries assigned to one of the primary categories and that all entries were labelled. For the conflicting labelled entries, we contacted the given annotator to clarify what they intended to label and updated them accordingly.

When preparing the data for the experiments we compared the labeling across the annotators for each data set. To assure that the annotators indeed reviewed the same text we made a check that compared the selftext and title columns. We did not observe differences between the texts except in some cells where an additional number was added at the end of the text. This indicated mistyping of labelling and the number where removed without contacting the designated annotator. Following this verification, we created a new data set and added the row id, selftext, title, and main categories from the rows in which at least two annotators had matching main category labelling. We combined all the data sets into a single data set called the merged data set in order to train our model. The distribution of labelling across the labelled data sets and the combined final data set is shown in Table 5.6.

To make the analysis more manageable, we merged the title of a post, noted as "title", with the content of a post/comment, noted as "selftext". If a post contains text in both columns, the title would be placed at the beginning of the selftext column, followed by the text originally in the selftext column. Some entries contained a title but had no text in the selftext column and vice versa. To address this issue, we substituted the Nan values with empty strings. Comments that did not have a title would therefore begin with an empty string followed by the content from the selftext column. A post that only consisted of a title, but no further content would have the title in the selftext column followed by an empty string. This allowed us to maintain the data's integrity while also ensuring that there were no gaps in the text data.

Table 5.6.: The table shows the distribution of the main categories where at least two annotators agree. The merged data set consists of all entries where at least two annotators agree from the data set listed above in the table.

Number of entries where at least two annotators agree					
Data set	**Non-sensitive**	**Private-sensitive**	**Unknown**	**Unintelligible**	**Total**
Dataset 1	639	197	65	19	920
Dataset 2	323	367	174	24	888
Dataset 3	291	36	82	576	985
Dataset 4	272	43	133	483	931
Extra data set	183	350	197	28	758
Merged data set	1708	993	651	1130	4482

5.9. Data Analysis of the Distribution of Sub-Categories

The distribution of the subcategories from the annotated data set is depicted in the bar diagram in Figure 5.1. The amount of occurrences of each subcategory is noted above the bar. The distribution is skewed, with a majority of 689 entries containing information that may reveal one's political orientation or religious beliefs. Other subcategories are substantially underrepresented. Other subcategories are significantly underrepresented, most notably the subcategory of Personal Identifiable Information (PII), with only two instances. Out of the 993 entries annotated as private-sensitive, 1017 subcategories received agreement from at least two annotators. This indicates that several entries have multiple subcategories linked to them.

5.10. Data Sets Used for Experiments

This section presents the data sets used for the experiments in Chapter 6. First, we will present the preprocessing and an analysis of the labelled data set, followed by the presentation and preprocessing of the unlabeled data set.

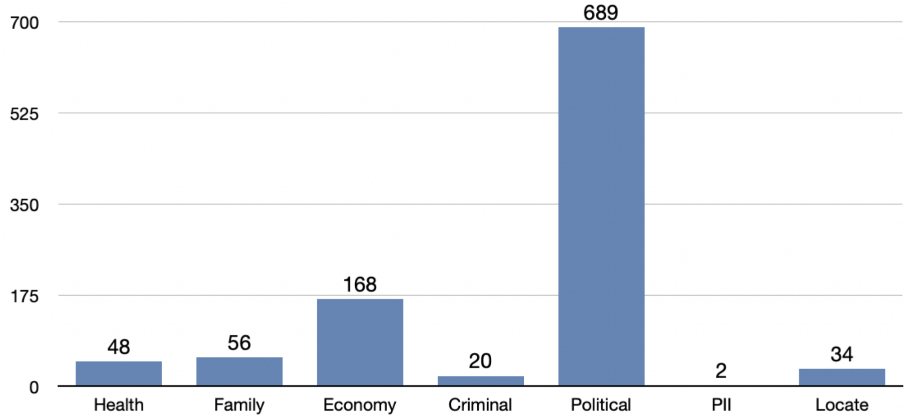


Figure 5.1.: Distribution of the subcategories within the entries labelled as private-sensitive.

5.10.1. Labelled Data Set

The labelled data set used in our experiments is a subset of the Merged data set, which was previously introduced in Table 5.6

Data Preprocessing

For the experiment, various preprocessing steps were performed on the data set. Firstly, all columns except "content," "non-sensitive," "unintelligible," "unknown," and "sensitive" were removed. The "unknown" and "unintelligible" columns were merged into a single column called "other." Furthermore, any NaN values were replaced with empty strings. Certain characters, such as ’]’, ’[’, ’(’, and ’)’, were removed, and all URLs were replaced with the placeholder "@LINK". Finally, all labels were combined into a single column called "Label," which can contain one of the three labels for each entry.

The resulting processed data set consists of 4,442 entries, with 981 labelled as "sensitive," 1,693 labelled as non-sensitive, and 1,768 labelled as unknown as presented in 5.7.

Data Analysis

From Figure 5.7, we observe that the data set is moderately imbalanced, with a more significant number of entries labelled as "Non-Sensitive" and "Other" compared to "Private-sensitive" entries.

Table 5.7.: Labeled data set used for experiments

Category	Count
Entries	4,442
Labelled as "Private-sensitive"	981
Labelled as "Non-Sensitive"	1,693
Labelled as "Other"	1,768

We wanted to analyze the text length distribution to better understand any lexicographic patterns. The box plot in 5.2 shows the distribution of the number of characters per entry in the data set. To further understand the relationship between text length and the different classes, we present a bar plot in 5.3 showing the average text length for each class.

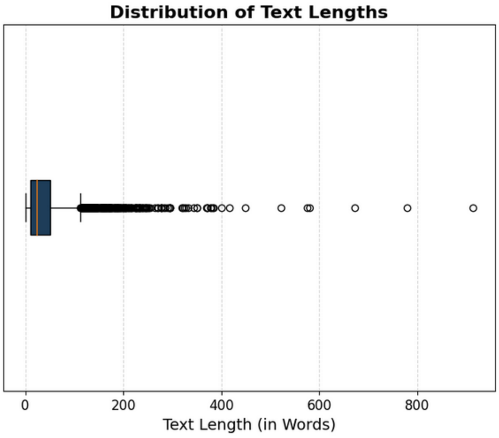


Figure 5.2.: Text length distribution

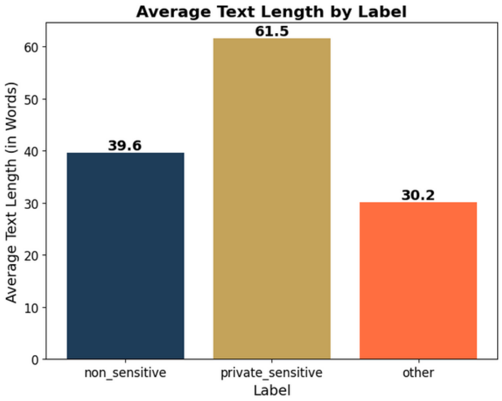


Figure 5.3.: Average text length in each class

In Figure 5.2, it is evident that the data set primarily comprises shorter texts, with a significant majority falling into this category. However, there are also outliers, represented by a few instances that contain over 800 words. Meanwhile, Figure 5.3 illustrates the average text lengths for different classes. The private-sensitive class exhibits the longest average text length, approximately 62 words, followed by the non-sensitive class at 40 words, and the other class with an average of 30 words.

5.10.2. Unlabeled Data Set

The unlabeled data set consists of 12,000 entrees that were scraped from the subreddits but not used for annotation. .

Data prepossessing

All columns except for "content" were removed from the data set. Additionally, any NaN values were replaced with empty strings. Certain characters were removed from the text, including ']', '[', '(', and ')'. Moreover, all URLs were replaced with the placeholder "@LINK". As a result of these preprocessing steps, the unlabeled data set now comprises 12,000 entries, with only one column, namely "content," containing the textual data.

6. Methods and Experiments

This chapter begins by providing a more detailed description of the methods employed in the experiments. Subsequently, it outlines the experimental plan and setup.

6.1. Methods

This section outlines the methods used to address detecting private-sensitive content in Norwegian social media for the experiments in this thesis. First, domain adaptation and task-specific training to enhance the performance of classifiers are presented. The chapter also describes the training and evaluation process, ensuring a comprehensive assessment of the models' performance.

6.1.1. Domain Adaptation and Task-Specific Training: MLM Fine-tuning

To improve the performance of the NB-BERT Base model to account for the distinctive nuances and characteristics of Norwegian social media language, and make the most of the available labelled and unlabeled data, we employ a domain adaptation technique. This is particularly crucial due to the limited size of our labelled data set and because our target domain is Norwegian social media language. Domain adaptation as an advantageous approach to address these challenges is emphasized in Chapter 3.

Our method for domain adapting the NB-BERT Base model is presented in Figure 6.1. We utilize the NB BERT-base model which is shown as the starting point in Figure 6.1. We then utilize 12,000 entries from our unlabeled data set to fine-tune the model using Masked language Model (MLM). Note that the data used for this process does not overlap with the data in the labelled data set. The outcome of the domain adaptation process is a domain-adapted language model that integrates the insights gained from MLM fine-tuning with the unlabeled data. The last step in this process is to make the model task specific to classify private-sensitive content as shown in 6.1.

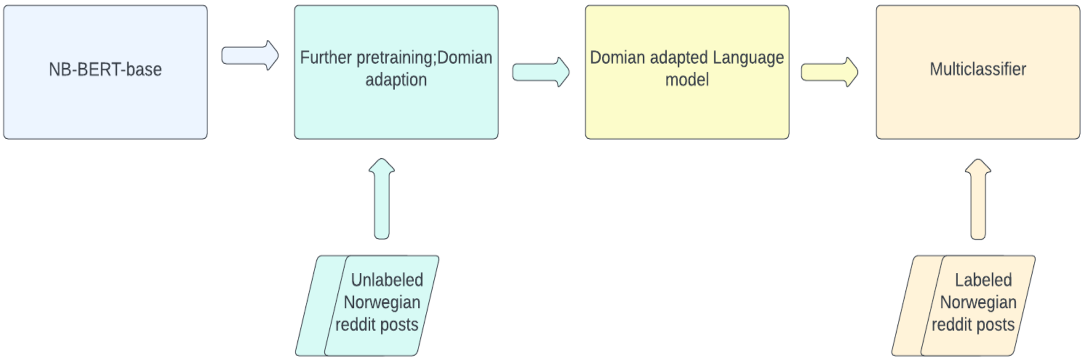

NB-BERT-base
Further pretraining;Domian adaption
Domian adapted Language model
Multiclassifier
Unlabeled Norwegian reddit posts
Labeled Norwegian reddit posts

Figure 6.1.: The process of domain adapting the model

Through the utilization of domain adaptation, our proposed method seeks to enhance the performance of the NB-BERT Base model in accurately classifying private-sensitive content within the domain of Norwegian social media.

6.1.2. Training and Evaluation Process

In this thesis, we follow a systematic process that encompasses balanced training, thorough evaluation through k-fold cross-validation, parameter optimization and evaluation on an unbalanced test set.

The training and evaluation process is presented in Figure 6.2. The process starts with using the labelled data set prepared for the experiments, presented in section 5.10.1. To address the class imbalance, a train/test split is performed resulting in two subsets: "Training data: Balanced" and "Test data: Unbalanced". A more detailed description of the train/test split is provided in Section 6.2. The balanced training data is further utilized in the k-fold cross-validation process. Figure 6.2 illustrates the splitting of the balanced training data into five folds (Fold 1 to Fold 5). Each of these folds serves as the test set once in its respective iteration (K1 to K5), allowing a thorough evaluation during training to mitigate the impact of data variability. The model's performance is assessed in each iteration, and the results are consolidated by calculating the average performance across all folds. The cross-validation process is repeated using grid search with various parameter combinations to identify the most optimal parameter combination, for the Bert model this includes utilizing early stopping as well.

The parameter combination resulting in the best performing model is utilized to train the final model using the entire balanced data set, referred to as 'Training data: Balanced' in figure 6.2. After building the final model, its performance is evaluated on the unbalanced

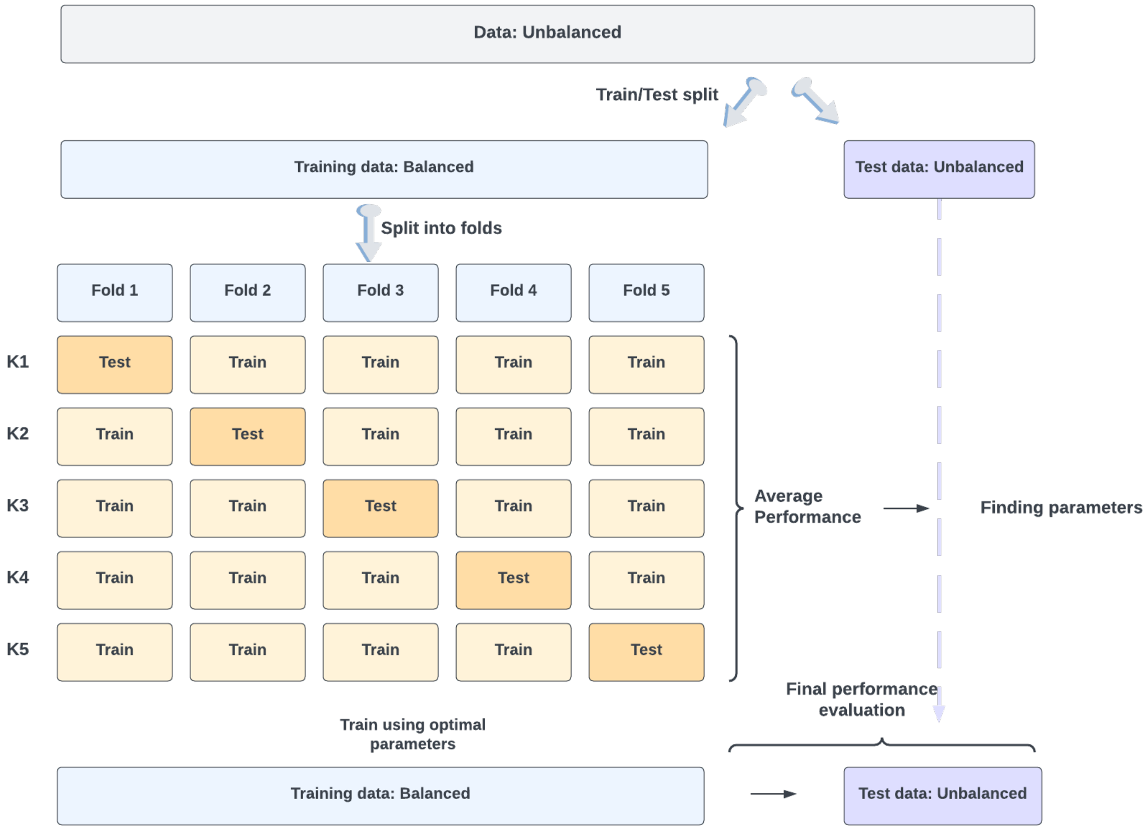

Data: Unbalanced
Train/Test split
Training data: Balanced
Test data: Unbalanced
Split into folds
Fold 1
Fold 2
Fold 3
Fold 4
Fold 5
K1 Test Train Train Train Train
K2 Train Test Train Train Train
K3 Train Train Test Train Train
K4 Train Train Train Test Train
K5 Train Train Train Train Test
Average Performance
Finding parameters
Final performance evaluation
Train using optimal parameters
Training data: Balanced
Test data: Unbalanced

Figure 6.2.: An Illustration of the Training and Evaluation Process: Visualizing the sequential steps encompassing data splitting, K-fold cross-validation, performance aggregation, parameter optimization, and final evaluation.

test data that has not been used during model selection and parameter tuning. This approach helps assess the model's ability to generalize beyond the training data.

Overall, this process allows us to use a substantial amount of the data for training the final model, while maintaining a thorough evaluation process through K-fold cross-validation, enhancing our confidence in the model's performance and ability to generalize to unseen data.

6.2. Experiments

This and the following sections present the experimental plan and setup designed to address Research Question 2: "How do different classifiers perform on detecting private-sensitive content in Norwegian social media?". The experiments are divided into two parts: (1) Detecting private-sensitive content using conventional classifiers, and (2) Detecting private-sensitive content using a transformer-based classifier. The results and discussion is presented in Chapter 7. We now present the process of splitting the labelled data set into a training set and a test set. The resulting training set and test set will be used in both experiments. Subsequently, we will present the experiments individually.

After completing both experiments, the results obtained from this transformer-based model will be compared to the results achieved by the best performing conventional classifier which will be presented in the subsequent experiment.

Splitting the Data Set into a Training Set and Test Set

The labelled data set is divided into a training set and a test set. The training set is carefully balanced, consisting of 940 entries for each of the three categories. This balanced distribution ensures equal representation during the training process. In total, the training set contains 2820 entries.

On the other hand, the test set is intentionally designed to be unbalanced. It aims to reflect the distribution of categories in the real-world domain being represented. The test set comprises approximately a 90/10 split, with 49 entries for the private-sensitive category, 115 entries for the non-sensitive category, and 120 entries for the other category. In total, the test set consists of 284 entries.

Category	Count
Private-sensitive	940
Non-sensitive	940
Other	940
Total	**2820**

Figure 6.3.: Training Set

Category	Count
Private-sensitive	49
Non private-sensitive	115
Other	120
Total	**284**

Figure 6.4.: Test Set

Each experiment follows the same final evaluation plan to ensure consistency and allow the production of comparable results.

6.3. Experiment 1: Detecting Private-Sensitive Content using Transformer-Based Classifier

This first experiment is set to address Research Question 2.2: "How does the NB BERT-based model, which is a transformer-based model, perform in detecting private-sensitive content using the collected data set?". This section presents the experimental plan and setup. The results of the experiment will be presented and discussed in Chapter 7.

6.3.1. Experimental Plan

The initial step in the experimental plan is to fine-tune the NB BERT-based model, which serves as the initial stage of domain adaptation thoroughly described in Section 6.1.1. The unlabeled data set presented in Section 5.10, which consists of 12,000 entries, is utilized for fine-tuning. The purpose of this step is to enhance the model's comprehension of Norwegian social media language.

The next step in the plan is to make the model task-specific for multi-class classification. For this, we utilize a labelled data set consisting of 2820 entries, as described in Section 6.2. To explore various hyperparameter settings, different combinations are tested by conducting a grid search aligned with cross-validation, as described in 6.1.2. The chosen parameters are the learning rate and the number of epochs. Additionally, early stopping is applied to help determine the optimal number of epochs by considering the different learning rates that were tested. The most optimal learning rate and number of epochs are decided based on the results from the grid search and the evaluation during early stopping.

Once the most optimal parameter configurations are identified, the most promising classifiers are trained on the labelled training set. This step ensures that the models can learn from the full range of available data, enhancing their ability to generalize to unseen samples.

Finally, the best performing model is evaluated on an independent test set, described in 6.2, which is separate from the data used during training. This evaluation step will contribute to providing a more unbiased assessment of the model's performance on previously unseen data.

6.3.2. Experimental Setup

Tools

The libraries used for implementation in this experiment are:

- **Transformers** [1]: Transformers is a library created by Hugging Face [2]that provides API's and tools for downloading and training pre-trained models.
- **PyTorch** [3]: PyTorch is a tensor library for deep learning optimized for GPU and CPU computation. .
- **Pandas** [4]: Pandas is a library used for data manipulation and analysis. It provides data structures like DataFrames to store and manipulate tabular data.
- **NumPy**[5]: NumPy is a library for numerical computing in Python. It provides support for large, multi-dimensional arrays and mathematical functions to operate on these arrays.
- **Scikit-learn**[6]: Scikit-learn is a popular machine learning library in Python. It provides various algorithms and tools for tasks such as classification, regression, clustering, and model evaluation.
- **Matplotlib**[7]: Matplotlib is a plotting library in Python. It provides functions for creating various types of plots and visualizations.
- **Datasets** [8]: The datasets library, part of the Hugging Face [9]ecosystem, provides a unified API to access and preprocess data sets for NLP tasks.
- **Seaborn**[10]: Seaborn is a matplotlib-based Python data visualization library which offers a high-level interface for creating statistical graphics.

Hardware Setup

During the initial fine-tuning phase on the unlabeled data set, we utilized Google Colab[11]with a T4 GPU and 12GB of memory. For the remainder of the experiment, we leveraged the NTNU IDUN computing cluster[12]as our primary computational infrastructure. Within this cluster, we made use of the Tesla P100 GPU, which offers 16 gigabytes of memory. The Tesla P100 GPU is configured with a maximum of 8 threads and 24 available cores. The NTNU IDUN cluster enabled us to conduct large-scale experiments with optimal performance and computational capacity. By combining the resources of Google Colab and the NTNU IDUN computing cluster, we were able to perform our experiments effectively.

Initial Fine-tuning using MLM

For the initial fine-tuning phase on the unlabeled data set, we employ the "NbAiLab/nb-bert-base"[13]pre-trained model from Huggingface, along with its tokenizer. Tokenization and data set preparation involves the creation of distinct tokenized data sets. These data sets are divided into training, testing, and unsupervised data. The target is to have 6000 tokens for both the training and testing data sets, while the unsupervised data should have a total of 12000 tokens. Text examples are chunked into 65t continuous token sequences, with masked tokens to improve the model's masked language modeling

[1] Python Software Foundation. transformers 2.1.0
`https://pypi.org/project/transformers/2.1.0/`. (Accessed 30.06.2023).

[2] Hugging Face
`https://huggingface.co/`. (Accessed 30.06.2023).

[3] PyTorch Foundation. PyTorch
`https://pytorch.org/docs/stable/index.html`(Accessed 30.06.2023).

[4] NumFOCUS, Inc. pandas
`https://pandas.pydata.org`. (Accessed 30.06.2023).

[5] NumPy
`https://numpy.org`. (Accessed 30.06.2023).

[6] Scikit-learn
`https://scikit-learn.org/stable/`. (Accessed 30.06.2023).

[7] Matplotlib
`https://matplotlib.org`. (Accessed 30.06.2023).

[8] Datasets
`https://huggingface.co/docs/datasets/index`. (Accessed 30.06.2023).

[9] Hugging Face
`https://huggingface.co/`. (Accessed 30.06.2023).

[10] Seaborn
`https://seaborn.pydata.org`. (Accessed 30.06.2023).

[11] Google Colab
`https://colab.research.google.com/notebooks/welcome.ipynb`. (Accessed 30.06.2023).

[12] NTNU.idun High performance computing
`https://www.hpc.ntnu.no/idun/`. (Accessed 05.06.2023).

(MLM) capabilities. The model is trained using a chunk size of 27, a batch size of 17, a total of 3 epochs, a learning rate of 2×10^{-5}, and a weight decay of 0.01. Finally, the fine-tuned model is deployed as "Haldis/nb-bert-base-domainaddapt-reddit"[14]on the Hugging Face model hub.

Training the Model to be Task-Specific: Multi-Class Classification

During the training process on the labelled data set, we employ the fine-tuned model "Haldis/nb-bert-base-domainaddapt-reddit"[15]from Hugging Face, which was deployed during the initial fine-tuning stage. During the grid search, the batch size is set to 16, and we explore epochs ranging from 2 to 6 with learning rates of 1×10^{-5} and 1×10^{-6}.

To evaluate the performance of the models during grid search, the average accuracy across all folds is calculated. Additionally, for each epoch within the cross-validation process, the validation loss and training loss are computed. This information is utilized to generate a graph depicting the average validation loss and training loss across all folds in the cross-validation. This graph serves as a useful tool for implementing early stopping

Final Evaluation on the Test Set

Once the optimal parameter combination is determined based on the evaluation results, the final model is trained. The entire training set is used to train the final model. Subsequently, the trained model undergoes a final evaluation using the dedicated test data set. To analyze the model's predictions and evaluate its performance, a confusion matrix is used. The confusion matrix provides valuable insights into the model's ability to correctly classify instances and identify any patterns of misclassification. Additionally, various evaluation metrics, such as accuracy, precision, recall, and macro average F1-score are evaluated.

To analyze the model's predictions and evaluate its performance, several metrics are assessed, including accuracy, precision, recall, and macro average F1-score. These metrics offer an evaluation of the model's predictive capabilities. Furthermore, a confusion matrix is utilized to visually represent the model's performance. The confusion matrix provides valuable insights into the model's ability to correctly classify instances and helps identify any patterns of misclassification.

[13] NbAiLab. nb-bert-base
`https://huggingface.co/NbAiLab/nb-bert-base`. (Accessed 01.07.2023).

[14] Haldis. nb-bert-base-domainaddapt-reddit
`https://huggingface.co/Haldis/nb-bert-base-domainaddapt-reddit).` (Accessed 01.07.2023).

[15] Haldis. nb-bert-base-domainaddapt-reddit
`https://huggingface.co/Haldis/nb-bert-base-domainaddapt-reddit).` (Accessed 01.07.2023).

6.4. Experiment 2: Detecting Private-Sensitive Content using Conventional Classifiers

This section presents the experimental plan and set-up for addressing Research Question 2.1: "How do conventional classifiers perform in detecting private-sensitive content using the collected data set?". This section presents the experimental plan and setup. The results of the experiment will be presented and discussed in Chapter 7.

6.4.1. Experimental Plan

The first step in the experimental plan is to conduct hyperparameter tuning for the conventional classifiers: Multinomial LR, Multinomial Naive Bayes, Random Forest and Linear SVM. The hyperparameter tuning is performed using grid search and 5-fold cross-validation.

We utilize grid search to find the most suitable hyperparameters. During the grid search process, we evaluate overfitting by analyzing a graph that displays the validation accuracy and test accuracy. The hyperparameters for each classifier are determined by taking into account both the overfitting graphs and the results that achieve a relatively high accuracy.

Once the hyperparameters are determined for each classifier, the classifiers are trained on the entire training set using the selected hyperparameters configurations from the grid search.

Finally, the trained models are evaluated on the independent test set to assess their performance in detecting private-sensitive content.

6.4.2. Experimental Setup

Tools

This experiment, like Experiment 1, makes use of the Python libraries Pandas, NumPy, Scikit-learn, Matplotlib and Seaborn. Scikit-learn is in this experiment used for the Tfidf vectorizing and the implementation of the classifiers: Multinomial LR, Multinomial Naive Bayes, Random Forrest and Linear SVM.

Hardware

As this experiment does not require as much computational force, we did not use the NTNU IDUN cluster[16]. During the experiment, we utilize Google Colab [17]with a Xeon CPU and 12GB of memory. The CPU is equipped with one physical core and two threads.

Hyperparameter Combinations

An overview of the specific hyperparameter combinations we explore for each classifier during grid search is shown in table 6.1. The table displays the parameters along with a list containing the range of values tested.

For the Multinomial LR, we aim to find the optimal level of regularization by experimenting with five values for the parameter denoted as "C". This parameter represents the inverse of the regularization strength and serves as a regularization parameter. Additionally, two additional parameters are considered. The first parameter is called "solver" which includes three different optimization algorithms. The second parameter, "multi_class", is set to multinomial. In the case of the Linear SVM classifier, we explore seven different values for the "C" parameter, while the penalty is set to "l2."

The Multinomial Naive Bayes classifier is tested with eight different values of the "alpha" parameter, which is an additive smoothing parameter for feature probabilities. In the case of the Random Forest classifier, we aim to find the optimal number of decision trees to be used and the optimal depth of these trees. The parameter "n_estimators" specifies the number of trees used, with three different values being tested. The parameter "max_depth" refers to the maximum depth of the decision tree and four different depths are tested.

Final Evaluation on the Test Set

Each of the models is evaluated on the test set in the same manner as in Experiment 1 in 6.3.

[16] NTNU.idun High performance computing
https://www.hpc.ntnu.no/idun/. (Accessed 05.06.2023).

[17] Google Colab
https://colab.research.google.com/notebooks/welcome.ipynb. (Accessed 30.06.2023).

Table 6.1.: Grid Search Parameters Tried for each Classifier

Classifier	**Grid Search Parameters**
Multinomial LR	C: [0.001, 0.05, 0.1, 1, 10] solver: ['lbfgs', 'sag', 'saga'] multi_class: 'multinomial'
Multinomial Naive Bayes	alpha: [0.0001, 0.001, 0.1, 1, 2, 5, 10, 100]
Random Forest	n_estimators: [50, 100, 200]] max_depth: [1, 2, 3, 4]
Linear SVM	C: [0.0001,0.0005,0.001, 0.02, 0.1, 0.5, 10] penalty: l2

7. Results and Discussion

This chapter presents the results of Experiment 1 along with a discussion of the results. We then use a similar approach to present the results of Experiment 2 and provide a corresponding discussion. Subsequently, we have a general discussion where we compare the results from the experiments with each other and related work. Additionally, we discuss the results in view of the definition of private-sensitive content and the labelling process. Any limitations experienced will also be discussed in this chapter.

7.1. Experiment 1: Detecting Private-Sensitive Content using an NB BERT-based Classifier

This section, first presents the results from the parameter tuning analysis, followed by the results of conducting early stopping by evaluating the validation loss and training loss. Finally, the final evaluation of the NB BERT-based model on the test set is presented. Each result will be followed by a discussion.

7.1.1. Results of Parameter Tuning Analysis

Result

We present the average accuracy of the models after performing 5-fold cross-validation. Table 7.1 displays the model's average accuracies when different learning rates or epochs are used. The variance is displayed together with the average accuracy. The highest average accuracy values for each of the two learning rates are emphasized in bold. Specifically, employing a learning rate of 1×10^{-5} resulted in the highest average accuracy of 0.8061 with a variance of 0.00046 after 5 epochs. Additionally, a learning rate of 1×10^{-6} attained an average accuracy of 0.8011 with a variance of 0.00013 after 6 epochs.

Table 7.1.: The average accuracies obtained from parameter combinations of different epochs and learning rate.

Learning Rate	Epochs	Batch Size	Average Accuracy
1×10^{-5}	2	16	0.8022 ± 0.00025
1×10^{-5}	3	16	0.7996 ± 0.00034
1×10^{-5}	4	16	0.8025 ± 0.00075
$\mathbf{1 \times 10^{-5}}$	**5**	**16**	$\mathbf{0.8061 \pm 0.00046}$
1×10^{-5}	6	16	0.7946 ± 0.00015
1×10^{-6}	2	16	0.7706 ± 0.00008
1×10^{-6}	3	16	0.7832 ± 0.00013
1×10^{-6}	4	16	0.7871 ± 0.00026
1×10^{-6}	5	16	0.7928 ± 0.00016
$\mathbf{1 \times 10^{-6}}$	**6**	**16**	$\mathbf{0.8011 \pm 0.00013}$

Discussion

The decision to explore relatively small learning rates was based on the understanding that smaller learning rates result in smaller updates to the model's parameters during training. This characteristic can be beneficial, particularly when working with smaller data sets where fine-tuning the model requires cautious parameter adjustments.

Comparing these results, the model with a learning rate of 1×10^{-5} achieved a higher accuracy than the model with a learning rate of 1×10^{-6}. The model with the lower learning rate demonstrated a smaller variance, indicating a more stable performance across different folds of the cross-validation process. A higher learning rate, such as 1×10^{-5} in this case, might allow the model to converge faster and achieve better overall accuracy. On the other hand, a lower learning rate like 1×10^{-6} leads to a more consistent performance with a lower variance but requires a longer training duration to reach optimal accuracy.

We can see that for a learning rate of 1×10^{-5}, a high accuracy of 0.8022 is already achieved after two epochs. While for learning rate 1×10^{-5} the highest accuracy is achieved after 6 epochs. It is important to note that average accuracy alone may not be sufficient to evaluate the performance accurately. It can be beneficial to consider other factors, such as stability and potential overfitting when choosing the most suitable parameter combination.

7.1.2. Validation Loss and Training Loss

Result

Figure 7.1a illustrates the average validation and training loss over 5 folds for 6 epochs, using a learning rate of 1×10^{-6}. The training loss consistently decreases throughout all 6 epochs. The validation loss also exhibits a downward trend, but the rate of decrease diminishes with each subsequent epoch. Specifically, from epoch 5 to 6, the decrease becomes marginal, resulting in a validation loss of approximately 0.5225 after 6 epochs.

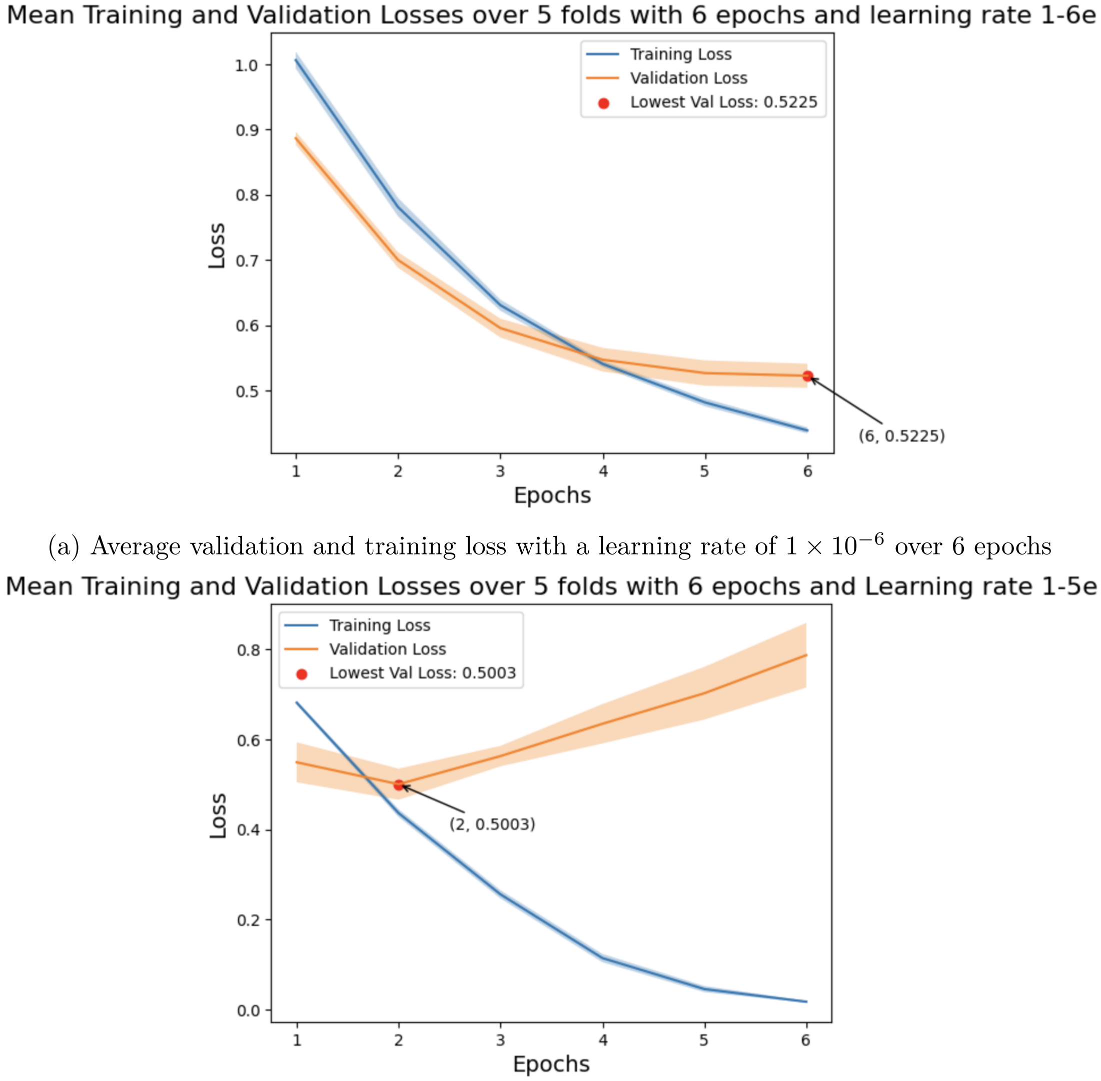
In Figure 7.1b, we observe the average training and validation loss over 5 folds for 6 epochs with a learning rate of 1×10^{-5}. The training loss demonstrates a consistent decline, whereas the validation loss decreases until the second epoch where it starts to increase. The red data point represents the lowest validation loss, which occurs at the second epoch and reaches a value of 0.5003.

Discussion

In Figure 7.1a, the learning rate of 1×10^{-6} demonstrates a consistent decrease in both the training and validation loss over the 6 epochs. This suggests that the model learns from the training data and generalizes to the validation data. It is important to note that the rate of decrease in the validation loss diminishes over time, suggesting a potential plateau in performance improvement.

Figure 7.1b, on the other hand, displays the results with a learning rate of 1×10^{-5}. The training loss exhibits a consistent decline, while the validation loss initially decreases until the second epoch. After epoch two the validation loss starts to increase. This suggests that the model updates were taking larger steps and converged more quickly than compared to using a learning rate of 1×10^{-6}. The rise in the validation loss also suggests a possibility of overfitting after two epochs, where the model excessively fits the training data and struggles to generalize to unseen examples in the validation data.

Despite achieving the highest accuracy, we have decided not to proceed with the parameter combination of a learning rate of 1×10^{-5} and 5 epochs. Instead, we will use a learning rate of 1×10^{-5} with only two epochs which is where the lowest validation loss and an accuracy of 0.8022 is obtained. This decision is intended to prevent potential overfitting of the training data. We encountered difficulty in confidently selecting only the aforementioned parameter combination due to the alternative option of using a learning rate of 1×10^{-6} with 6 epochs. This alternative achieved a nearly equal average accuracy of 0.8011 while demonstrating less than half the variance.



(a) Average validation and training loss with a learning rate of 1×10^{-6} over 6 epochs

(b) Average validation and training loss with a learning rate of 1×10^{-5} over 6 epochs

Figure 7.1.: Average validation and training loss over five folds for 6 epochs

7.1.3. Final Evaluation on Test Set

Results

Table 7.2 illustrates the results from the fine-tuned NB BERT-based models on the test set, highlighting that the best performing model was trained with 2 epochs and a learning rate of 1×10^{-5}. This model achieved an accuracy of 0.8275, an F1 score of 0.8239, a precision of 0.8206, and a recall of 0.8525.

Table 7.2.: This table presents the accuracy and macro-average F1 score, precision and recall of the test set obtained from the NB BERT-based models with different learning rates and numbers of epochs.

Learning Rate	Epochs	Accuracy	F1 score	Precision	Recall
1e-6	6	0.8028	0.7969	0.7895	0.8292
1e-5	**2**	**0.8275**	**0.8239**	**0.8106**	**0.8525**

Table 7.3 showcases the performance metrics specifically for the private-sensitive class. Similarly, the best performing model was trained with 2 epochs and a learning rate of 1×10^{-5}. This model achieved an F1 score of 0.8103, a precision of 0.7015, and a recall of 0.9592.

Table 7.3.: This table presents the macro-averages performance on the test set for the private-sensitive class.

Learning rate	Epochs	F1-score	Precision	Recall
1e-6	6	0.7603	0.6389	0.9388
1e-5	**2**	**0.8103**	**0.7015**	**0.9592**

Figure 7.2 shows the confusion matrix of predicted labels and true labels on the test data set using the NB BERT-based classifier with the hyperparameter combination that yielded the best results on the test set (2 epochs with a learning rate of 1×10^{-5}). The matrix indicates the number of samples classified correctly and incorrectly for each label category. It provides insights into the performance of the classifier for different label categories. The diagonal elements represent the number of correctly classified samples, while off-diagonal elements indicate misclassifications.

The "Non Private-sensitive" category had 87 correct predictions out of 115 instances. Some misclassifications occurred, with 16 instances predicted as "Private-sensitive" and 12 instances as "Other." For the "Private-sensitive" category, the model performed well by correctly predicting 47 out of 49 instances. 2 instances were misclassified as "Non Private-sensitive." The "Other" category had the largest number of instances, and the model correctly predicted a majority of them, with 101 out of 120 instances. 15 instances were predicted as "Non Private-sensitive" and 4 instances as "Private-sensitive."

Discussion

Instead of solely relying on accuracy we have utilized additional performance metrics. The choice to use the macro-average of F1-score, recall, and precision, was deliberate in order to prioritize the smaller class of private-sensitive instances.

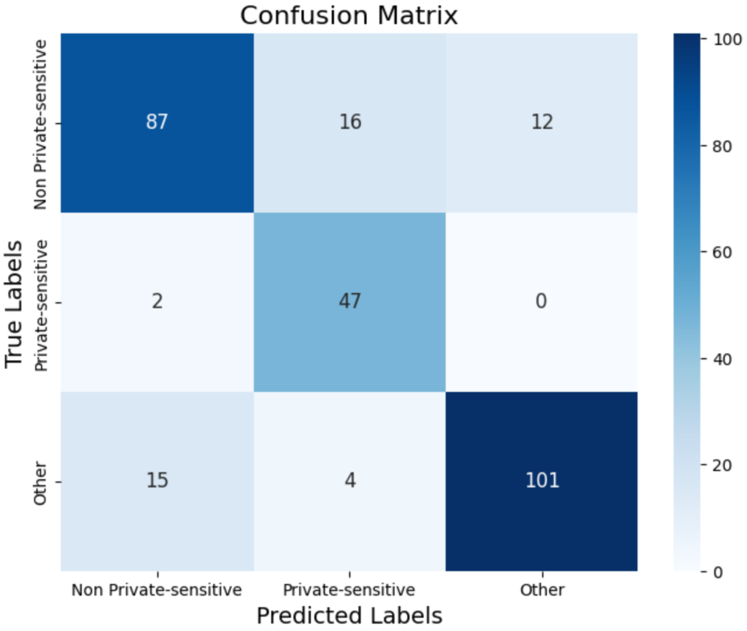


Figure 7.2.: Confusion matrix of predicted labels and true labels on the test data set with the NB BERT-Base classifier using 2 epoch with a learning rate of 1e-5

The overall macro-average F1-score on the test set highlighted in Table 7.2 is 0.8239, with slightly lower precision and somewhat higher recall. These results suggest that the model performs quite well on the new unbalanced and unseen data. The accuracy achieved, displayed in 7.2 is 0.8275 which is somewhat higher than the validation average accuracy of 0.8022 obtained during cross-validation. This can be due to the test data set being unbalanced or simply that the models utilized the whole training set gain 20% more instances to train on.

Considering our research question, which specifically focuses on detecting private-sensitive data, our primary interest lies in evaluating the model's performance on this particular class. The F1-macro score for the private-sensitive label was slightly lower at 0.8103, with a considerably higher recall of 0.9592 and a lower precision of 0.7015 compared to the overall performance scores on the test set. These results suggest that the model is adept at recognizing and correctly labelling instances that are private-sensitive, as further evidenced by the confusion matrix where only 2 private-sensitive instances were misclassified.

However, the lower precision highlighted by the confusion matrix indicates that the model predicted a substantial number of instances (20) as private-sensitive that were not actually in that category. This could be attributed to the class imbalance in the test data, where the model may be inclined to predict more instances as private-sensitive due to its training on a balanced data set. It is also possible that the model occasionally struggles to differentiate between non-sensitive and private-sensitive instances, as shown in the confusion matrix 7.2.

In summary, the fine-tuned NB BERT-based model demonstrates promising performance in detecting private-sensitive data, as reflected in its higher recall and reasonably high F1-score. However, the lower precision suggests room for improvement in accurately discerning between non-sensitive and private-sensitive instances.

7.2. Detecting Private-Sensitive Content using Conventional Classifiers

This section presents the results from experiment 2 followed by a discussion of each result.

7.2.1. Grid Search Results

Result

Table 7.4 presents the average validation accuracy for the parameter combinations chosen during grid search. Graphs displaying the validation accuracy and training accuracy for each parameter combination used in the grid search for Multinomial LR, Multinomial NB, Random Forest and Linear SVM are visualized in respectively Appendix B, C,D and E.

Table 7.4.: The chosen parameter combinations

Classifier	Best Parameters	Average Accuracy
Multinomial LR	**C: 0.1, solver: saga, multi_class: multinomial**	**0.7250 ± 0.00039**
Multinomial NB	alpha: 5	0.6262 ± 0.00031
Random Forest	n_estimators: 200, max_depth: 4	0.6566 ± 0.00048
Linear SVM	C: 0.001, penalty: l2	0.6688 ± 0.00052

Table 7.4 displays the selected parameter combinations and their respective accuracies. For Multinomial LR, the chosen parameters are: ’C’ set to ’0.1’, ’solver’ set to ’saga’, and ’multi_class’ set to ’multinomial’. For Multinomial Naive Bayes, the parameter ’alpha’ is set to 5. In the case of Random Forest, the parameter ’n_estimators’ is set to 200 and ’max_depth’ is set to 4. Finally, for Linear SVM, the parameter ’C’ is set to 0.001 with the penalty set to "l2".

The average accuracy represents the overall accuracy of each classifier on the data set, considering all folds. According to the results presented in Table 7.4, the average accuracies range from 0.6262 to 0.7250. Notably, Multinomial LR exhibits the best average accuracy of 0.7250 with a variance of 0.00039.

Appendix B, C,D and E contains graphs that depict the training and validation accuracy for each parameter combination. In Table 7.5, we present a subset of these combinations, highlighting their respective validation accuracy and training accuracy. It's important to note that these results were selected for discussion purposes and may not necessarily represent the best performing combinations.

Table 7.5.: Hyperparameter combinations with corresponding training and validation accuracy. The chosen combinations are highlighted in bold text.

Classifier	**Hyperparameter combinations**	**Training Accuracy**	**Validation Accuracy**
Multinomial LR	C: 10.0, solver: saga, multi_class: multinomial	1.0	0.74
Multinomial LR	**C: 0.1, solver: saga, multi_class: multinomial**	**0.81**	**0.73**
Multinomial LR	C: 0.001, solver: saga, multi_class: multinomial	0.74	0.69
Multinomial NB	alpha:0.0001	0.97	0.73
Multinomial NB	alpha: 100	0.62	0.58
Multinomial NB	**alpha: 5**	**0.72**	**0.63**
Random Forest	**n_estimators: 200, max_depth: 4**	**0.69**	**0.66**
Random Forest	n_estimators: 50, max_depth: 1	0.62	0.62
Linear SVM	C: 0.5, penalty: l2	0.99	0.74
Linear SVM	C: 0.0001, penalty: l2	0.66	0.62
Linear SVM	**C: 0.001, penalty: l2**	**0.73**	**0.67**

Discussion

During the grid search, we intended to carefully select a parameter combination that could yield high accuracy without causing overfitting on the training data. The following discussion on how we arrived at the selected parameter combination will frequently refer to Table 7.5.

It appeared that using the 'saga' solver for Multinomial LR led to higher validation accuracy without widening the gap between training and validation accuracy. When using a value of 10 for 'C', the validation accuracy is 0.74, but the training accuracy is 1.0. The large gap in training accuracy and validation accuracy could mean it is severely overfitting. Choosing a low 'C' value, like 0.001, could reduce the gap between validation and training accuracy of 0.05, but also came at the expense of a lower overall validation accuracy of 0.69. To navigate this trade-off, we opted for a 'C' value of 0.1, which yielded a training accuracy of 0.81 and a validation accuracy of 0.73. While we acknowledge the remaining risk of overfitting, we selected this combination with the belief that it could strike a balance between achieving a high validation accuracy and avoiding excessive overfitting.

For Multinomial Naive Bayes, employing a small alpha value like 0.0001 resulted in an accuracy of 0.73, but the training accuracy was 0.97, creating an accuracy gap of 0.24. Conversely, the lowest accuracy gap of 0.04 was achieved with an alpha value of 100, but it led to a considerably lower validation accuracy of 0.58. In line with our approach for Multinomial LR, we aimed to find a combination that could strike a balance between reaching a high validation accuracy and not overfitting the training data. Consequently, we settled on an alpha value of 5, which yielded a validation accuracy of 0.63 and a training accuracy of 0.72.

In the case of Random Forest, the highest accuracy of 0.66 was achieved with 'n_-estimators' set to 200 and 'max_depth' set to 4. This combination resulted in a modest accuracy gap of 3%. Although a 'max_depth' of 1 almost eliminated the gap between training and validation accuracy, it resulted in a lower validation accuracy of 0.62. Considering these factors, we concluded that a 0.3 accuracy gap was acceptable to achieve a higher validation accuracy, indicating a lower risk of overfitting the training data.

For Linear SVM, the highest validation accuracy of 0.74 was achieved by setting 'C' to 0.5. It resulted in a training accuracy of 0.99, indicating that it may be overtraining on the training data, which can lead to overfitting. When C was 0.0001, it gave a lower validation accuracy of 0.62 with a training and accuracy gap of 0.04. We therefore also choose the middle road with 'C' set to 0.001, which gives an accuracy gap of 0.6 with a validation accuracy of 0.67.

Despite utilising grid search and cross-validation, finding the optimal parameter combination that would deliver superior performance on unseen data posed a challenge. Our objective was to strike a balance between mitigating overfitting and underfitting while achieving a high validation score. The risk of overfitting can be more pronounced given the small data set. Although we decided to discard parameter combinations that appeared to overfit, because they had a substantially higher training accuracy compared to validation accuracy, it is important to note that such combinations may still exhibit favourable performance, since it still has a higher validation accuracy.

7.2.2. Final Evaluation on the Test Set

Result

Table 7.6 presents the results of the classifiers performance on the unbalanced test set with the selected parameter combinations from Table 7.4. The accuracy and the macro averaged F1-score, precision score and recall score is presented. For all metrics, the Multinomial LR achieved the best performance with an accuracy of 0.7430, F1-score of 0.7290, precision of 0.7228 and a recall of 0.7637.

Table 7.6.: This table presents the accuracy and macro-average F1 score, precision and recall on the test set obtained from the conventional classifiers.

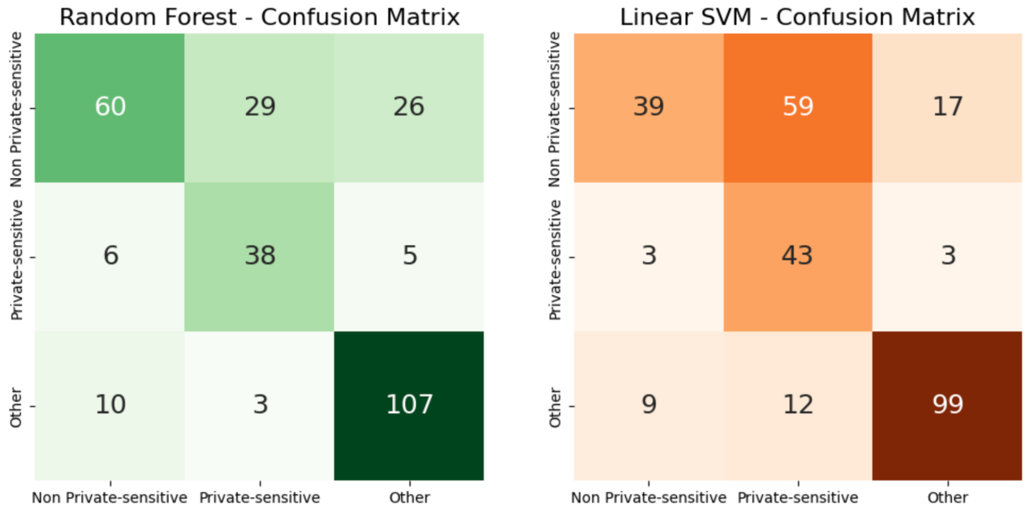
Classifier	Accuracy	F1-score	Precision	Recall
Multinomial LR	**0.7430**	**0.7290**	**0.7228**	**0.7637**
Multinomial Naive Bayes	0.5563	0.5383	0.6610	0.6312
Random Forest	0.7218	0.6988	0.7026	0.7296
Linear SVM	0.6373	0.6086	0.6579	0.6806

Table 7.7 displays the performance metrics specifically for the private-sensitive class with the same selected parameters. Multinomial LR achieves the highest F1-score of 0.6666 and precision of 0.5455. Multinomial Naive Bayes archives the best Recall of 0.9592. Multinomial LR demonstrates the smallest decline in F1-score, with an approximate decrease of 0.08, while achieving the highest F1-score.

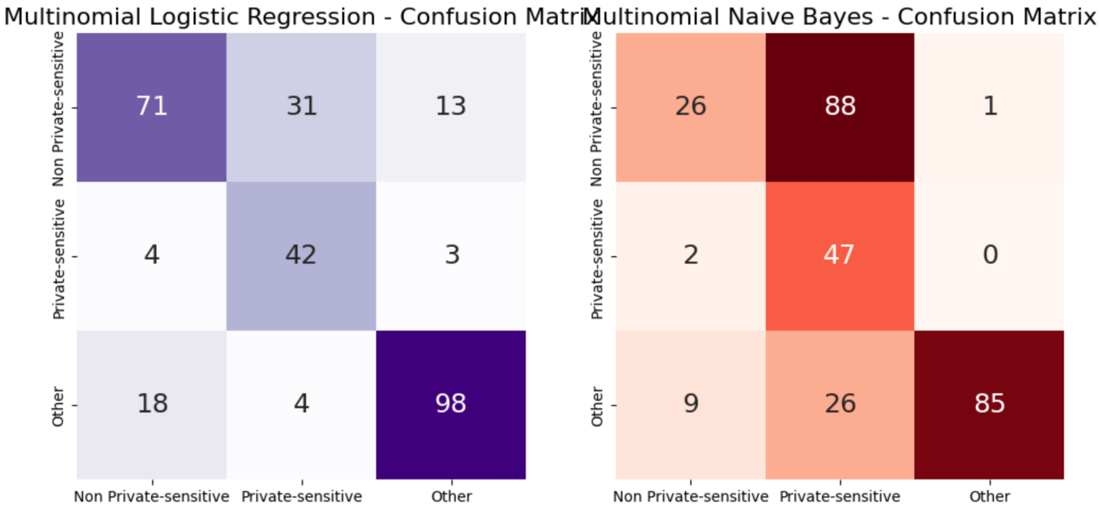
Classifier	F1-score	Precision	Recall
Multinomial LR	**0.6666**	**0.5455**	0.8571
Multinomial Naive Bayes	0.4476	0.2919	**0.9592**
Random Forest	0.6386	0.5429	0.7755
Linear SVM	0.5276	0.3772	0.8775

Table 7.7.: This table presents the macro-average performance on the test set for the private-sensitive class

Figure 7.3a and 7.3b show the confusion matrix of predicted labels and true labels for the conventional classifiers on the test data set using the selected hyperparameter combination. The confusion matrix indicates the number of samples classified correctly and incorrectly for each label category. It provides insights into the performance of the classifiers for different label categories. The diagonal elements represent the number of correctly classified samples, while off-diagonal elements indicate misclassifications.



(a) Confusion Matrix of Predicted Labels and True Labels on the Test Data set with the classifiers Random Forest and Linear SVM



(b) Confusion Matrix of Predicted Labels and True Labels on the Test Data set with the classifier Multinomial LR and Multinomial Naive Bayes

Discussion

Both Multinomial LR and Random Forest models demonstrated higher accuracy on the overall test set in comparison to the accuracy observed on the validation sets during parameter tuning. These results indicate that these models adapt well to the unbalanced test data. Notably, the Random Forest model exhibited an accuracy improvement of approximately 0.07, which could be attributed to addressing potential underfitting issues. The Multinomial LR classifier showed a smaller increase of 0.02 in accuracy, which might also be related to underfitting. Linear SVM and Multinomial Naive Bayes exhibited lower accuracy and macro-averaged F1-scores on the test set than on the validation sets. The decrease in performance for Linear SVM and Multinomial Naive Bayes might be due to overfitting on the training data or their struggle to adapt to the unbalanced test data after being trained on a balanced data set.

The macro-averaged F1-scores for the classifiers, in general, tend to be slightly lower than their corresponding accuracy scores, as shown in Table 7.6. This may suggest that while the classifiers achieve relatively high accuracy in predicting the majority classes, they struggle to maintain the same level of performance in accurately identifying and classifying instances from the minority classes.

In the context of our research question, which focuses on detecting private-sensitive content, the performance of the classifiers on this specific class becomes highly relevant. Interestingly, when comparing their performance on the private-sensitive class to the overall results, it is evident that the classifiers exhibit a decrease in F1-score. This observation may suggest that the classifiers struggle more when dealing with instances from the private-sensitive class. Analyzing precision and recall provides further insights into the classifiers' performance on the private-sensitive class. It is worth noting that the classifiers exhibit a significantly higher recall than precision, suggesting their ability to accurately predict true labels for private-sensitive content more effectively than falsely predicting other classes as private-sensitive. This pattern is further supported by the examination of the confusion matrices. Interestingly, Multinomial Naive Bayes stand out with the highest recall, missing only two predictions for private-sensitive instances. However, it also predicts a substantial number of instances (161) to be private-sensitive when only 47 actually belong to this category, resulting in a low precision.

Considering the overall performance of the classifiers on the test set, as well as their performance specifically on the private-sensitive class, Multinomial LR emerges as the best performing classifier across various metrics, followed by Random Forest. Despite Multinomial Naive Bayes and Linear SVM achieving slightly higher recall, the Multinomial LR achieved the overall highest F1-score and precision.

7.3. General Discussion

7.3.1. Comparing Methods and Results from the Experiments

In Table 7.8 we repeat the results of the best performing model from experiments 1 and 2 for comparison.

Classifier	Class	F1-score	Precision	Recall
NB BERT-based	all classes	**0.8239**	**0.8106**	**0.8525**
Multinomial LR	all classes	0.7430	0.7290	0.7637
NB BERT-base	private-sensitive	**0.8103**	**0.7015**	**0.9592**
Multinomial LR	private-sensitive	0.6666	0.5455	0.8571

Table 7.8.: Evaluation Metrics for Best Performing Models: Comparing the NB BERT-based and Conventional Classifiers on Test Set and Private-Sensitive Class.

The fine-tuned NB BERT-based model overall achieves the highest performance metrics on the unbalanced test set. This result shows that the NB BERT-based model is generally better at predicting all three classes in the test data. With a macro-averaged F1-score of 0.82, NB BERT-based model outperforms Multinomial LR. However, both models experience a decrease in F1-score when focusing solely on the private-sensitive class. Notably, Multinomial LR's F1-score drops from 0.74 to 0.67, whereas the NB BERT-based model experiences a drop down to 0.81. This suggests that the NB BERT-based model still performs relatively well on the private-sensitive class compared to Multinomial LR, which is particularly significant considering the goal of detecting private-sensitive content.

The better performance of the NB BERT-based model, which is based on the transformer architecture, may stem from a variety of factors. One possible reason could be the transformer model's inherent capability to effectively capture the contextual relationships between words and phrases. This aspect might prove advantageous, especially in predicting private-sensitive content, where the context often plays a significant role. Additionally, the incorporation of fine-tuning, specifically through MLM, is likely an important factor in the improved performance of the NB BERT-based model.

It is important to acknowledge the differences in methods between experiments 1 and 2, which can complicate a direct comparison of the results. Despite our efforts to maintain similarities by employing hyperparameter tuning with grid search, considering accuracy, overfitting/underfitting analysis, and cross-validation, there are still variations. One notable difference is the domain adaptation method used in experiment 1 for the NB BERT-based model, providing it with an advantage in acquiring a general understanding of Norwegian social media language before initiating classification training. The analysis of overfitting and underfitting also differs, with experiment 1 focusing on validation loss and training loss over epochs for the NB BERT-based model, while experiment 2 considers validation accuracy and training accuracy over training size for the conventional classifiers. These differences may influence the selection of optimal hyperparameter combinations. Additionally, it is worth mentioning that the NB BERT-based model requires more computational resources and training time compared to the conventional classifiers due to its increased complexity. This trade-off between complexity and computational requirements should be taken into account when deciding on a model.

7.3.2. Limitations of Experiments

We encountered limitations in both experiments. The most significant limitation revolved around the limited number of labelled entries. Increasing the number of entries could have potentially enhanced the results and improved the models' robustness. In the case of the experiment using the NB BERT-based model, we faced the constraint of a 16 GB memory size, which restricted us to a batch size of 16. Exploring larger batch sizes could have improved the model's performance. Additionally, conducting a more comprehensive grid search by testing additional parameters might have further enhanced the results, but we did not do due to restricted time and resources. Furthermore, it is worth mentioning that BERT models have a maximum token limit of 512, which led us to remove 40 entries from consideration.

7.3.3. Comparison with Related Work and Addressing Limitations in Our Work

In order to discuss our work in light of research within the domain of private-sensitive detection, we will compare our work to studies addressed in Chapter 3. Direct comparisons, however, are difficult for a variety of reasons. Variations in training and test set sizes, differences in the definitions used to determine what qualifies as private-sensitive, and the fact that the test set used in our work has not been validated on the other systems mentioned are all examples of this. A comparison with results for detecting private-sensitive content in Norwegian is impossible as we did not find any research papers specifically addressing this.

In this section we will compare our approach of defining private-sensitive content, our data set creation and our results from the experiments with related work. Furthermore, we will address limitations within our work.

Definition in Comparison to Related Work

Similar to our work, Petrolini et al. (2022) focuses on GDPR's definition of personal data and sensitive data. However, in their task of detection, they simplified it to determine whether the input contained one of four sensitive topics. Contradicting GDPR's definition, they removed the link between the privacy concerning content and the individual. Similarly to the work of Bioglio and Pensa (2022), our definition requires a link to an individual. In our definition, it is sufficient that the author directly or implicitly links themselves or other relatives or close friends to the private-sensitive content. We think it is important to differentiate between a private-sensitive topic in general and private-sensitive information regarding an individual. This is especially important for our long-term goal of warning users before they publish something online. To that extent, our work is different from the task of Petrolini et al. (2022) which is motivated to provide companies with better knowledge of the data they possess. Considering that companies retrieve severe fines for non-compliance with GDPR, this simplification can be valuable in ensuring that sensitive data is not overlooked during the detection process. However, in the context of an alarm system for users on social media, this simplification can be more problematic as it may result in an excessive number of warnings. For instance, it can trigger alerts even when health-related posts are unrelated to the author or other individuals mentioned.

Despite aiming to align our definition with GDPR, it has some contradictions with GDPR's definition. For example, physical appearance is not part of our definition. However, we argue that in terms of the goal of creating an alarm system for social media, it may be excessive to issue warnings to users based on all aspects of the broad definition of GDPR. As we have made decisions like this throughout the definition process, we acknowledge that it reflects our subjective opinions as well. Despite efforts to obtain an objective definition in line with GDPR, there have been choices which reflect our own subjective view. A consequence of this is that our definition does not have complete compliance with GDPR.

Data Set Creation in Comparison to Related Work

In the work Bioglio and Pensa (2022), the authors rely on a corpus of 9917 Facebook posts collected by Facebook over ten years ago due to restrictions set by the Facebook API. In contrast, the authors of the work Petrolini et al. (2022) collected the "hottest" posts from Reddit, which include data that are up-to-date. This aligns with our approach, as we gathered our data from Reddit and explicitly focused on gathering the newest posts available at the time of data collection. For the purpose of detecting private-sensitive content in social media, we argue that data from personal Facebook profiles are better than posts from Reddit as the usernames often are pseudonyms. However, as described in

Chapter 5 obtaining such data has legal issues. While this ensures users' privacy, it also presents a significant obstacle to our task. As our long-term goal is warning user before posting something that may contain private-sensitive content, we believe it is important to have trained the data on up-to-date language used on social media.

In contrast to our work, Petrolini et al. (2022) collected their data from over 60 different subreddits named after topics such as health, politics, sexuality and other sensetive topics. Using multiple data sources within Reddit may lead to a more diverse data set which contains a more balanced representation of different types of private-sensitive content. This biased approach can also help in obtaining a sufficient amount of private-sensitive data. However, our work faces a limitation due to the scarcity of subreddits containing Norwegian language, and only two subreddits were utilized. As presented in Chapter 5, our subcategories within the private-sensitive main category are skewed. The majority includes content that may reveal political or religious beliefs, while other subcategories, such as PII, have very few instances in our labelled data set. This is a significant limitation in terms of training a model to detect private-sensitive content involving the underrepresented subcategories. Another limitation of our work is that we only use two subreddits from one platform as our source, and we have not tested the models' ability to generalize on other social media platforms. Hence, our results present the performance on the language from the two subreddits, but we do not know how well they would perform on other platforms such as Twitter or Facebook. We also note that our models are limited to the training set, and their ability to adapt to new events within politics or other social changes may therefore be challenging.

As Fleiss' kappa can be affected by the difficulty of the data each group of annotators label, we do not aim to draw a direct comparison between related work's results, but rather the overall trends observed. The annotation process conducted in the work of Bioglio and Pensa, contained 12 annotators where each annotator labeled approximately 2480 entries. Since our guidelines are inspired by their guidelines, there are similarities in terms of determining whether an entry is private-sensitive, unknown, unintelligible or non-sensitive. In the work Bioglio and Pensa (2022), the Fleiss' kappa obtained from each group of three annotators was between 0.22 and 0.42. . In contrast, our data sets labelled by three annotators exhibited a substantial difference, with Fleiss' kappa scores of 0.25 and 0.77 which can be considered a fair and substantial agreement, respectively. This can show variance in the quality of annotation, but also be due to the variance in the data being labelled in the two groups. We note that the group with the highest score contains multiple English entries, which we argue is relatively easier to determine as unintelligible than the other entries. Similar to our work, the authors experience a decrease when all three annotators agree versus when at least two agree. This may suggest that the task of annotating into one of the four categories is a challenging task. This can be due to the subjectivity of privacy and what people think of as private-sensitive.

Experiments in Comparison to Related Work

As addressed in Chapter 3, the work of Petrolini et al. (2022) provided a flat multi-label model obtaining an F1 score of 0.94 on the binary task of detecting whether an input belongs to a sensitive topic or not. The authors also obtained a F1-score of 0.95 on a binary model which determines whether the post contains a sensitive topic or not. Similar to their research, we utilize BERT-base as the model base. However, we specifically utilize the NB BERT-base model for the Norwegian language, as our focus is on detecting Norwegian language instead of English. Our trained NB BERT-base model obtained a lower macro F1-score on the multi-label task presented in our work. The higher F1 scores observed in their study may be influenced by factors such as the simplification of the annotation task or the utilization of a larger data set during training and testing, which differs from our approach.

In the work of Bioglio and Pensa (2022) they treat private-sensitive detection as a binary problem between sensitive and non-sensitive content. Hence they remove the unintelligible and unknown classes. In contrast, we merged these classes together as we do not want to directly assume that these do not contain private-sensitive content. On the binary task, the authors obtained a macro average F1-score of 0.78 on their SENS2 data set. In more detail, they obtained an F1-score of 0.73 for the sensitive class. Our fine-tuned NB BERT-Base model obtained a higher macro average F1-score of 0.82 on the multi-label task and obtained an F1-score of 0.81 for the private-sensitive class.

Similarly to our work, their SENS2 data set included entries where at least two annotators agreed on the annotation. Moreover, they obtained a macro-average F1-score of 0.89 on their SENS3 data set which contained entries where all annotators agreed. However, as we had limited resources for annotation we did not have enough labeled data within all classes to train our model on a data set only including the entries where all annotators agree. However, their data set consisted of more entries than ours. SENS2 contained 8765 posts where 3336 were labelled as sensitive, and SENS3 contained 4046 entries where 1444 were labelled as sensitive. However, our training set contained 2820 entries where 940 were labelled as sensitive. As we have trained our model on less entries, our results may not be as robust as the related work conducted using more data.

Our work aims to empower Norwegian users when sharing information online to protect their privacy. However, we recognize the ethical issues with the work conducted in this field, as it has the potential for being misused to detect personal sensitive information about others without their consent. Finally, we acknowledge the difficulties of obtaining data sets for research purposes for detecting private-sensitive content. The dilemma arises from the inherent contradiction within the problem domain itself. On one hand, the objective is to protect users from disclosing private-sensitive information. On the other hand, achieving this goal often requires the collection of potentially sensitive data from users. As our purpose is to protect users' privacy online, we have only used public data and none of the usernames are included in the data set.

8. Conclusion and Future work

In this chapter, we present the final conclusions of our thesis. We will specifically address the research questions and highlight the contributions our thesis has made to the field. In addition, we will offer suggestions for future research to address the limitations of our work, enhance its quality, and further progress towards our goal.

8.1. Conclusion

In this thesis, we collected a data set from Norwegian subreddits, and through our annotation process obtained a labelled data set of 4482 entries. We explored four conventional classifiers and one NB BERT-based classifier on the task of detecting private-sensitive content in Norwegian social media language utilizing the labelled data set. The results from the different classifiers have been compared. According to our results, the NB BERT-based model performed better than conventional classifiers. This is in line with other research noted in Chapter 3. However, as more data is always desirable, it is interesting to see such promising results given the small data set utilized. The NB BERT-based model performed well on the task of classifying text into private-sensitive, non-sensitive and other with a macro F1 score of 0.82, and an F1 score of 0.81 for the private-sensitive class. Among the conventional classifiers, the Multinomial LR model performed best with a macro F1-score of 0.74 and an F1-score of 0.66 for the private-sensitive class. However, we note that utilizing more data from different Norwegian social media platforms can help to make the model better to generalize to Norwegian social media language.

8.1.1. Research Questions

This section provides insight into how the research questions have been addressed in this thesis.

RQ1 *How can we define private-sensitive content and optimize the labels for classification?*

To define private-sensitive content and optimize the labels for classification, we conducted a literature review and tested different labels for annotation. Based on the literature review findings, we decided to develop a definition that draws inspiration from existing work while also adhering to the GDPR's definition of personal and sensitive data. The definition has a trade-off between following GDPR's definition and avoiding being overly broad in terms of alerting users when sharing information online. We iteratively refined and optimized the definition for labelling purposes. The resulting categories and subcategories were designed to align with GDPR and minimize overlap. We established four main categories: Unknown, Unintelligible, Private-sensitive, and Non-sensitive, along with seven subcategories specifically addressing private-sensitive content.

RQ2.1 *How do conventional classifiers perform in detecting private-sensitive content using the collected data set?*

We trained and evaluated several conventional classifiers, namely Random Forest, Multinomial Naive Bayes, Linear SVM, and Multinomial LR. The conventional classifiers were trained using TF-IDF vectorization. Among these classifiers, the Multinomial LR model exhibited the highest performance, achieving a macro F1 score of 0.74. Specifically, for the private-sensitive class, it attained an F1 score of 0.66.

RQ2.2 *How does the NB BERT-based model, which is a transformer-based model, perform in detecting private-sensitive content using the collected data set?*

The NB BERT-based model performed well on the task of classifying text into private-sensitive, non-sensitive and other with a macro F1 score of 0.82. Specifically, for the private-sensitive class, it obtained an F1 score of 0.81.

RQ2 *How do different classifiers perform on detecting private-sensitive content in Norwegian social media?*

In order to answer RQ2, we compare the results from RQ2.1 and RQ2.2 to provide information on how conventional classifiers and the NB BERT-based classifier perform on the collected data set. According to our results, the NB BERT-based model performed better than the conventional classifiers. It also demonstrated comparable performance specifically on the private-sensitive class, in contrast to the conventional classifiers that experienced a significant decline in performance. These findings seem to align with previous research mentioned in Chapter 3, suggesting that transformer-based models may be more effective in capturing the contextual sensitivity in the text compared to conventional classifiers. With the limited data set used, it is interesting to see promising results given the small data set utilized. It is important to acknowledge that the data is collected solely from Reddit, which may limit the generalizability of the model to other social media content in Norwegian.

8.1.2. Contributions

Our main contribution to the research field of private-sensitive detection is a labelled data set consisting of content from Norwegian language on social media. In order to obtain up-to-date language, we collected the newest entries available from the social media platform Reddit. As mentioned in Chapter 3, research within the detection of personal or private-sensitive information struggles with obtaining a data set containing English language. Furthermore, to the best of our knowledge, there are no available labelled data sets in Norwegian for the task of private-sensitive detection. Hence, we address the existing gap in resources available for training and evaluating private detection models specifically in this language domain.

Another contribution is our definition which draws inspiration from existing work while also adhering to the GDPR's definition of personal and sensitive data. This definition has the potential to assist in future private-sensitive detection. Our contribution provides a definition of private-sensitive content. Our definition has the potential to assist in ensuring compliance with privacy regulations and promoting effective data protection practices in social media content in the Norwegian language.

Additionally, we make available a fine-tuned NB BERT-base model specifically designed for Norwegian social media text. This model has been trained on a data set of 12,000 unlabeled entries obtained from two Norwegian channels on Reddit. It can serve as a valuable tool for analyzing and processing Norwegian social media content.

Our last contribution is a fine-tuned multi-class NB BERT-base model trained on Norwegian language derived from Reddit's social media platform. This model has the capability to classify content within the data set created for this thesis into three categories: Private-sensitive, Non-sensitive, and Other (combining unintelligible and unknown sensitivity). It provides a means to assess the sensitivity of social media content in the Norwegian language.

8.2. Future Work

We will now present areas for future work that aim to address the limitations within our work. Additionally, we suggest further work to improve our work in order to progress towards achieving our overall goal of detecting and warning Norwegian users before private-sensitive content is published.

- **Multi-label model for detecting subcategories:** Developing a multi-label model for predicting the subcategories may be useful in terms of the long-term goal of developing an alarm system. It can be interesting to weigh different subcategories stronger than others in order to align to a specific user's preferences.

- **Expanding data sources:** In order to obtain a model that generalizes Norwegian social media, it can be interesting to expand the data set with entries from different sources. This would both improve the data amount and introduce diversity in terms of data sources. Furthermore, testing it on unseen data from different platforms can help improve the evaluation of the model's performance. However, this would demand resources in terms of annotation.

- **Further fine-tuning:** In order to stay relevant to new events in the political landscape or other social changes, it can be useful to further fine-tune the BERT model on up-to-date language on social media platforms.

- **Amount of labelled data:** Annotation is a time-consuming task and our labelled data amount is limited for this reason. Consequently, the underrepresentation of certain private-sensitive subcategories poses a limitation in terms of detecting these. To address this limitation, it would be beneficial to conduct further research in determining the necessary amount of labelled data required for effectively detecting all of the distinct subcategories of private-sensitive data.

- **Comparing classification with other transformer based models:** It would be interesting to compare our results from the NB BERT-based model, with other state-of-the-art transformer based models, such as GPT3.

- **Computational time:** In order to achieve our goal of warning users prior to sharing private or sensitive information, it would be beneficial to look into the computational time required by the model. This could help assess the time taken by the model to generate predictions.

- **Notify users:** In order to leverage the practical applications of the model, we recommended developing a system that delivers real-time notifications to users, utilizing the predictions generated by our model. This can be accomplished by integrating the model with an existing notification service or constructing a tailored notification system. Furthermore, it would be valuable to gather user feedback to gain deeper insights into the practical utility of the prediction model.

Bibliography

Alessandro Acquisti, Laura Brandimarte, and George Loewenstein. Privacy and human biasehavior in the age of information. *Science (New York, N.Y.)*, 347:509–514, 2015.

Ken Barker, Mina Askari, Mishtu Banerjee, Kambiz Ghazinour, Brenan Mackas, Maryam Majedi, Sampson Pun, and Adepele Williams. A data privacy taxonomy. In *Dataspace: The Final Frontier: 26th British National Conference on Databases, BNCOD 26, Birmingham, UK, July 7-9, 2009. Proceedings 26*, pages 42–54. Springer, 2009.

Trisha Dowerah Baruah. Effectiveness of social media as a tool of communication and its potential for technology enabled connections: A micro-level study. *International journal of scientific and research publications*, 2(5):1–10, 2012.

Livio Bioglio and Ruggero G. Pensa. Analysis and classification of privacy-sensitive content in social media posts. *SpringerOpen Journal*, 1:1–24, 2022.

Haldis Borgen and Oline Zachariassen. Preparing for the master thesis "gdpr and the opportunities it brings. Project report in TTM4502, Department of Information Security and Communication Technology, NTNU – Norwegian University of Science and Technology, Dec. 2022.

Hinrich Schütze Christopher D. Manning, Prabhakar Raghavan. *Introduction to Information Retrieval*, volume 39, page 281. Cambridge University Press, 2008.

Denzil Correa, Leandro Araújo Silva, Mainack Mondal, Fabrício Benevenuto, and Krishna P. Gummadi. The many shades of anonymity: Characterizing anonymous social media content. *Proceedings of the International AAAI Conference on Web and Social Media*, 9:71–80, 2015.

Silva L. Mondal M. Benevenuto F. Gummadi Correa, D. Privacy dictionary: A new resource for the automated content analysis of privacy. *Proceedings of the International AAAI Conference on Web and Social Media*, 9:71–80, 2015.

Gabriela Csurka. Domain adaptation for visual applications: A comprehensive survey. *arXiv preprint arXiv:1702.05374*, 2017.

Adele Cutler, D Richard Cutler, and John R Stevens. Random forests. In *Ensemble machine learning*, pages 157–175. Springer, 2012.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1912.07076*, October 2018.

Meng Joo Er, Rajasekar Venkatesan, and Ning Wang. An online universal classifier for binary, multi-class and multi-label classification. In *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 003701–003706, 2016.

European Parliament and Council of the European Union. Regulation (EU) 2016/679 of the European Parliament and of the Council. URL `https://data.europa.eu/eli/reg/2016/679/oj`.

Rosa Falotico and Piero Quatto. Fleiss' kappa statistic without paradoxes. *Quality & Quantity*, 49:463–467, 2015.

Cyril Goutte and Eric Gaussier. A probabilistic interpretation of precision, recall and f-score, with implication for evaluation. In *European conference on information retrieval*, pages 345–359. Springer, 2005.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning (2nd ed.)*, pages 587–593. Number 1. Springer, New York, USA, August 2009a.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning (2nd ed.)*, pages 210–211. Number 1. Springer, New York, USA, August 2009b.

Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009c.

Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *Resampling Methods*, pages 175–201. Springer New York, New York, NY, 2013. ISBN 978-1-4614-7138-7.

Christian Janiesch, Patrick Zschech, and Kai Heinrich. Machine learning and deep learning. *Electronic Markets*, 31(3):685–695, 2021.

Hang Jiang, Yining Hua, Doug Beeferman, and Deb Roy. Annotating the tweebank corpus on named entity recognition and building nlp models for social media analysis. *arXiv preprint arXiv:2201.07281*, 2022.

Michael I Jordan and Tom M Mitchell. Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245):255–260, 2015.

Sotiris B Kotsiantis, Ioannis Zaharakis, P Pintelas, et al. *Supervised machine learning: A review of classification techniques*, volume 160, pages 3–24. IOS Press, Amsterdam, Netherlands, 2007.

Per E Kummervold, Javier De la Rosa, Freddy Wetjen, and Svein Arne Brygfjeld. Operationalizing a national digital library: The case for a norwegian transformer model. *arXiv preprint arXiv:2104.09617*, 2021.

Tarald O Kvålseth. Note on cohen's kappa. *Psychological reports*, 65(1):223–226, 1989.

Chanyeong Kwak and Alan Clayton-Matthews. Multinomial logistic regression. *Nursing research*, 51(6):404–410, 2002.

J. Richard Landis and Gary G. Koch. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):164–165, 1977. ISSN 0006341X, 15410420.

Hongmin Li, Nicolais Guevara, Nic Herndon, Doina Caragea, Kishore Neppalli, Cornelia Caragea, Anna Cinzia Squicciarini, and Andrea H Tapia. Twitter mining for disaster response: A domain adaptation approach. In *ISCRAM*, 2015.

Petro Liashchynskyi and Pavlo Liashchynskyi. Grid search, random search, genetic algorithm: a big comparison for nas. *arXiv preprint arXiv:1912.06059*, 2019.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35, 2023.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Patrick Murmann and Farzaneh Karegar. From design requirements to effective privacy notifications: Empowering users of online services to make informed decisions. *International Journal of Human–Computer Interaction*, 37:1823–1848, 2021.

Kevin P Murphy et al. Naive bayes classifiers. *University of British Columbia*, 18(60): 1–8, 2006.

Sreerama K Murthy. Automatic construction of decision trees from data: A multi-disciplinary survey. *Data mining and knowledge discovery*, 2:345–389, 1998.

Prakash M Nadkarni, Lucila Ohno-Machado, and Wendy W Chapman. Natural language processing: an introduction. *Journal of the American Medical Informatics Association*, 18(5):544–551, 2011.

Michael Petrolini, Stefano Cagnoni, and Monica Mordonini. Automatic detection of sensitive data using transformer- based classifiers. *Future Internet*, 14:228 – 228, 2022.

Krithika Ramesh, Sunayana Sitaram, and Monojit Choudhury. Fairness in language models beyond english: Gaps and challenges. *arXiv preprint arXiv:2302.12578v2*, February 2023.

Alice Richardson. Logistic regression: A self-learning text, third edition by david g. kleinbaum, mitchel klein. *International Statistical Review*, 79:296–296, 08 2011.

Alexander Rietzler, Sebastian Stabinger, Paul Opitz, and Stefan Engl. Adapt or get left behind: Domain adaptation through bert language model finetuning for aspect-target sentiment classification. *arXiv preprint arXiv:1908.11860*, 2019.

Lior Rokach and Oded Maimon. Top-down induction of decision trees classifiers - a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 35(4):476–487, 2005.

Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*, pages 808–809 and 497–499 and 869. Prentice Hall, 3 edition, 2010.

Manya Sleeper, Justin Cranshaw, Patrick Gage Kelley, Blase Ur, Alessandro Acquisti, Lorrie Faith Cranor, and Norman Sadeh. " i read my twitter the next morning and was astonished" a conversational perspective on twitter regrets. pages 3277–3286, 2013.

Shan Suthaharan and Shan Suthaharan. Support vector machine. *Machine learning models and algorithms for big data classification: thinking with examples for effective learning*, page 207, 2016.

Marie Andreassen Svanes and Tora Seim Gunstad. Detecting and grading hateful messages in the norwegian language. Master's thesis, Dept. of Computer Science, Norwegian University of Science and Technology, Trondheim, Norway, June 2020. URL https://ntnuopen.ntnu.no/ntnu-xmlui/handle/11250/2777836.

Philip H Swain and Hans Hauska. The decision tree classifier: Design and potential. *IEEE Transactions on Geoscience Electronics*, 15(3):142–147, 1977.

Welderufael B Tesfay, Jetzabel M Serna, and Sebastian Pape. Challenges in detecting privacy revealing information in unstructured text. In *PrivOn@ ISWC*, 2016.

Welderufael Berhane Tesfay, Jetzabel M. Serna, and Kai Rannenberg. Privacybot: Detecting privacy sensitive information in unstructured texts. *2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS)*, pages 53–60, 2019.

Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *ArXiv*, abs/1706.03762:1–15, June 2017.

S. D. Warren and L. D. Brandeis. “the right to privacy”. *Harvard Law Review*, 4(5): 193–220, 1980.

Alan F. Westin. “privacy and freedomy”. *Wash. Lee L. Rev.*, 25(1):166–167, 1968.

Yige Xu, Xipeng Qiu, Ligao Zhou, and Xuanjing Huang. Improving bert fine-tuning via self-ensemble and self-distillation. *arXiv preprint arXiv:2002.10345*, 2020.

Xue Ying. An overview of overfitting and its solutions. *Journal of physics: Conference series*, 1168:022022, February 2019.

Shihong Yue, Ping Li, and Peiyi Hao. Svm classification: Its contents and challenges. *Applied Mathematics-A Journal of Chinese Universities*, 18(3):332–342, 2003.

Leila Zahedi, Farid Ghareh Mohammadi, Shabnam Rezapour, Matthew W Ohland, and M Hadi Amini. Search algorithms for automated hyper-parameter tuning. *arXiv preprint arXiv:2104.14677*, 2021.

A. One Entry in the Annotated Data Set

Table A.1.: A table presenting one entry in the annotated data set. The column titles have been shortened to fit the page.

Annotation Data set												
selftext	subreddit	unintelligible	non-sensitive	unknown	private-sensitive	PII	locate	health	family	economy	political	criminal
Jeg fikk 5000 kr i bot for å sjekke hvor mye batteri som var igjen på mobilen da jeg sto i (lang) kø og ventet på grønt lys for n'te gang i morgenrushet.	norge	0	0	0	1	0	0	0	0	0	0	1

B. Training and Validation Accuracy of Parameter Combinations for Multinomial LR

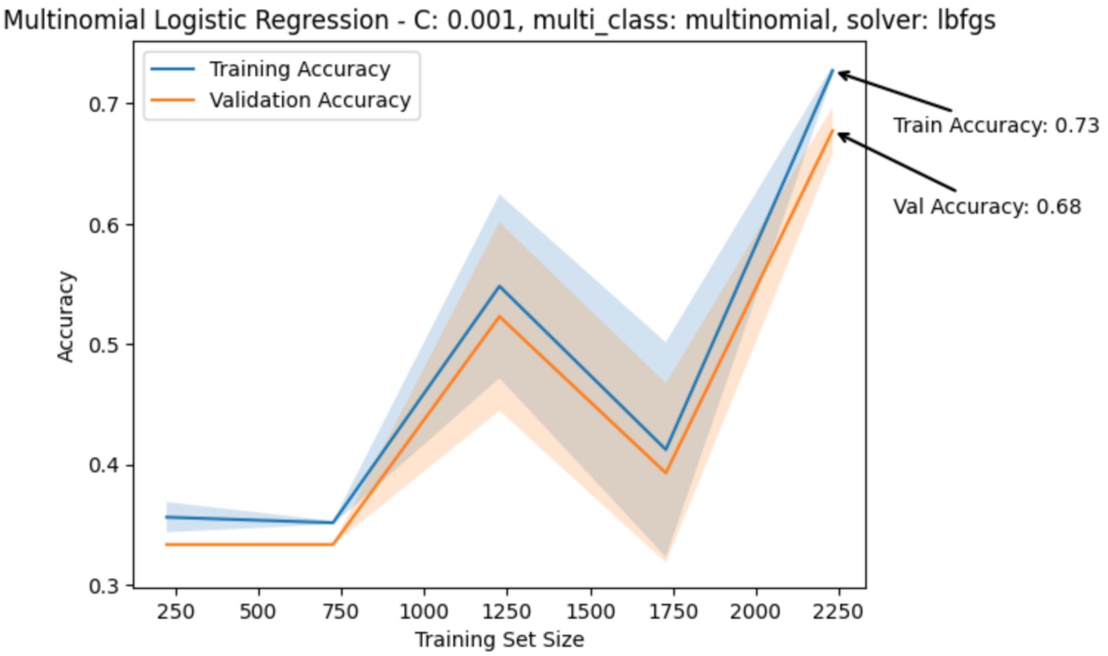


Figure B.1.: Multinomial LR: Training and validation accuracy with parameter combination: C:0.001, multi_class: multinomial, solver: lbgfs

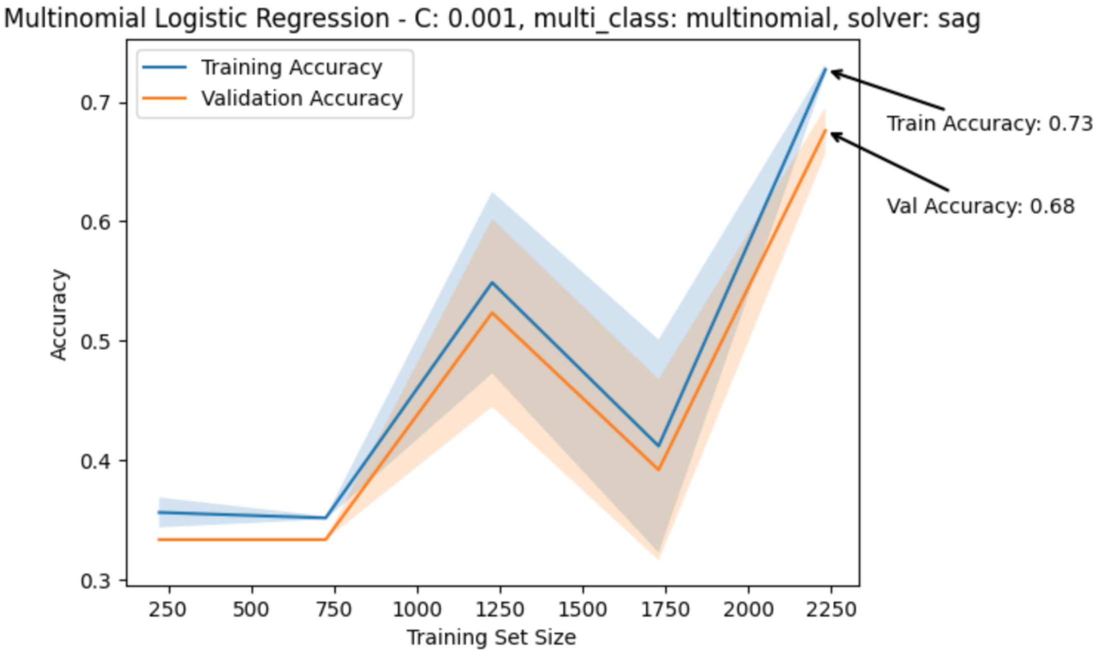


Figure B.2.: Multinomial LR: Training and validation accuracy with parameter combination: C:0.001, multi_class: multinomial, solver: sag

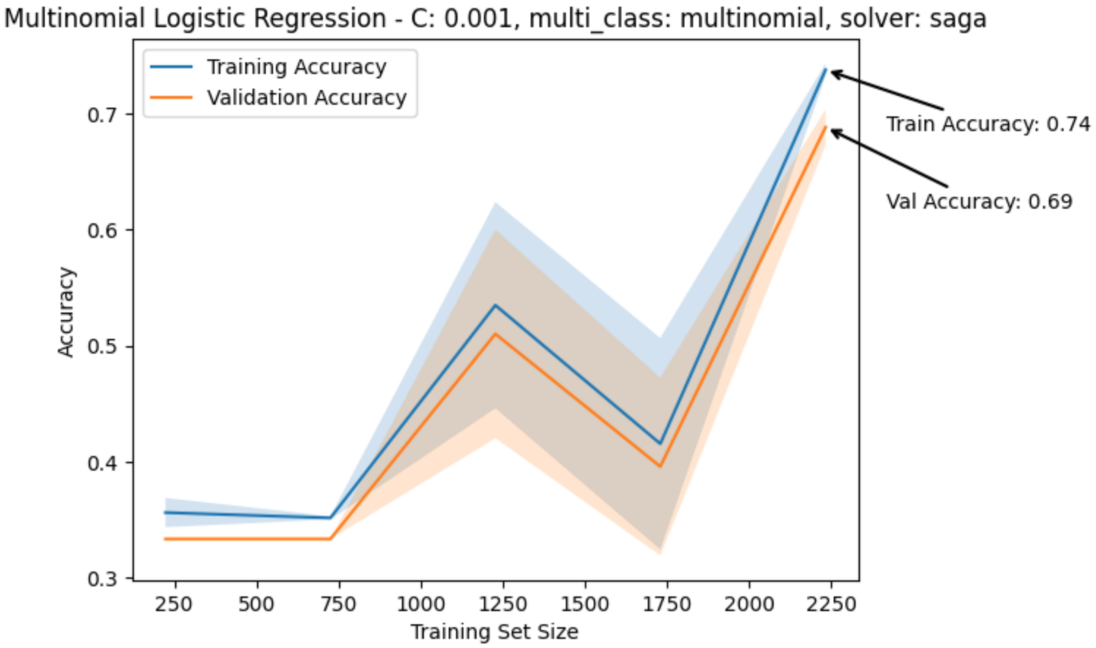


Figure B.3.: Multinomial LR: Training and validation accuracy with parameter combination: C:0.001, multi_class: multinomial, solver: saga

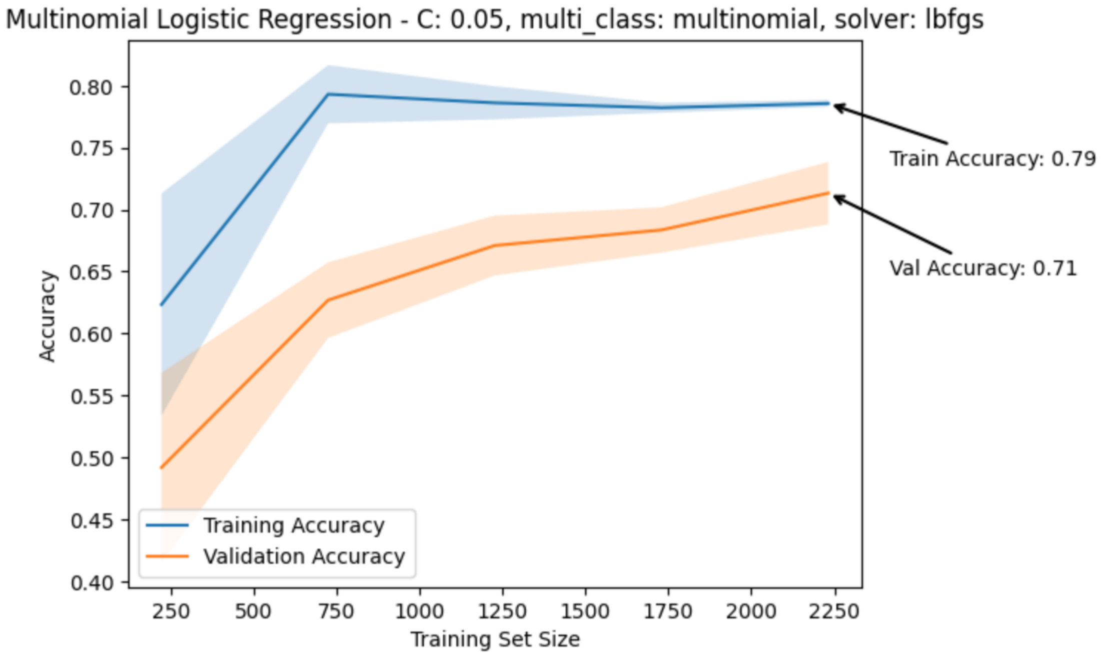


Figure B.4.: MultinomialL R: Training and validation accuracy with parameter combination: C:0.05, multi_class: multinomial, solver: lbgfs

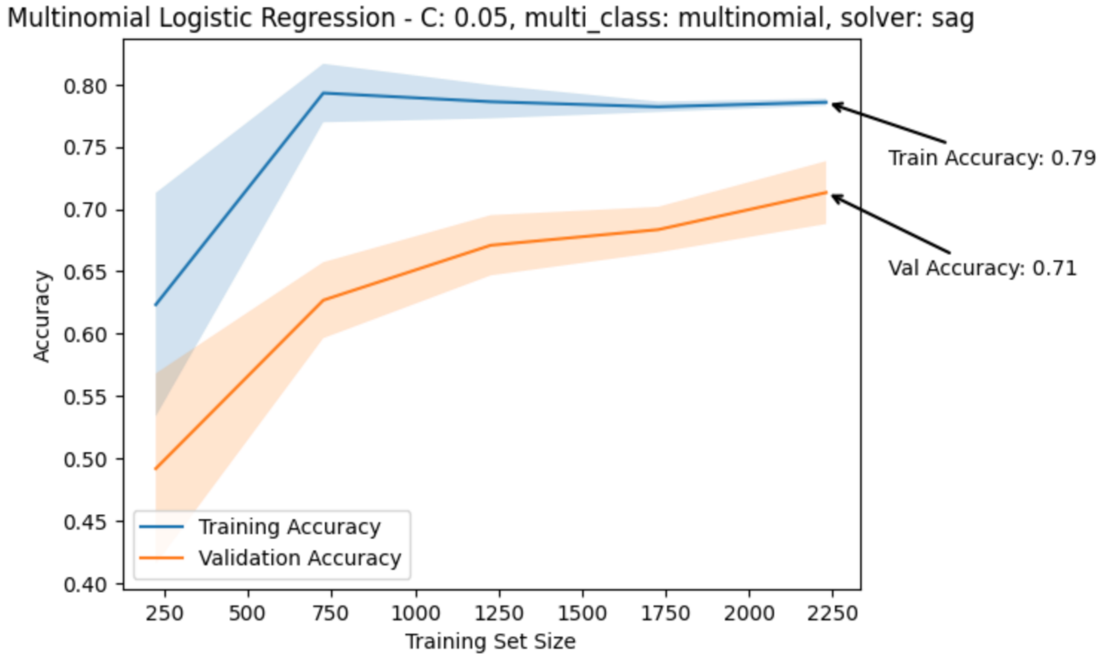


Figure B.5.: Multinomial LR: Training and validation accuracy with parameter combination: C:0.05, multi_class: multinomial, solver: sag

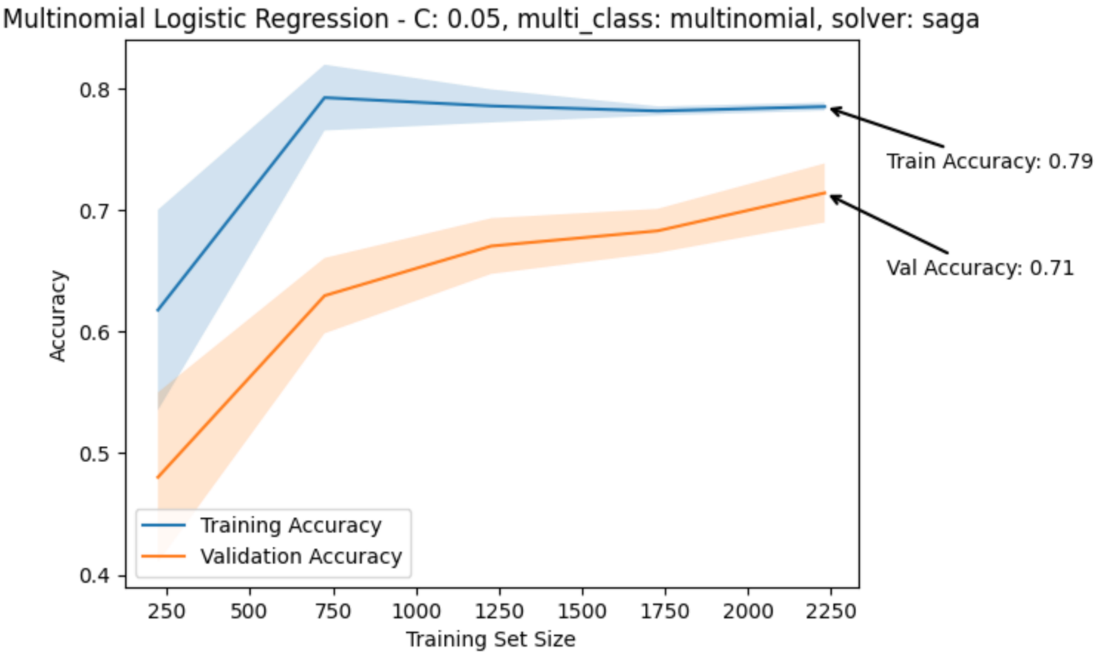


Figure B.6.: Multinomial LR: Training and validation accuracy with parameter combination: C:0.05, multi_class: multinomial, solver: saga

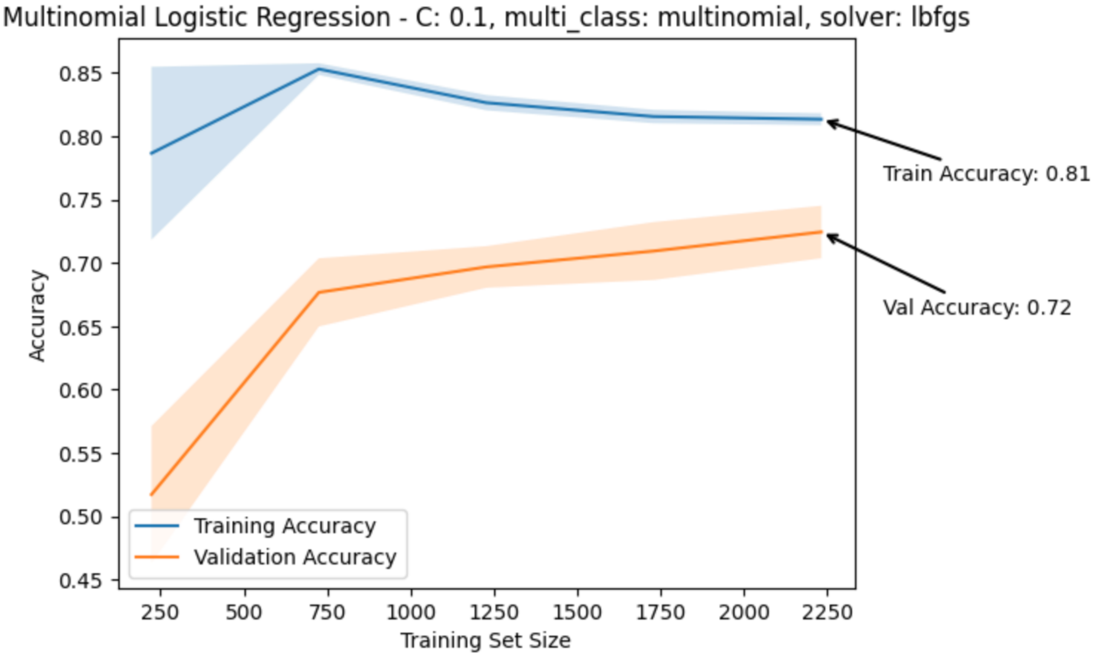


Figure B.7.: Multinomial LR: Training and validation accuracy with parameter combination: C:0.1, multi_class: multinomial, solver: lbfgs

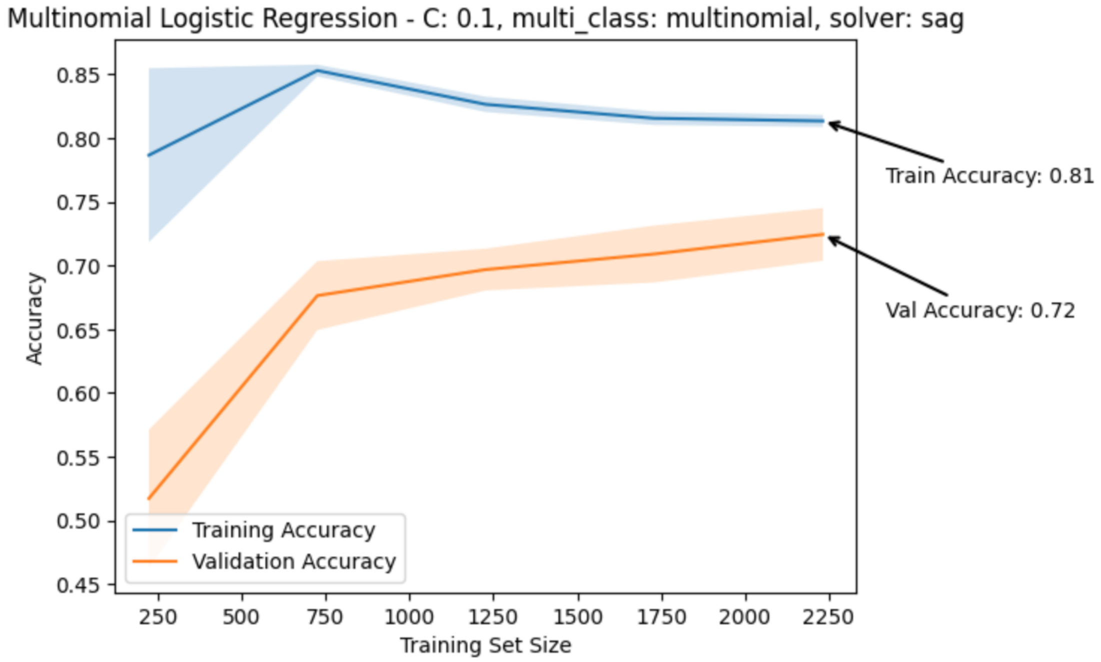


Figure B.8.: Multinomial LR: Training and validation accuracy with parameter combination: C:0.1, multi_class: multinomial, solver: sag

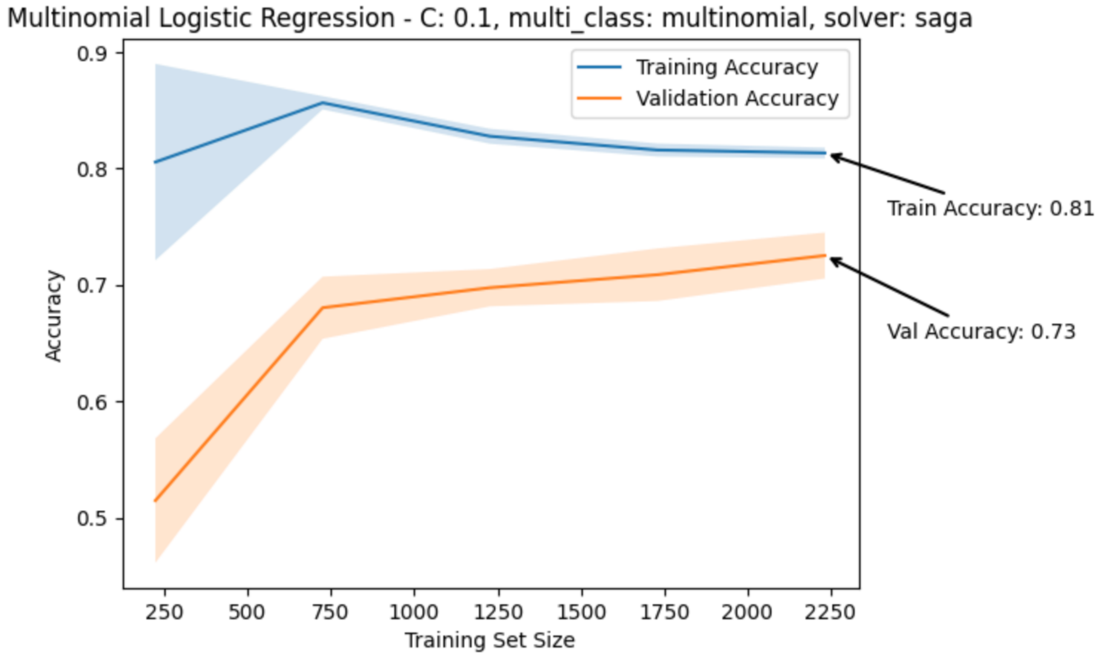


Figure B.9.: Multinomial LR: Training and validation accuracy with parameter combination: C:0.1, multi_class: multinomial, solver: saga

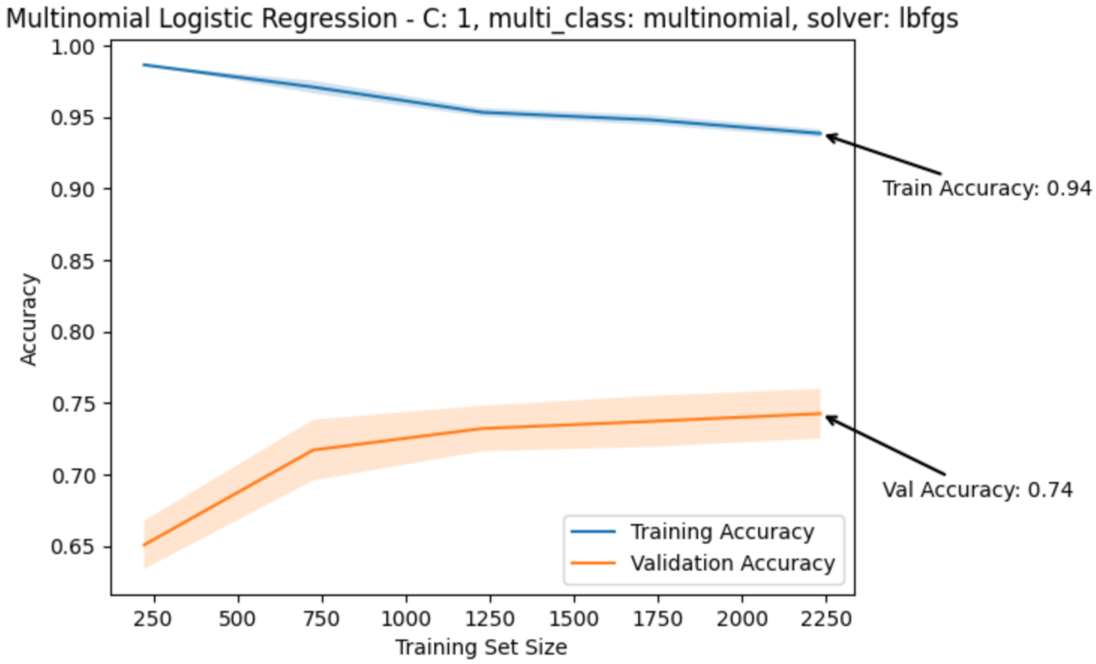


Figure B.10.: Multinomial LR: Training and validation accuracy with parameter combination: C:1, multi_class: multinomial, solver: lbfgs

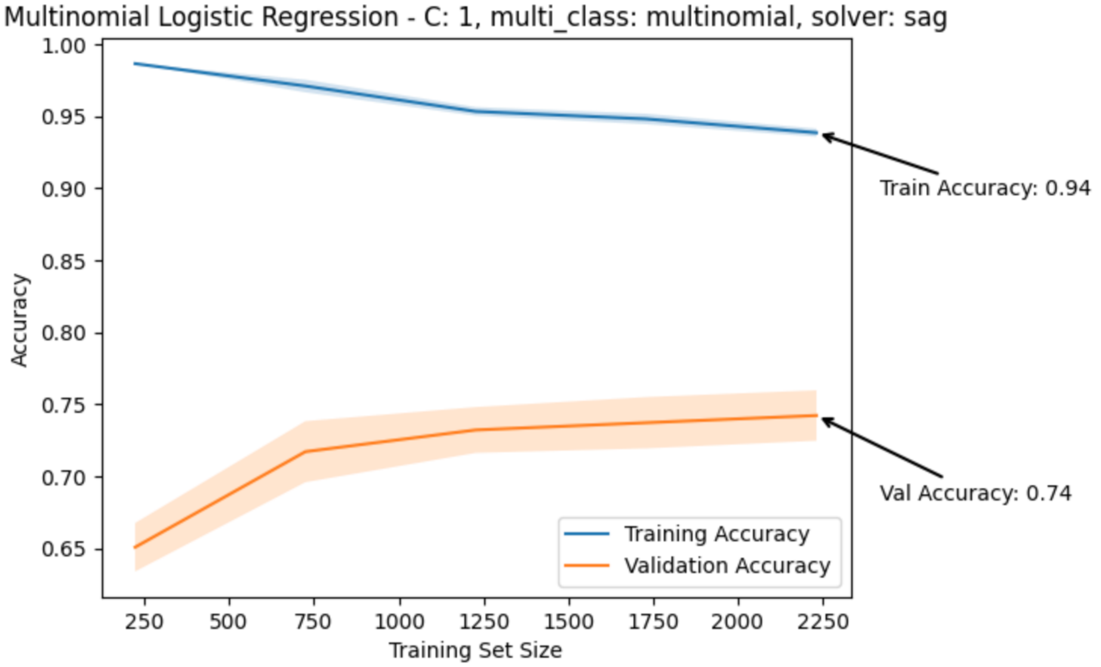


Figure B.11.: Multinomial LR: Training and validation accuracy with parameter combination: C:1, multi_class: multinomial, solver: sag

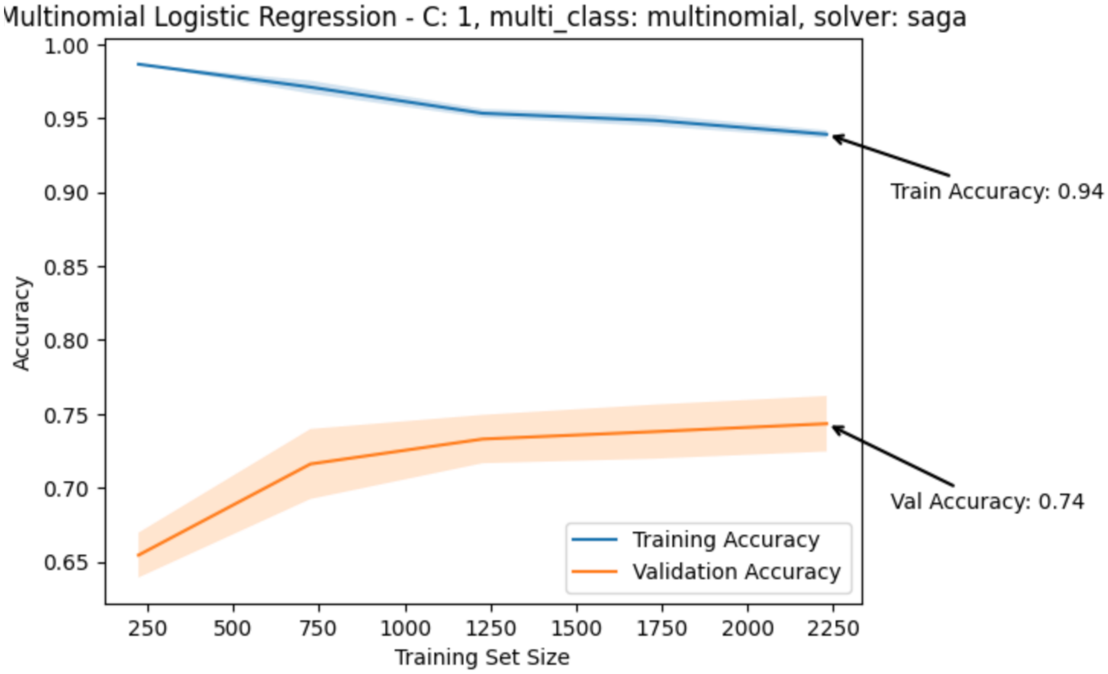


Figure B.12.: Multinomial LR: Training and validation accuracy with parameter combination: C:1, multi_class: multinomial, solver: saga

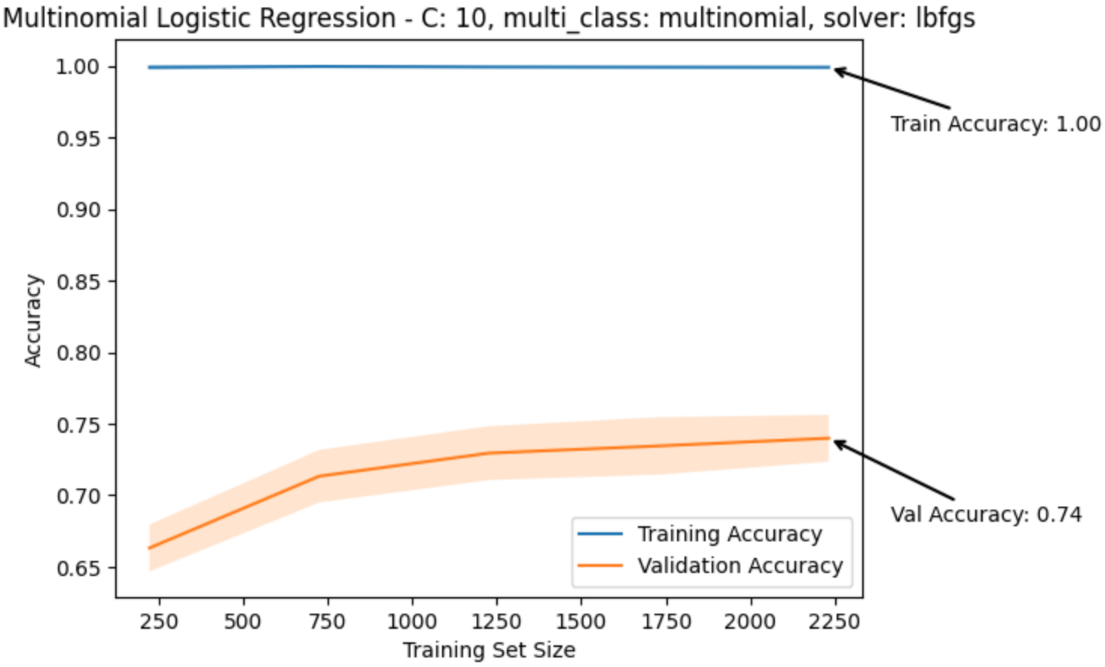


Figure B.13.: Multinomial LR: Training and validation accuracy with parameter combination: C:10, multi_class: multinomial, solver: lbfgs

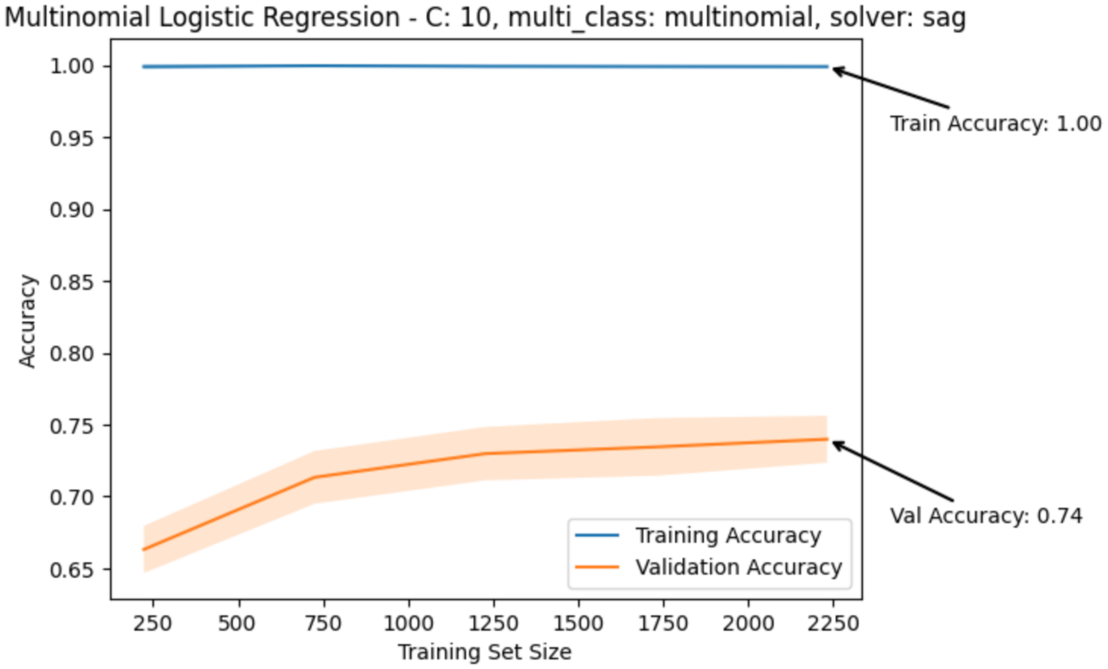


Figure B.14.: Multinomial LR: Training and validation accuracy with parameter combination: C:10, multi_class: multinomial, solver: sag

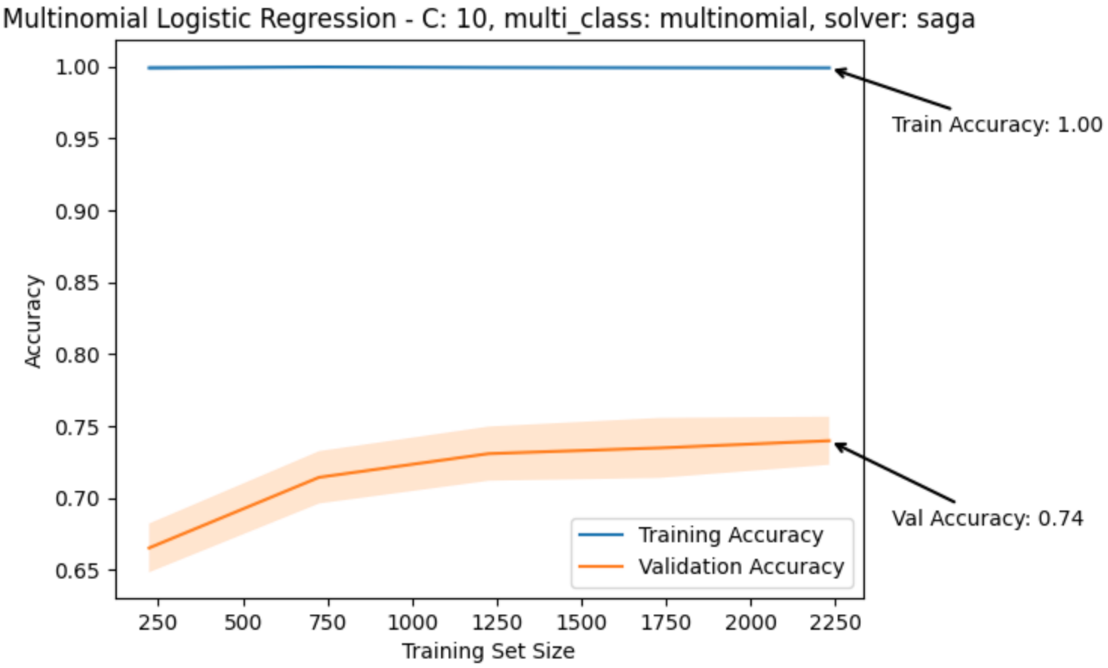


Figure B.15.: Multinomial LR: Training and validation accuracy with parameter combination: C:10, multi_class: multinomial, solver: saga

C. Training and Validation Accuracy of Parameter Combinations for Multinomial NB

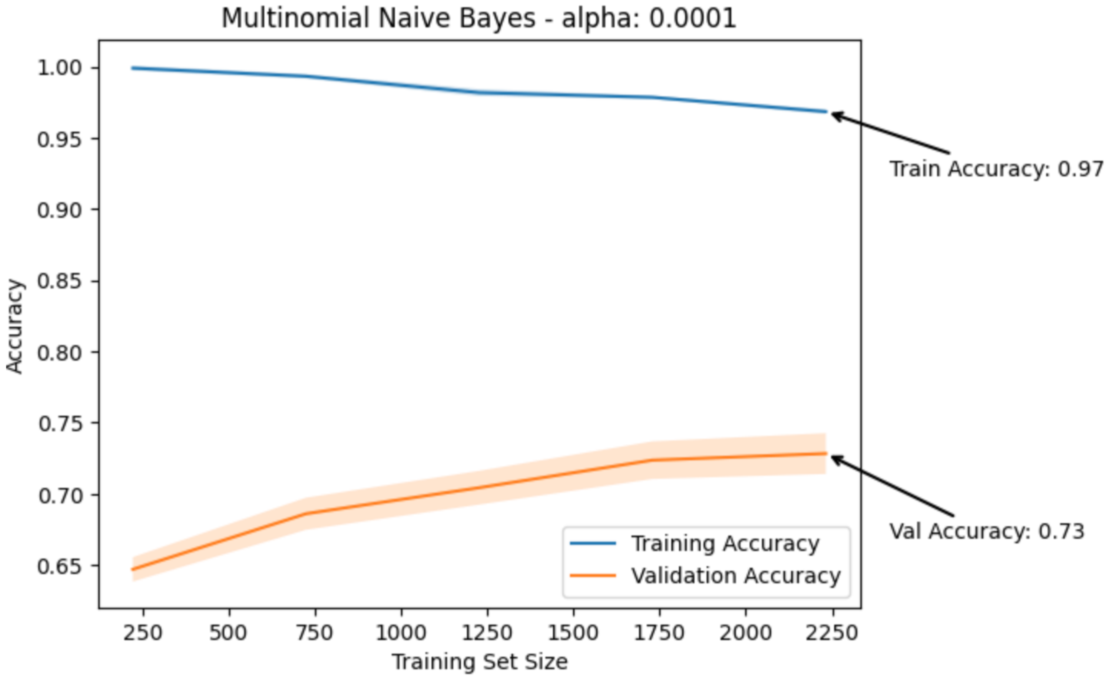


Figure C.1.: Multinomial NB: Training and validation accuracy with parameter combination: alpha: 0.0001

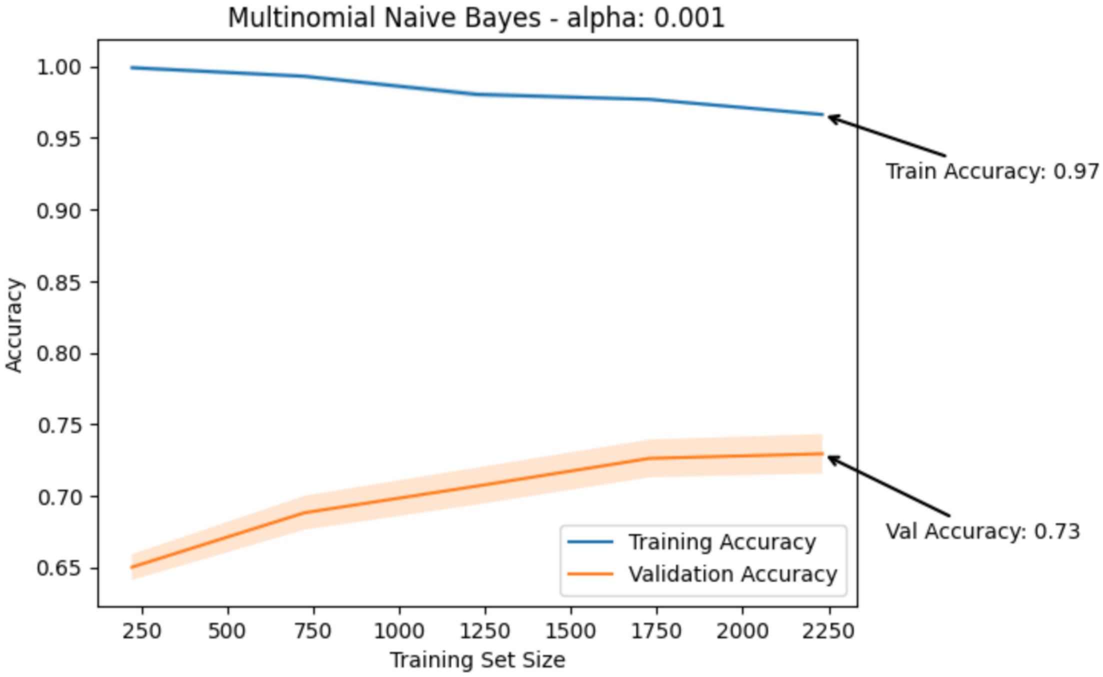


Figure C.2.: Multinomial NB: Training and validation accuracy with parameter combination: alpha: 0.001

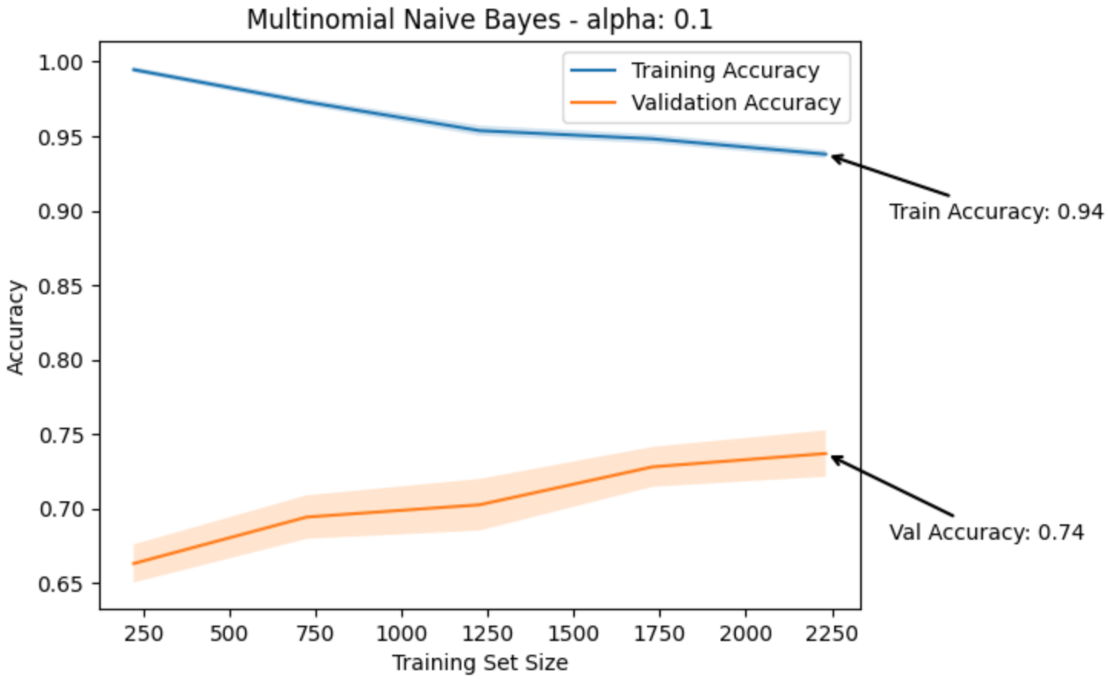


Figure C.3.: Multinomial NB: Training and validation accuracy with parameter combination: alpha: 0.1

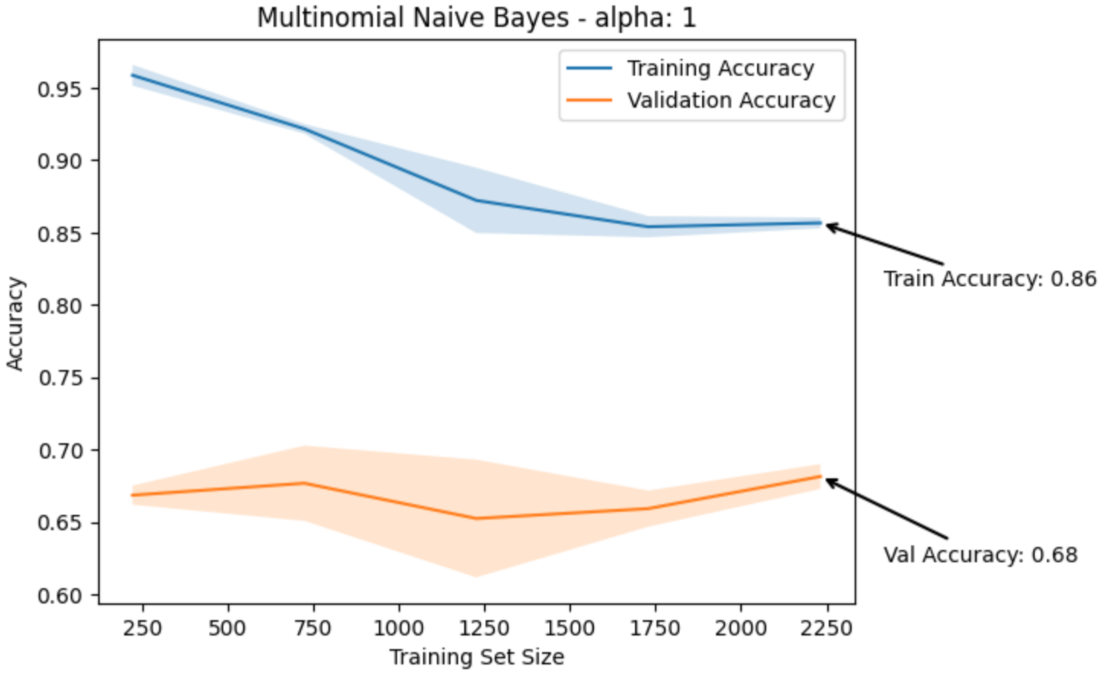


Figure C.4.: Multinomial NB: Training and validation accuracy with parameter combination: alpha: 1

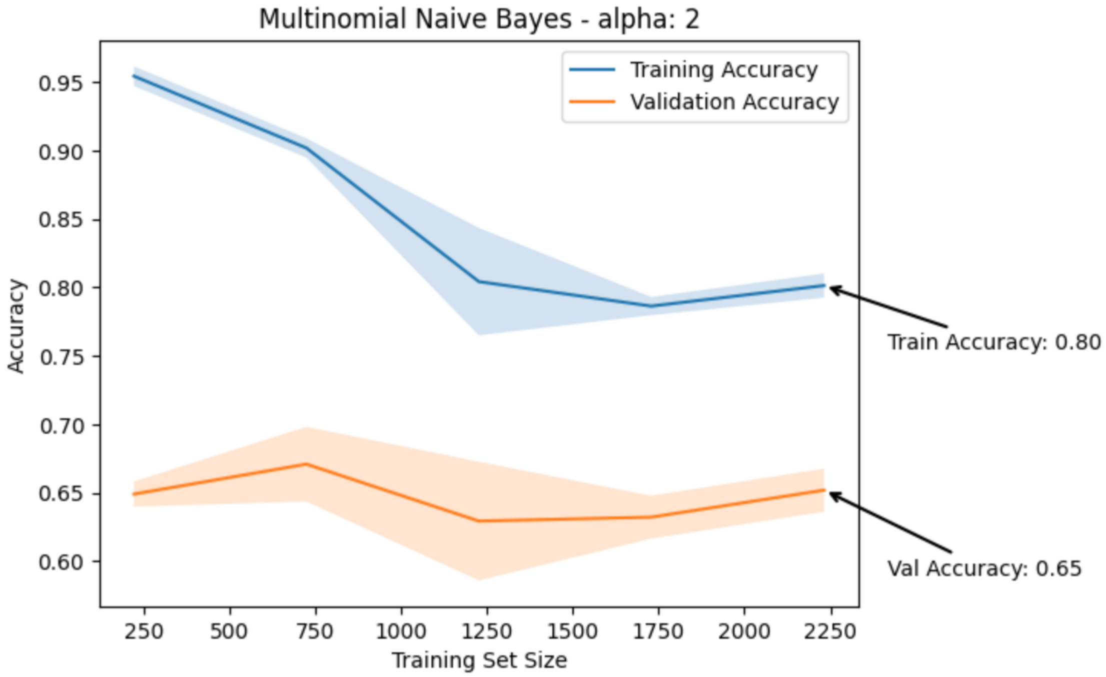


Figure C.5.: Multinomial NB: Training and validation accuracy with parameter combination: alpha: 2

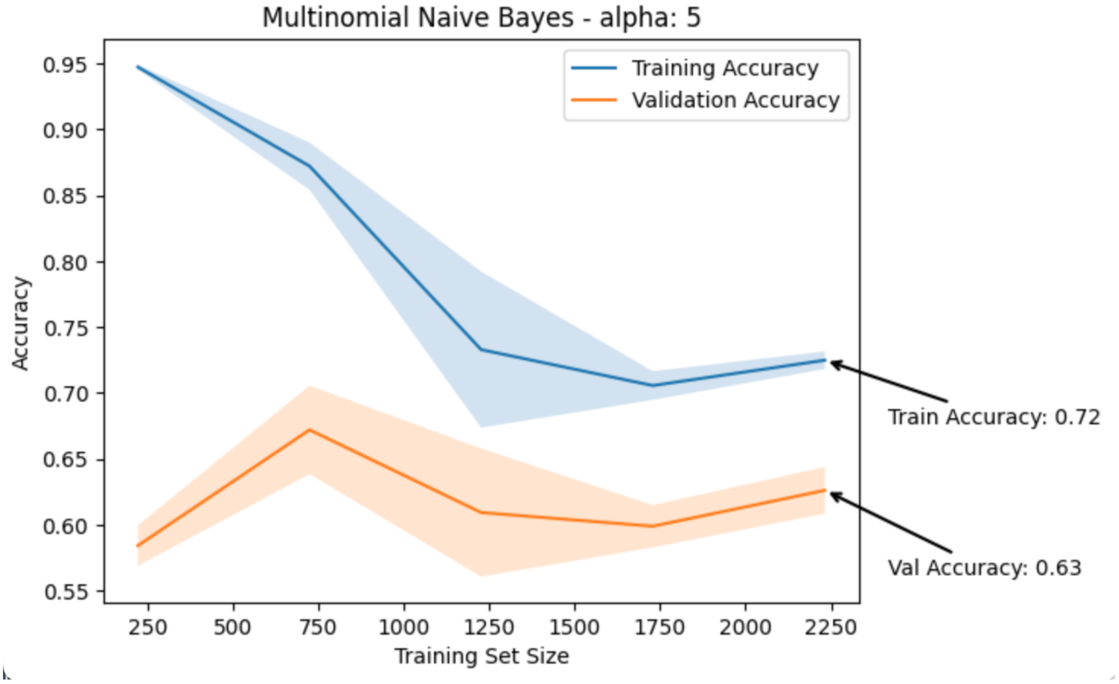


Figure C.6.: Multinomial NB: Training and validation accuracy with parameter combination: alpha: 5

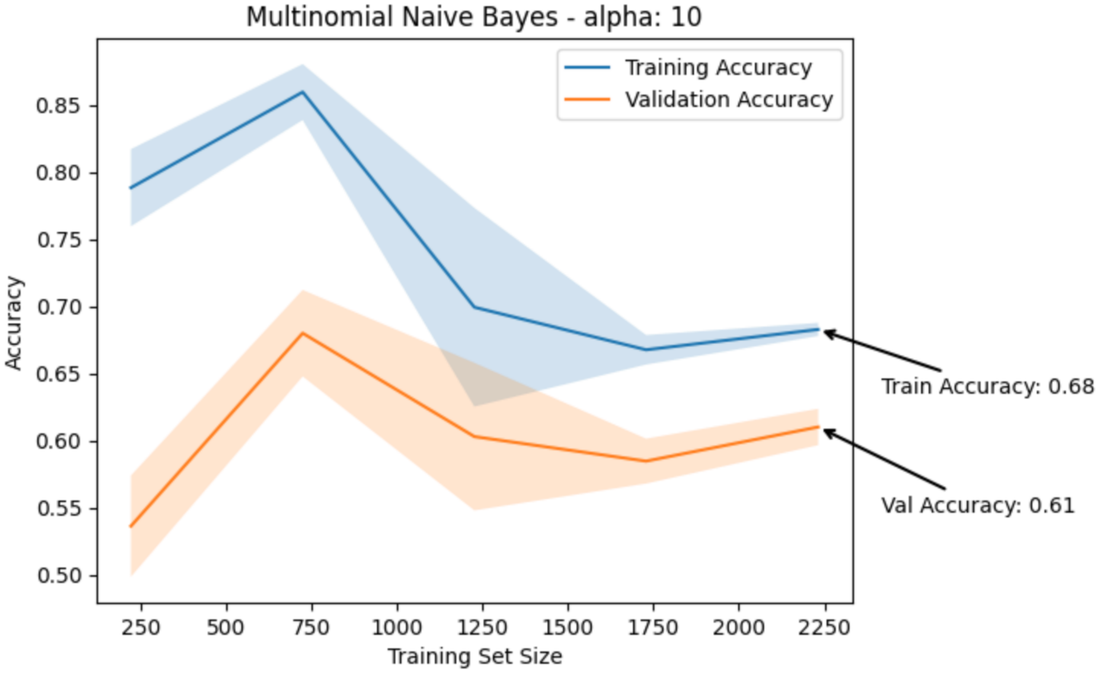


Figure C.7.: Multinomial NB: Training and validation accuracy with parameter combination: alpha: 10

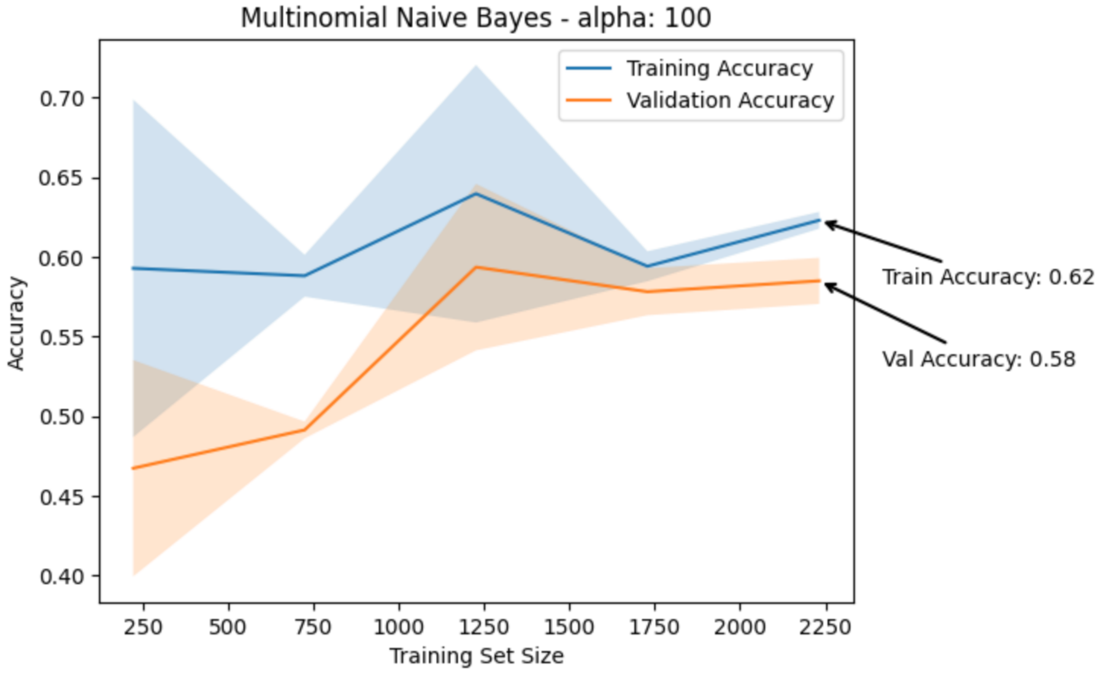


Figure C.8.: Multinomial NB: Training and validation accuracy with parameter combination: alpha: 100

D. Training and Validation Accuracy of with Parameter Combinations for Random Forest

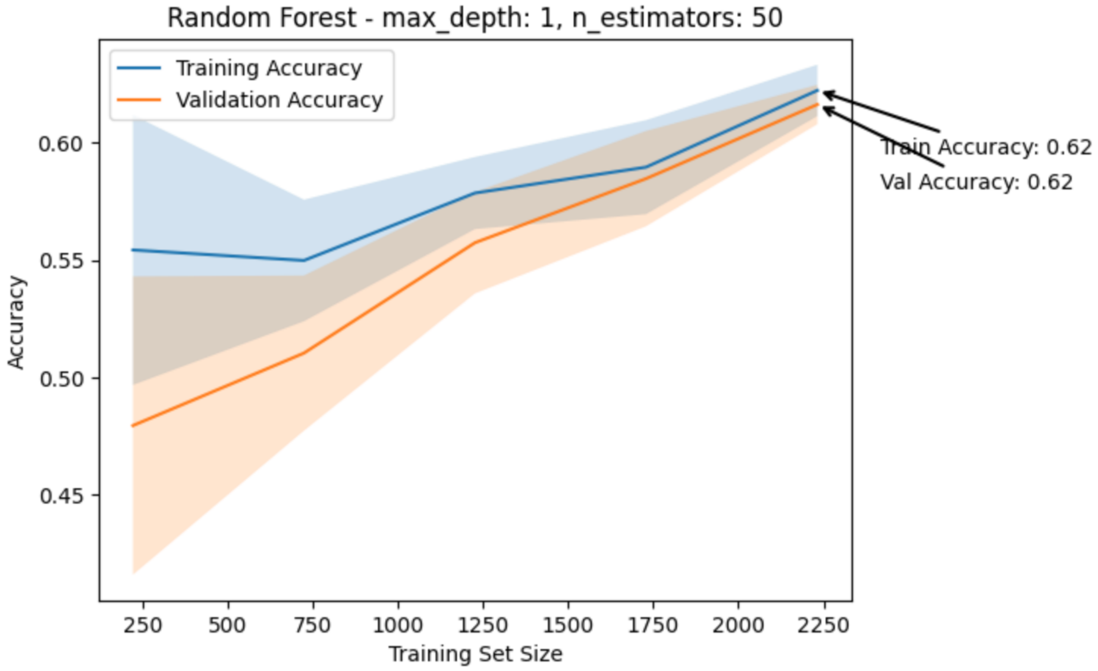


Figure D.1.: RF: Training and validation accuracy with parameter combination: max_-depth: 1 and n_estimators 50

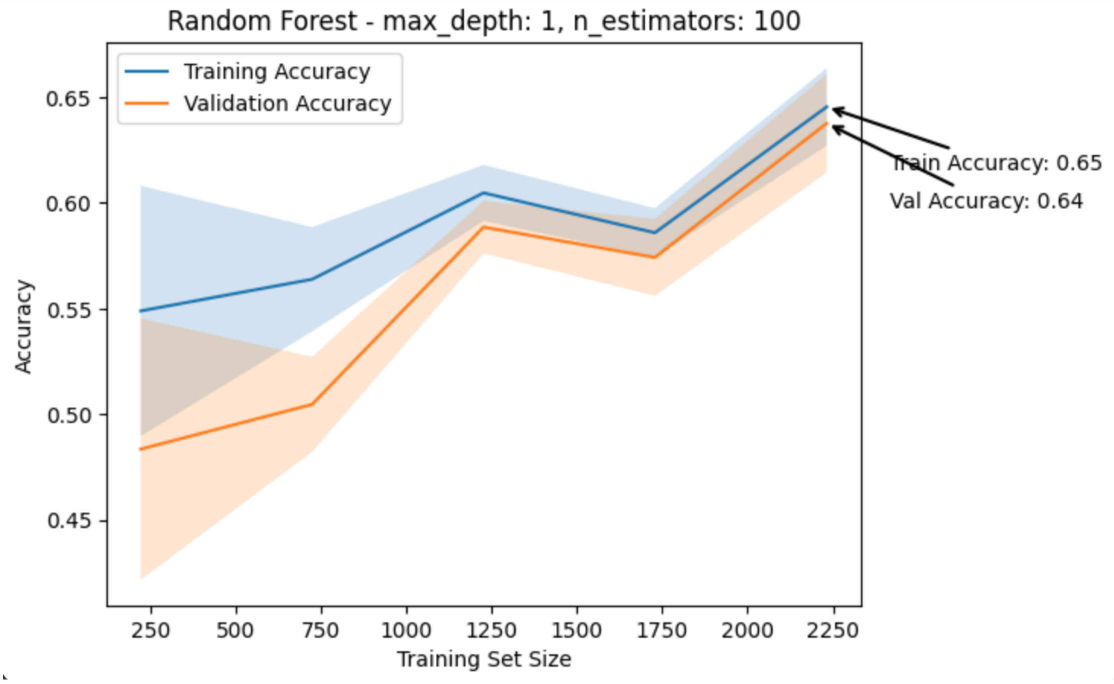


Figure D.2.: RF: Training and validation accuracy with parameter combination: max_-depth: 1 and n_estimators 100

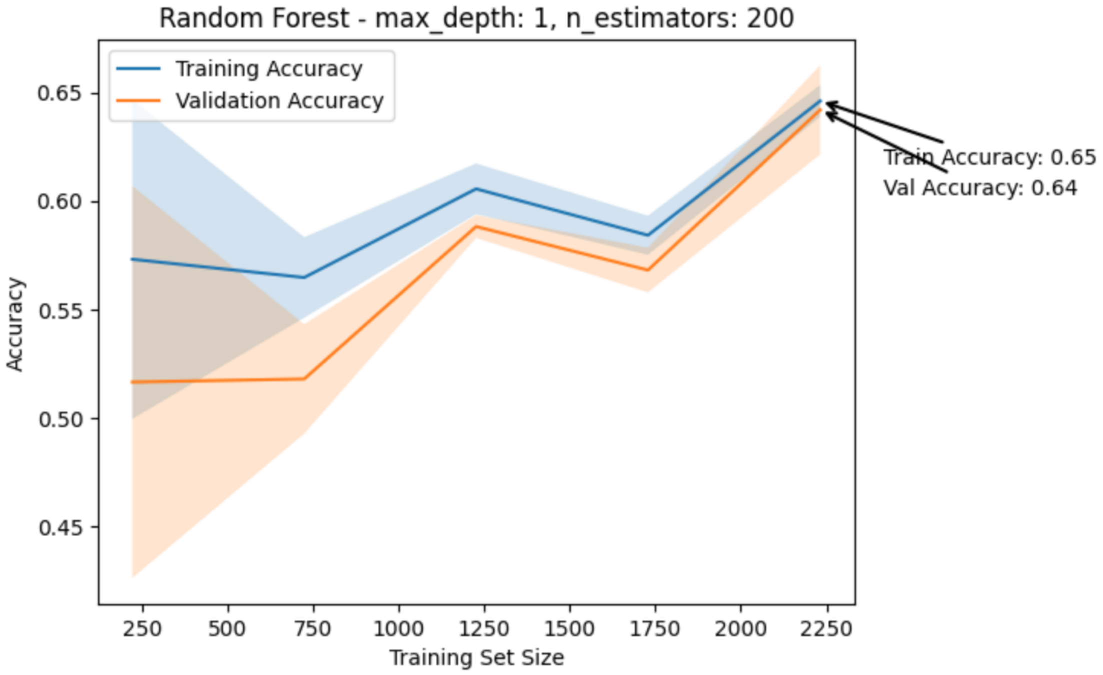


Figure D.3.: RF: Training and validation accuracy with parameter combination: max_depth: 1 and n_estimators 200

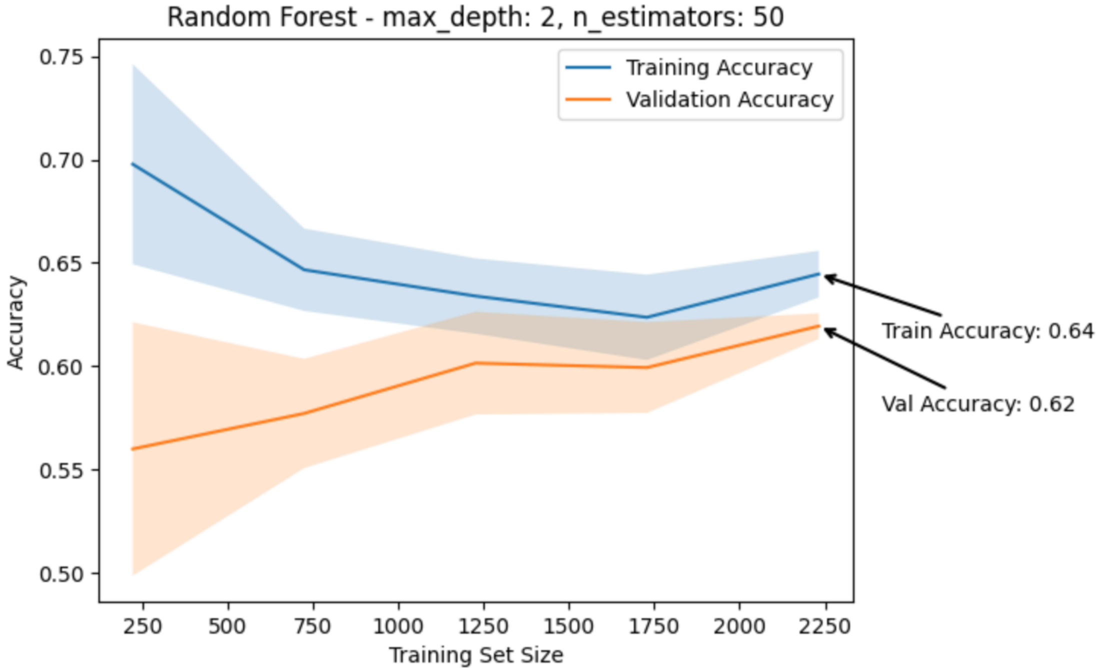


Figure D.4.: RF: Training and validation accuracy with parameter combination: max_depth: 2 and n_estimators 50

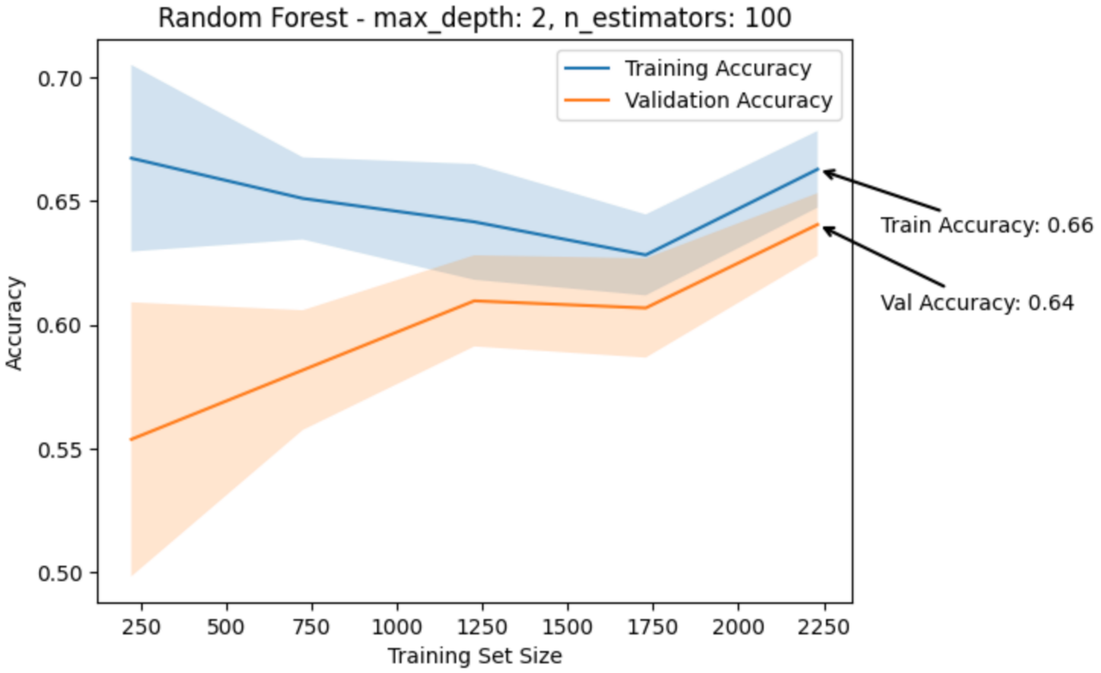


Figure D.5.: RF: Training and validation accuracy with parameter combination: max_-depth: 2 and n_estimators 100

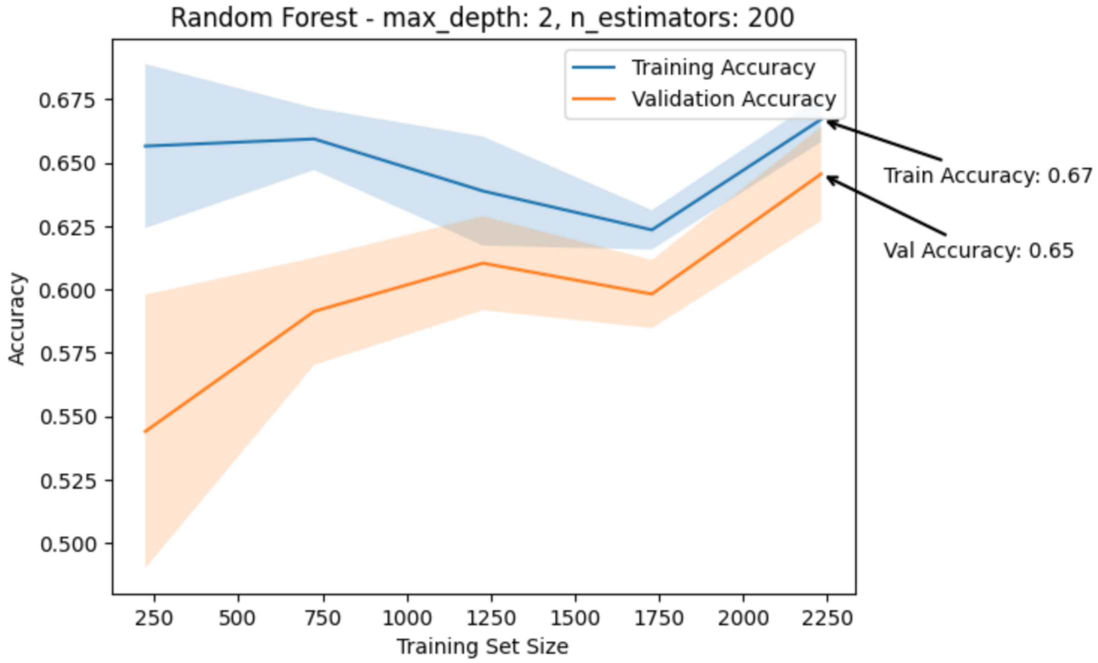


Figure D.6.: RF: Training and validation accuracy with parameter combination: max_-depth: 2 and n_estimators 200

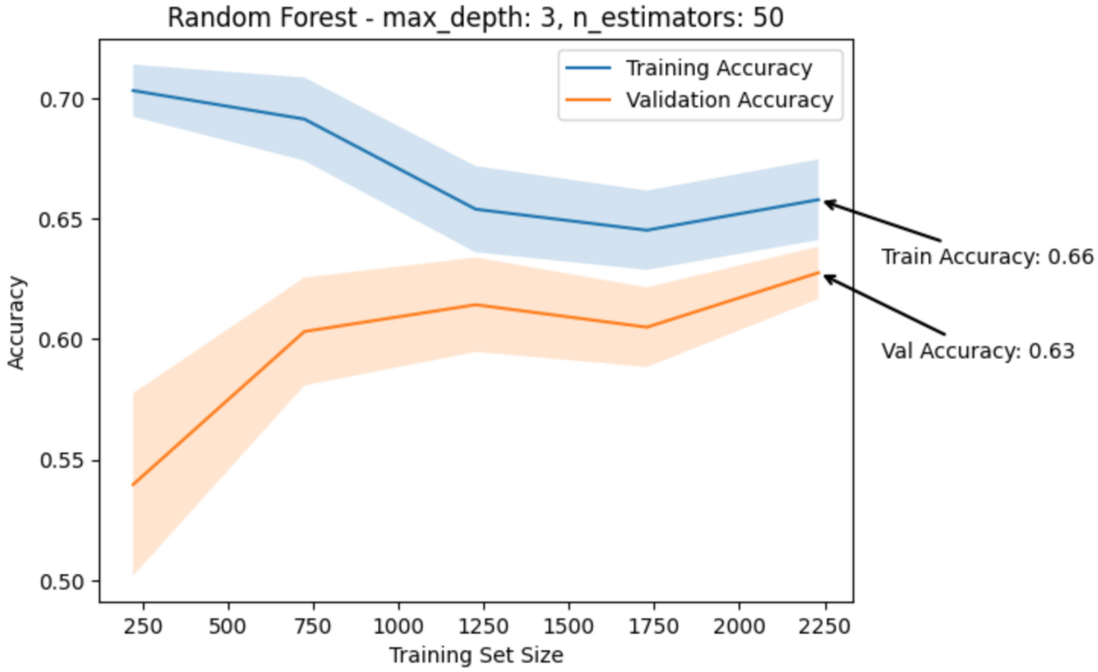


Figure D.7.: RF: Training and validation accuracy with parameter combination: max_depth: 3 and n_estimators 50

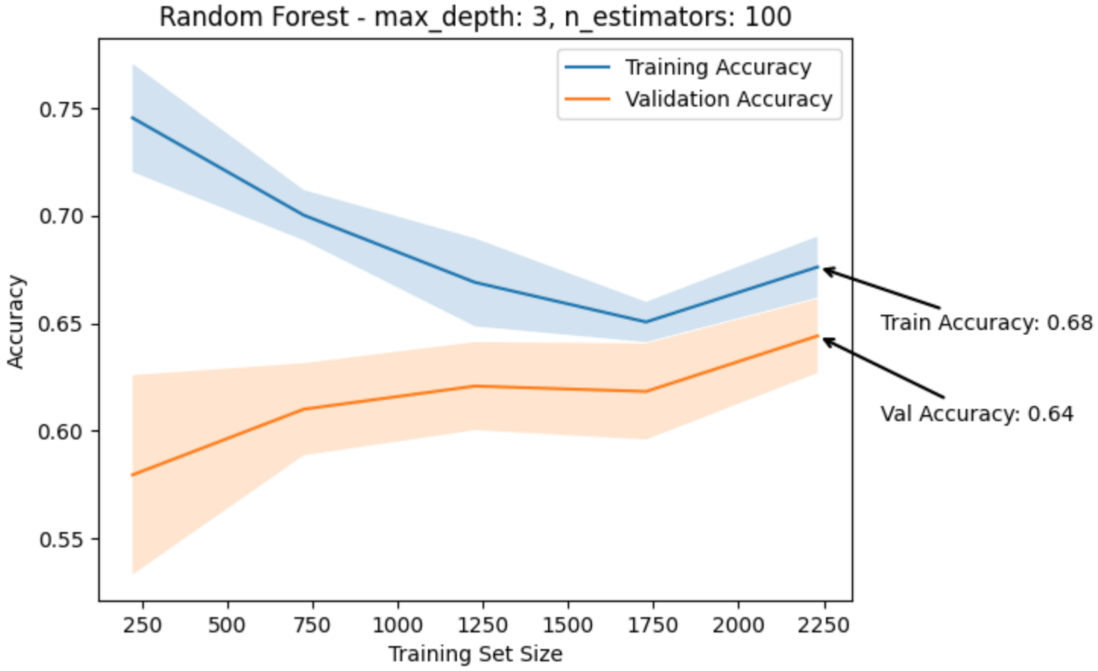


Figure D.8.: RF: Training and validation accuracy with parameter combination: max_depth: 3 and n_estimators 100

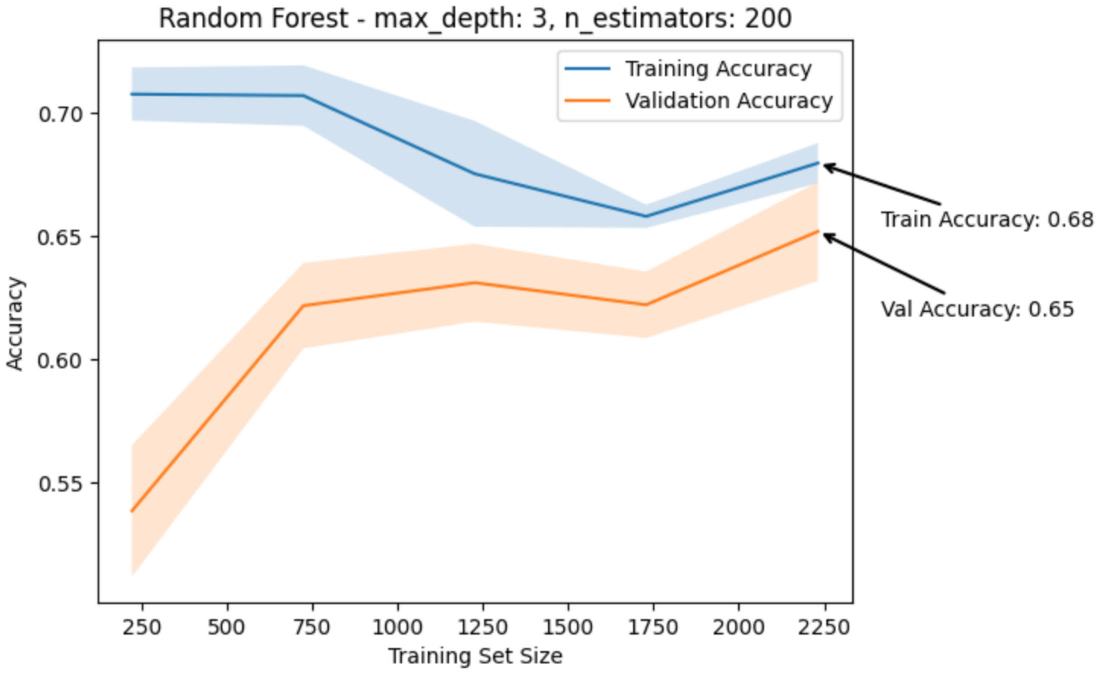


Figure D.9.: RF: Training and validation accuracy with parameter combination: max_-depth: 3 and n_estimators 200

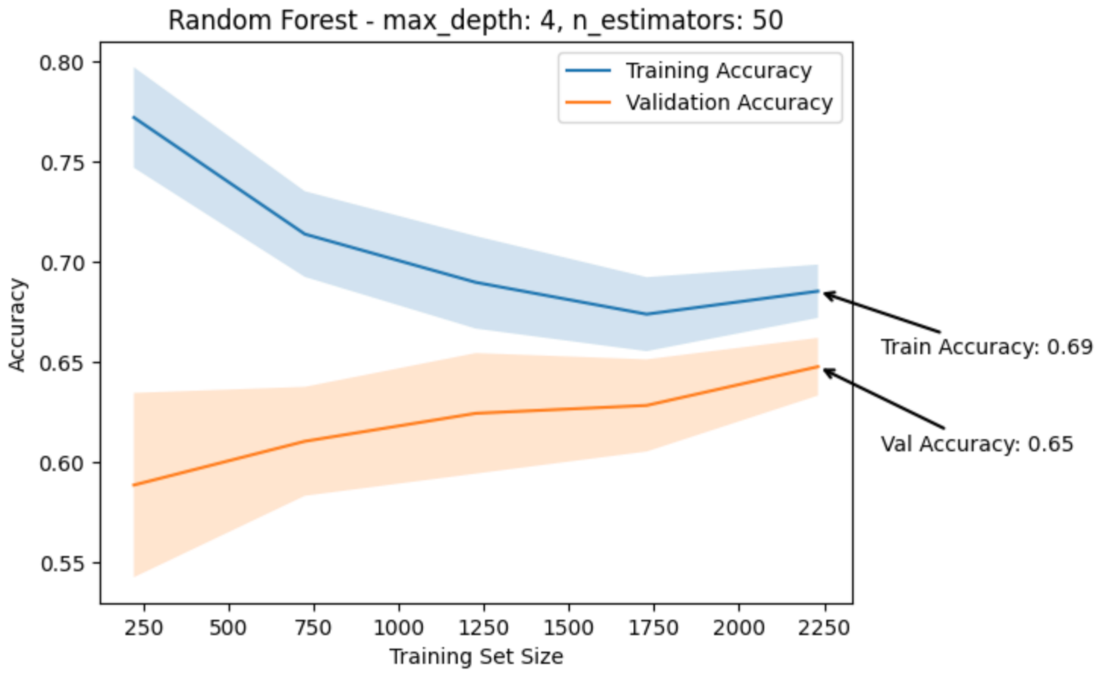


Figure D.10.: RF: Training and validation accuracy with parameter combination: max_-depth: 4 and n_estimators 50

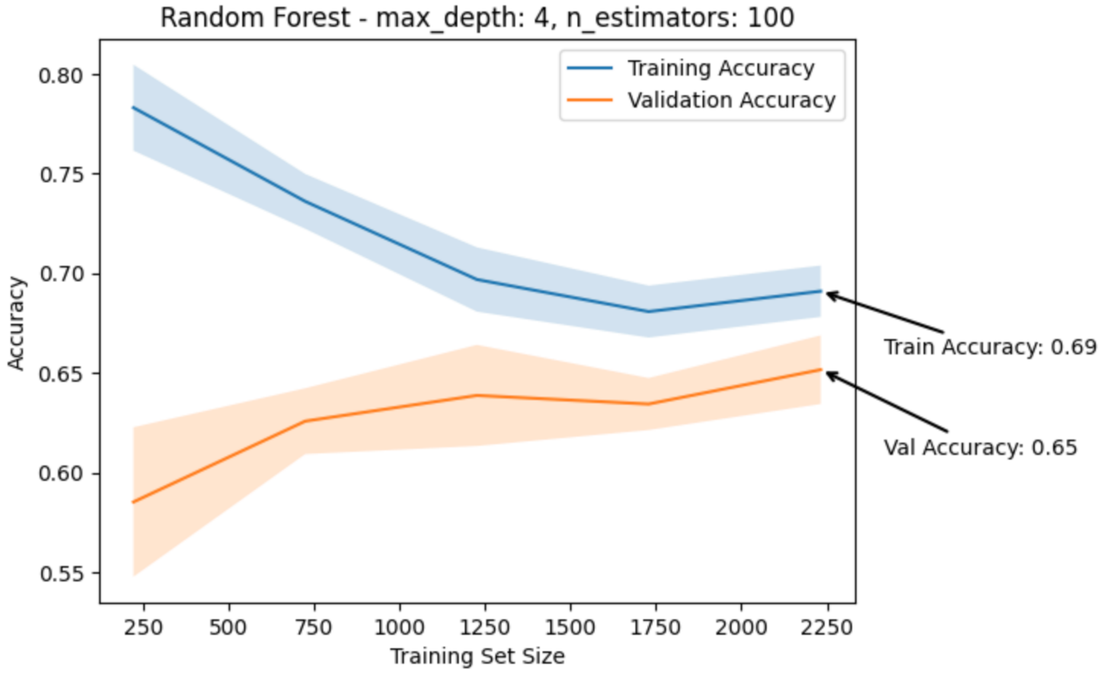


Figure D.11.: RF: Training and validation accuracy with parameter combination: max_-depth: 4 and n_estimators 100

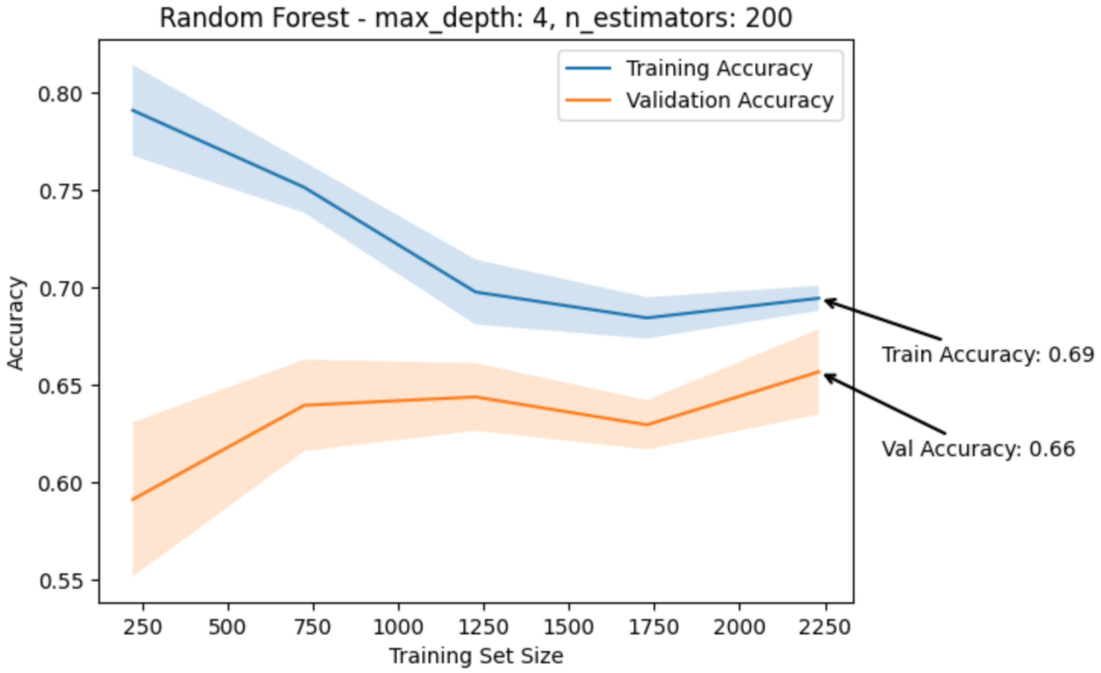


Figure D.12.: RF: Training and validation accuracy with parameter combination: max_-depth: 4 and n_estimators 200

E. Training and Validation Accuracy of Parameter Combinations for Linear SVM

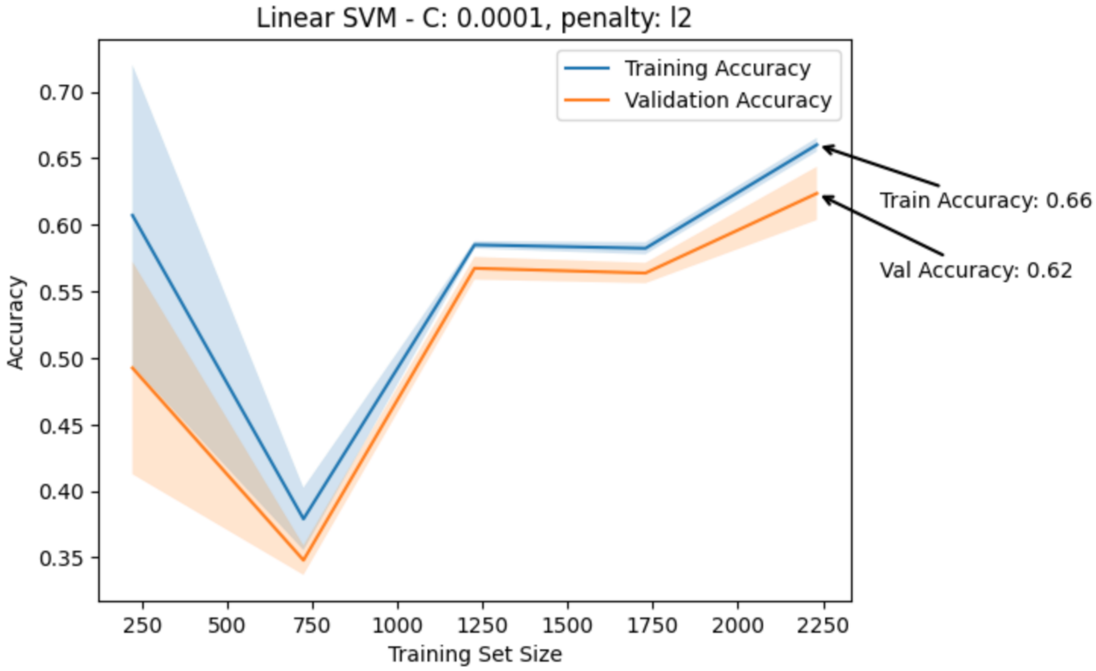


Figure E.1.: Linear SVM: Training and validation accuracy with parameter combination: C: 0.0001 and penalty: l2

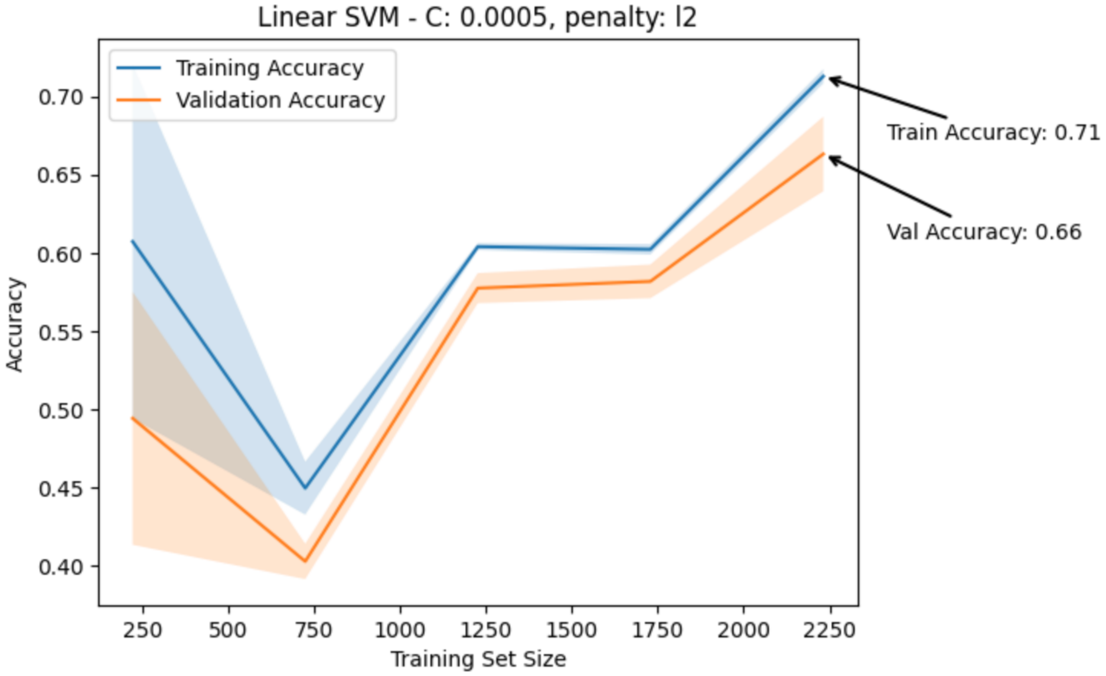


Figure E.2.: Linear SVM: Training and validation accuracy with parameter combination: C: 0.0005 and penalty: l2

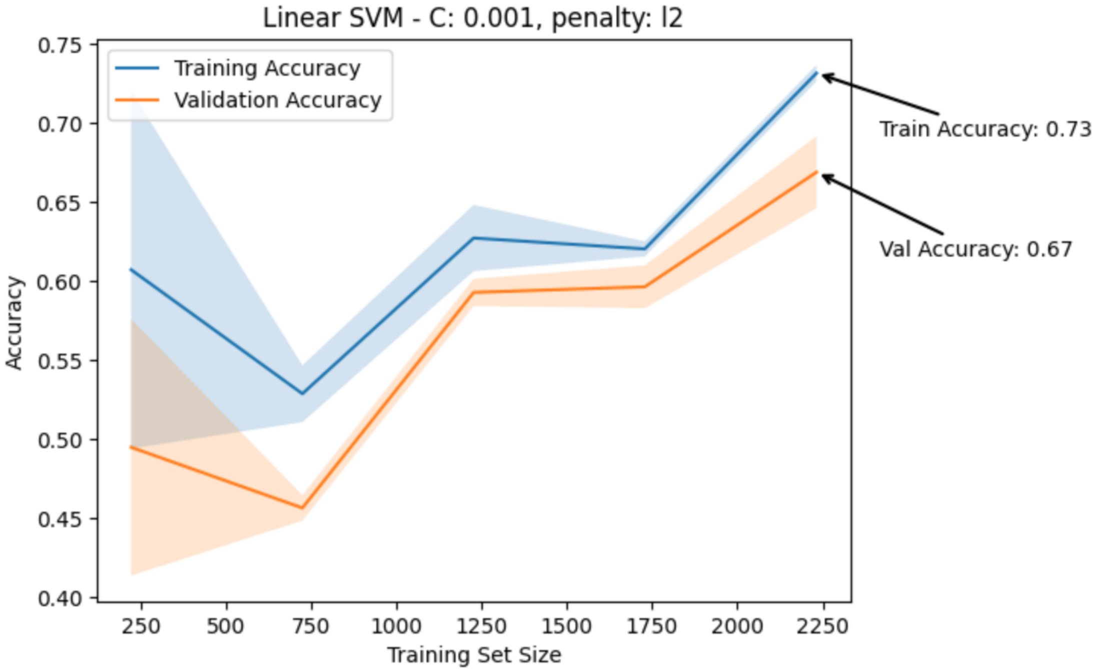


Figure E.3.: Linear SVM: Training and validation accuracy with parameter combination: C: 0.001 and penalty: l2

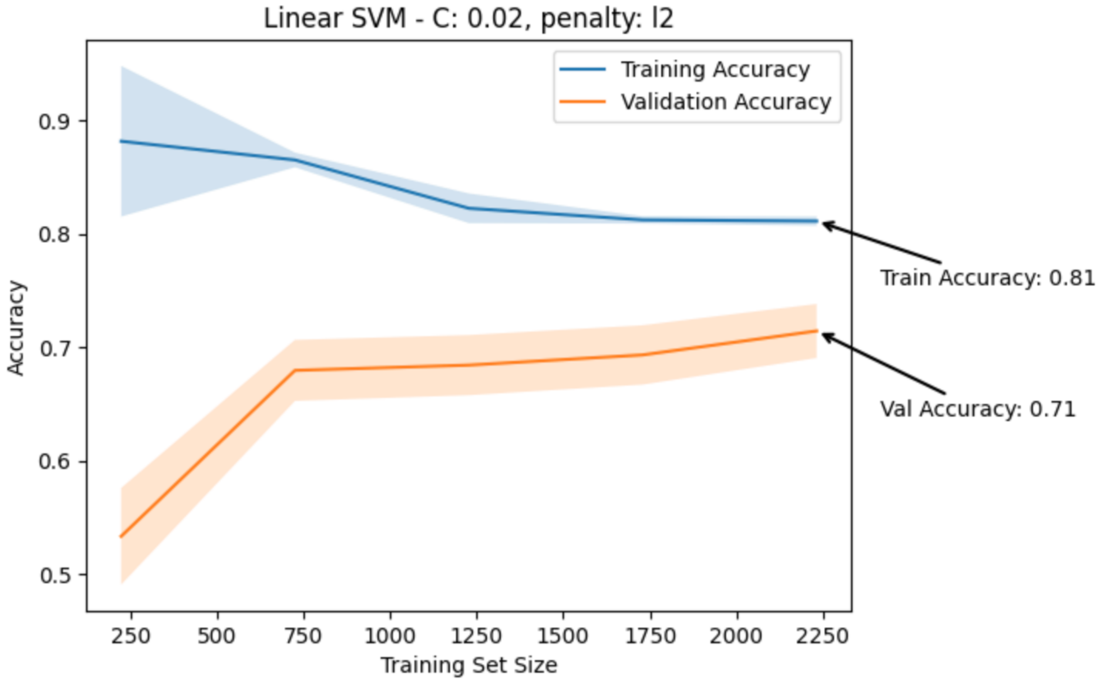


Figure E.4.: Linear SVM: Training and validation accuracy with parameter combination: C: 0.02 and penalty: l2

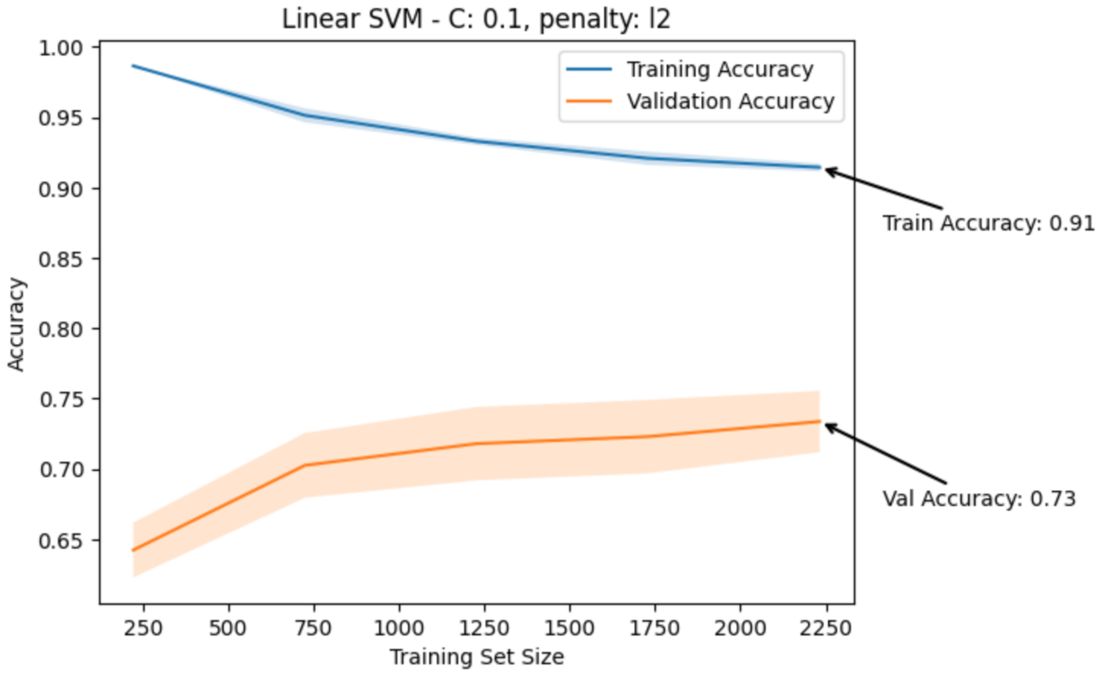


Figure E.5.: Linear SVM: Training and validation accuracy with parameter combination: C: 0.1 and penalty: l2

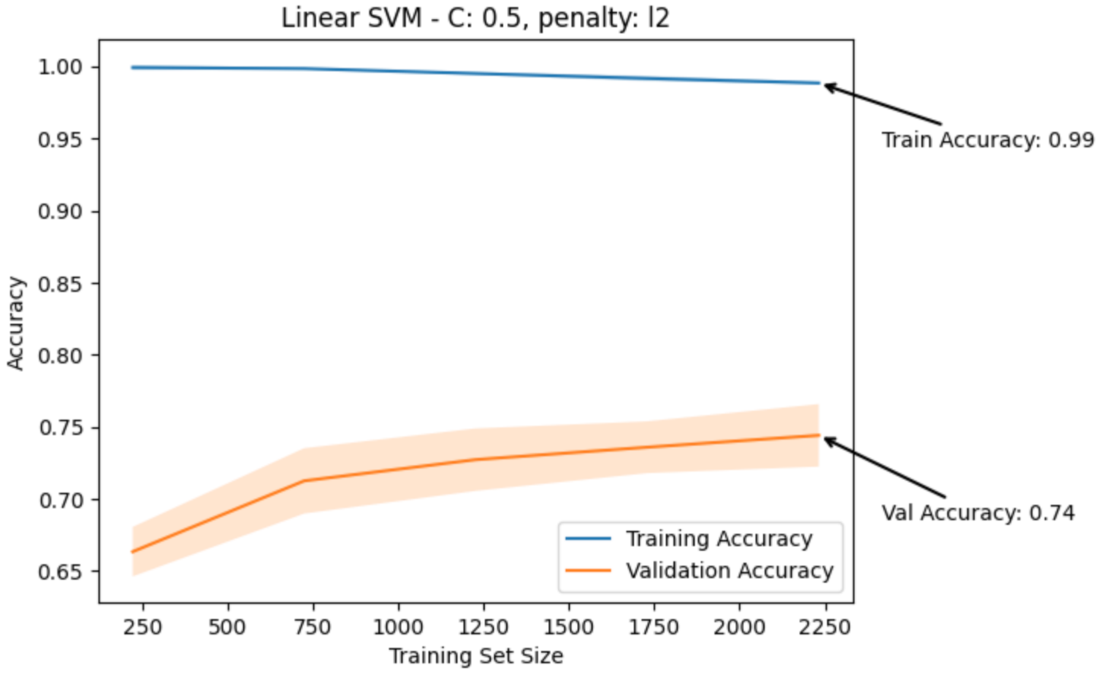


Figure E.6.: Linear SVM: Training and validation accuracy with parameter combination: C: 0.5 and penalty: l2

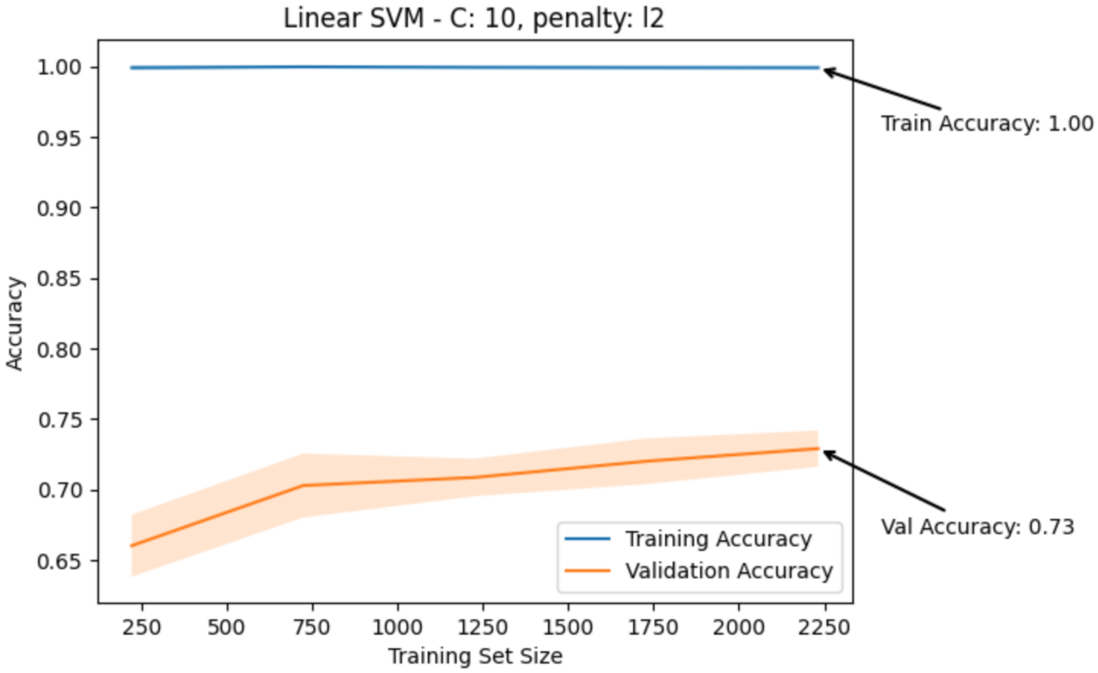


Figure E.7.: Linear SVM: Training and validation accuracy with parameter combination: C: 10 and penalty: l2

NTNU

Norwegian University of
Science and Technology