Frikk Hald Andersen, Eirik Dahlen

Sesame Street Pays Attention to Pro-Eating Disorder

Classification of Pro-Eating Disorder Posts on Social Media Using Attention-Based Models

Master's thesis in Computer Science
Supervisor: Björn Gambäck
June 2021

Master's thesis

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

NTNU
Kunnskap for en bedre verden

Frikk Hald Andersen, Eirik Dahlen

Sesame Street Pays Attention to Pro-Eating Disorder

Classification of Pro-Eating Disorder Posts on Social Media Using Attention-Based Models

Master's thesis in Computer Science
Supervisor: Björn Gambäck
June 2021

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 has made it easier for people to access content and create online communities with like-minded individuals. One such online community is called pro-eating disorder (abbreviated pro-ED), which has a positive attitude towards eating disorders, despite it being the mental illness with the highest mortality rate. These communities promote eating disorders as a lifestyle choice rather than acknowledging it as a deadly mental disease and encourage each other to maintain dangerous behavior. Although some social media platforms have taken measures to restrict the publication of pro-ED content, these communities are still active today. Recent studies show that pro-ED users can successfully be classified on Twitter using standard machine learning algorithms and natural language processing techniques. However, a more natural approach would be to look at pro-ED *posts*, as social media users often write about more than one topic in their posts.

In recent years, new deep learning language models based on the *Attention* mechanism and *Transformer* architecture have been proposed. Although these attention-based models provide state-of-the-art results for a large number of natural language processing tasks, applying them to classify pro-ED posts is still untested. This Thesis focuses on the implementation and fine-tuning of several attention-based models originating from the renowned model *Bidirectional Encoder Representations from Transformers* (BERT) and how they can be applied to the task of classifying pro-ED posts from several social media platforms. In order to do so, three new pro-ED datasets were collected, processed, and annotated from the Twitter and Reddit platforms. A set of annotation criteria was constructed to label a post as either *pro-ED*, *pro-recovery*, or *unrelated* based on its content.

The main dataset contribution is a manually annotated Twitter dataset consisting of 16 389 posts, while a test dataset of 376 manually annotated Reddit posts and a semi-automatically annotated Twitter dataset of 136 846 posts were also collected. These datasets were used in three experiments to investigate how the attention-based models performed on the task of classifying pro-ED posts. The models were tested and evaluated both individually and in several ensemble architectures.

The results show that the attention-based models outperform standard machine learning algorithms on the task of classifying pro-ED posts in social medias. The best-performing systems were all based on a stacked ensemble architecture, achieving a weighted average macro F_1-score of 0.939 when fine-tuned and tested on Twitter data and 0.816 when tested on Reddit data. Individually, BERTweet was the best model for classifying tweets, while ERNIE 2.0 proved the most robust model when evaluated on cross-platform tasks. These results show that attention-based models can be combined to create state-of-the-art systems for the automatic classification of pro-ED posts.

Sammendrag

Sosiale medier har gjort det lettere å finne innhold og skape nettsamfunn med likesinnede individer. Ett av disse nettsamfunnene kalles pro-eating disorder (forkortet pro-ED), som kjennetegnes ved at de har en positiv holdning til det å ha en spiseforstyrrelse, til tross for at det er den mentale lidelsen med høyest dødsrate. Disse nettsamfunnene fremmer spiseforstyrrelser som et livsstilsvalg heller enn en dødelig mental lidelse, ved å oppfordre til skadelig oppførsel og ved å motivere hverandre til å opprettholde sykdommen sin. Selv om enkelte sosiale medieplattformer har tatt grep for å begrense spredning av pro-ED-innhold er disse nettsamfunnene fortsatt aktive i dag. Tidligere studier har vist at pro-ED-brukere kan bli klassifisert på Twitter ved bruk av standard maskinlæringsmodeller og språkbehandlingsteknikker. Dette er ikke nødvendigvis den mest effektive tilnærmingen til problemstillingen, ettersom brukere av sosiale medier ofte publiserer innhold om mer enn ett tema. En mer naturlig tilnærming vil derfor være å se på selve postene i stedet.

De siste årene har det kommet mange nye dyp lærings-modeller basert på *Attention*-mekanismen og *Transformer*-arkitekturen. Disse attention-baserte modellene har gitt state-of-the-art resultater på mange spårkbehandlingsoppgaver, men har enda ikke blitt anvendt til å klassifisere pro-ED-poster. Denne masteroppgaven fokuserer på å implementere flere attention-baserte modeller basert på den kjente modellen *Bidirectional Encoder Representations from Transformers* (BERT), og hvordan disse kan bli anvendt på oppgaven å klassifisere pro-ED-poster på flere sosiale medieplattformer. For å gjennomføre dette ble tre nye pro-ED-datasett fra Twitter og Reddit samlet inn, prosessert og annotert. Det ble definert en mengde annoteringskriterier for å klassifisere innlegg som enten *pro-ED*, *pro-recovery* eller *unrelated* basert på innholdet i innlegget.

Et av hovedbidragene fra denne oppgaven er et manuelt annotert datasett fra Twitter, bestående av 16 389 tweeter. I tillegg har et testdatasett med 376 manuelt annoterte innlegg fra Reddit og et semiautomatisk annotert dataset bestående av 136 846 innlegg fra Twitter blitt annotert. Disse datasettene ble brukt i tre eksperimenter som undersøkte hvordan attention-baserte modeller presterte på problemet å klassifisere pro-ED-poster fra sosiale medier. Modellene ble testet og evaluert både individuelt, og i forskjellige ensemble-arktitekturer.

Resultatene viser at attention-baserte modeller utkonkurrerer standard maskinlæringsmodeller på oppgaven. Systemene som ga best resultater var basert på en stablet ensemble-arkitektur, med en vektet markrogjennomsnittlig F_1-verdi på 0.939, når den er trent og testet på data fra Twitter, og en verdi på 0.816 når den er testet på data fra Reddit. Ser man på de individuelle modellene, er BERTweet den beste på å klassifisere tweeter, mens ERNIE 2.0 er best når det gjelder kryssplatformsoppgaver. Disse resultatene viser at attention-baserte modeller kan bli kombinert i ensemble-arkitekturer for å gi state-of-the-art systemer på oppgaven å automatisk klassifisere pro-ED-poster på sosiale medier.

Preface

This Master's Thesis was written during the spring of 2021, as a part of our Master of Science (MSc) degree in Computer Science at the Department of Computer Science (IDI) at the Norwegian University of Science and Technology (NTNU) in Trondheim, Norway.

We would like to thank our great supervisor Björn Gambäck for providing interesting articles and helping us define the research, as well as giving advice on model choice and valuable feedback in general. We would also like to thank Ingrid Giæver, Andrea Nornes, and Martine Gran for their previous work on their Master's Theses on the task of classifying pro-ED users on Twitter, which laid the foundation for this Thesis. Additionally, a thank goes out to Tao Wang for providing us his complete list of tags and keywords for collecting eating disorder data from social media. We would also like to thank the Prodigy team for providing a interim license to their annotation software. Finally, we would like to thank Jacob Devlin and Yu Sun for permitting the use of figures from their papers.

Frikk Hald Andersen, Eirik Dahlen

Trondheim, June 5, 2021

Contents

List of Figures

List of Tables

Acronyms

ALBERT A Lite BERT. ix, 21, 23, 77, 78, 81, 84

ANN Artificial Neural Network. ix, 15, 16

BERT Bidirectional Encoder Representations from Transformers. vii, ix, 3, 9, 19, 20, 21, 22, 23, 25, 32, 41, 42, 43, 44, 77, 78, 81, 84, 85, 86, 92, 95, 97, 102, 104, 105, 111, 112, 113, 114, 119, 122

BERTweet BERT pre-trained on English Tweets. ix, 9, 23, 42, 84, 85, 86, 92, 97, 98, 99, 100, 101, 102, 103, 104, 111, 112, 114, 118, 119, 122

BoW Bag of Words. ix, 30, 32

BPE Byte-Pair Encoding. ix, 33

DistilBERT Distilled version of BERT. ix, 9, 21, 22, 32, 43, 77, 78, 80, 81, 84, 85, 86, 92, 97, 99, 100, 101, 102, 103, 112, 114, 116, 122

ED Eating Disorder. ix, 1, 2, 7, 37, 39, 40

ERNIE 2.0 ERNIE: A Continual Pre-training Framework for Language Understanding. vii, ix, 3, 9, 24, 25, 84, 85, 86, 92, 97, 102, 103, 104, 105, 112, 113, 114, 118, 119, 122

FFNN Feed-Forward Neural Network. ix, 16, 17, 99, 100, 102, 103, 115, 118, 122

LDA Latent Dirichlet Allocation. ix, 32

LR Logistic Regression. ix, 14

MLM Masked Language Model. ix, 21, 22, 23, 25

MLP Multilayer Perceptron. ix, 16

NLP Natural Language Processing. ix, 2, 3, 15, 17, 18, 21, 22, 23, 29, 30, 32, 33, 41, 42, 43, 86, 112, 121

NLP4IF Natural Language Processing for Internet Freedom: Censorship, Disinformation, and Propaganda. ix, 42

NSP Next Sentence Prediction. ix, 20, 21, 23

NTNU Norwegian University of Science and Technology. iii, ix, 96

OOV Out-of-Vocabulary. ix, 30, 33

pro-ED Pro-Eating Disorder. ix, 1, 2, 3, 6, 7, 8, 9, 12, 37, 38, 39, 40, 41, 44, 45, 47, 48, 49, 50, 51, 52, 55, 57, 58, 59, 60, 61, 62, 63, 64, 65, 68, 69, 70, 71, 72, 73, 74, 75, 77, 81, 83, 84, 86, 92, 93, 97, 98, 99, 100, 101, 102, 103, 105, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124

RNN Recurrent Neural Network. ix, 16, 17, 18, 19

RoBERTa A Robustly Optimized BERT Pretraining Approach. ix, 3, 9, 23, 42, 84, 85, 86, 92, 98, 99, 100, 102, 103, 112, 122

SMM4H Social Media Mining for Health Applications Workshop and Shared Task. ix, 41, 42

SVM Support Vector Machine. ix, 9, 13, 40, 41, 43, 80, 81, 83, 87, 91, 92, 93, 94, 96, 98, 99, 100, 102, 104, 112, 113, 114, 118, 119, 122

TF-IDF Term Frequency-Inverse Document Frequency. ix, 31, 41, 43, 83, 96, 117, 118

1 Introduction

Pro-eating disorder (pro-ED) is a term referring to an individual or a group of individuals who have a positive view on eating disorders (EDs). With the rise of social media and micro-blogging sites like Twitter and Reddit, or websites in general, subcultures of pro-ED users have emerged over the years. A pro-ED user thinks that having an eating disorder is a lifestyle choice and encourages dangerous behaviors like starving, self-harm, and extreme weight loss techniques. This is done by sharing graphical and textual content to encourage, support, and motivate other users to continue their efforts with the disorder (Borzekowski et al., 2010). Much of this content is considered harmful and triggering for people who have a predisposition to disordered eating. This Master's Thesis focus on the detection of such content by exploiting newly developed techniques in the field of deep learning. These new techniques are based on Attention and Transformers and will be further described in Section 2.2.

This introductory chapter presents the background for this Thesis and the motivation for contributing research to the field of classifying pro-ED content on social media. In the following sections, social media, eating disorders, and pro-ED are presented along with the presence of pro-ED content on social media. The fourth section will present the main goal of this Thesis. To structure the study towards reaching the goal, four research questions are formulated and presented together with the goal. The fifth section describe the methodologies used for our research, followed by the research's contributions. Lastly, an overview of the structure of this Master's Thesis is presented.

1.1 Background and Motivation

Eating disorders (EDs) are mental illnesses affecting a large part of the world's population and can lead to serious health consequences (Smink et al., 2012). There are several types of EDs, with Anorexia Nervosa, Bulimia Nervosa, and Binge Eating Disorder (anorexia, bulimia, and binge eating, for short) being the most common. Anorexia experiences the highest mortality rate of *all* mental illnesses, where one out of five anorexia deaths were suicides (Arcelus et al., 2011).

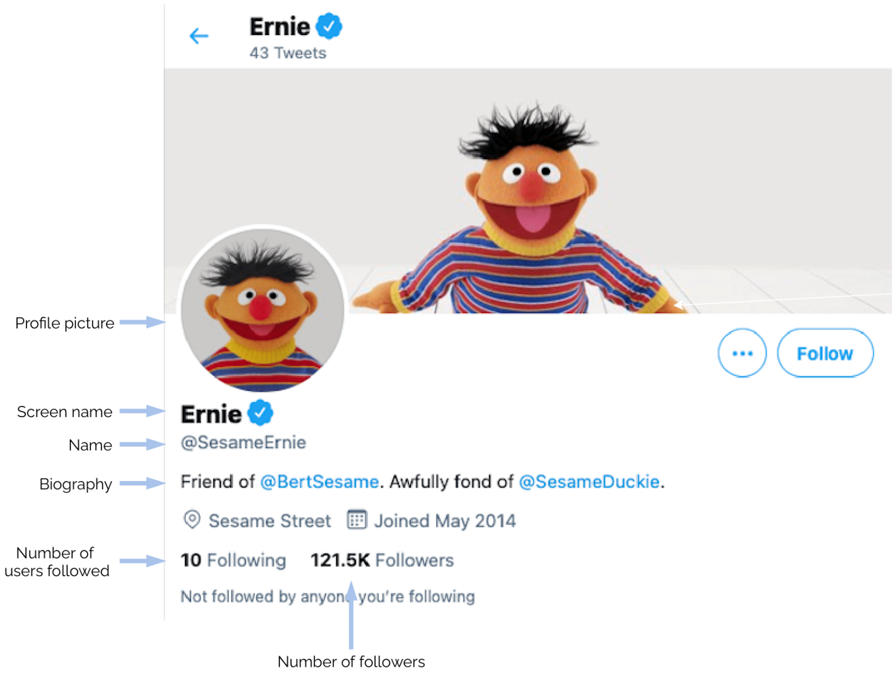
The combination of a huge global prevalence of people suffering from eating disorders and the growth of social media has made it possible for pro-ED communities to emerge online. Although some of the users and communities focus on recovery, others focus on keeping their eating disorder. Previous research has investigated the motivation behind posting pro-ED content, and found the driving factors to be the need for social support and having a venue where users could express themselves without judgment (Yeshua-Katz and Martins, 2012). Additionally, this online activity was used as a way of coping with the stigmatization associated with EDs. The same study showed that people suffering from an ED felt more comfortable online because they did not reveal their true identity and were not exposed to humiliation from friends and family while still receiving the support they needed. Although many find support in these communities, some of the posted content can be harmful and have great negative effects on the viewer. Bardone-Cone and Cass (2007) have shown that viewing pro-ED content can lead to lower social self-esteem, lower appearance self-efficacy, and a higher need to exercise.

One challenge for social media platforms is to deal with unwanted content while also preserving free speech. When Twitter developed new rules for addressing misleading and manipulated media, their research showed exactly that (Roth and Achuthan, 2020). 45 % of the participants in the study who opposed the removal of manipulated media addressed the impact of free expression and censorship. On the other hand, the participants recognized the threat this type of posts poses, and 90 % agreed that placing some label or public service announcement would be acceptable. This labeling is mainly done manually by moderators today. Recently, companies have developed efficient Natural Language Processing (NLP) algorithms that can detect and remove such data from their platforms automatically,[1] because of the increasing amount of content. Previous research has also shown that such methods can be applied to detect pro-ED users on Twitter with good results using standard machine learning algorithms (Giæver, 2018; Nornes and Gran, 2019). During the fall of 2020, a preliminary study for this Master's Thesis was conducted, which built on the studies by Nornes and Gran, and Giæver. The preliminary study focused on a comparative analysis for the task of classifying pro-ED users on social media, between new deep learning algorithms based on the *Attention* mechanism and *Transformer* architecture (presented in Section 2.2), and the state-of-the-art language models from Nornes and Gran. The results from the study are presented in Chapter 6 (Page 77). Although showing promising results when detecting pro-ED **users**, this does not necessarily mean that every post by a pro-ED user includes pro-ED content. By building on the work done by Nornes and Gran, and Giæver, a system for detecting pro-ED **posts** is within reach.

Previous research has also mainly focused on pro-ED communities on **Twitter**, but they are also present on other social media. The contents posted by pro-ED communities across different social media platforms are found to be thematic similar, although the usage and language are in general different (Borzekowski et al., 2010; Branley and Covey, 2017; Cavazos-Rehg et al., 2019). A classification system developed for detecting pro-ED content on one social media platform could possibly also be applicable to classify content on other platforms. By gathering data from different social media platforms, a generalized classification system could be tested.

[1] https://help.Instagram.com/700284123459336?ref=ig_about

The latest research in the field of NLP shows that pre-trained language models using attention-based methods have gained state-of-the-art results on several text processing tasks. One such model is Google's Bidirectional Encoder Representations from Transformers, BERT for short (Devlin et al., 2018). Although achieving high performance, extensions like RoBERTa and ERNIE 2.0 have proved to perform better by optimizing the learning procedures of BERT. However, the current state-of-the-art in automatic classification of pro-ED users does not utilize attention-based methods. This Master's Thesis will, therefore, apply state-of-the-art models for text classification to the domain of pro-ED content on social media.



Ernie

43 Tweets

Ernie

@SesameErnie

Friend of @BertSesame. Awfully fond of @SesameDuckie.

Sesame Street Joined May 2014

10 Following 121.5K Followers

Not followed by anyone you're following

Figure 1.1: Example of a Twitter profile.

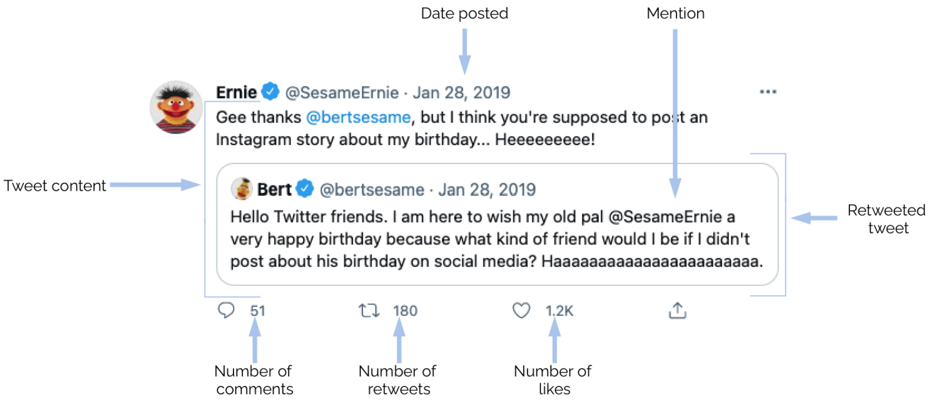
1.2 Social Media

Social media refers to digital tools that allow people to efficiently communicate and share content in real-time using the Internet. The ability to interact with others at any time using digital devices, such as smartphones and laptops, has brought a new dimension to how people can build and maintain relations with others. Several types of social media have emerged over the years, e.g., social networks, media networks, and discussion networks. Common in all is the existence of *communities*, which refers to a group of people that share the same type of content on social medias.

In a social network, the participants are linked together through virtual friendships or *followers*, and exchange thoughts, ideas, and content. Twitter and Facebook are examples of social networks. Media networks specialize in the sharing of media content, like photos and videos. Discussion networks are networks designed for in-depth discussions and open conversation, like Reddit.

1.2.1 Twitter

Twitter is a type of social medium called a micro-blogging service. The users can publish short posts, called tweets, and view tweets published by other users. A tweet can not be longer than 280 characters, which requires the users to be straight to the point in each post. This paragraph has now exceeded this limit.[2] Twitter is a social network where the users are able to *follow* each other, and by following another user, a connection between the users is made. When several users are followed by and following others, it can be thought of as a network of virtual social connections. Figure 1.1 shows what a user profile page looks like, with the follower count and user information.



Ernie @SesameErnie · Jan 28, 2019

Gee thanks @bertsesame, but I think you're supposed to post an Instagram story about my birthday... Heeeeeeeee!

Bert @bertsesame · Jan 28, 2019

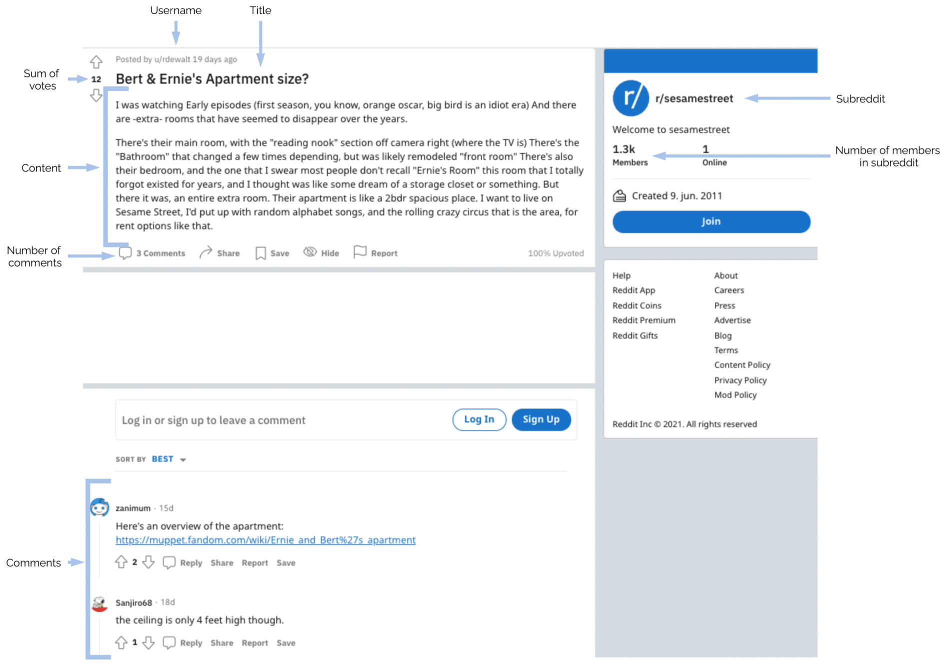
Hello Twitter friends. I am here to wish my old pal @SesameErnie a very happy birthday because what kind of friend would I be if I didn't post about his birthday on social media? Haaaaaaaaaaaaaaaaaaaaaa.

Figure 1.2: Example of a tweet.

Tweets are, as seen, short posts that often consider a specific topic. To highlight the topic of the tweet, users tend to include a hashtag followed by the topic. Hashtags are widely

[2] the underlined r was character number 280.

used on Twitter and make it easy to find specific content. When using the search feature for a hashtag, all tweets including the hashtag will appear. Thus, finding a community that cares about the same as oneself is quite easy. Other commonly used features on Twitter are mentions and retweets. A mention is simply a mention of another user, denoted as *@username*, where the mentioned user gets a notification. A retweet is to re-post another user's tweet. Figure 1.2 shows what a tweet could look like.



Posted by u/rdewalt 19 days ago

Bert & Ernie's Apartment size?

I was watching Early episodes (first season, you know, orange oscar, big bird is an idiot era) And there are -extra- rooms that have seemed to disappear over the years.

There's their main room, with the "reading nook" section off camera right (where the TV is) There's the "Bathroom" that changed a few times depending, but was likely remodeled "front room" There's also their bedroom, and the one that I swear most people don't recall "Ernie's Room" this room that I totally forgot existed for years, and I thought was like some dream of a storage closet or something. But there it was, an entire extra room. Their apartment is like a 2bdr spacious place. I want to live on Sesame Street, I'd put up with random alphabet songs, and the rolling crazy circus that is the area, for rent options like that.

Welcome to sesamestreet

1.3k Members
1 Online

Created 9. jun. 2011

Reddit Inc © 2021. All rights reserved

Log in or sign up to leave a comment

zanimum · 15d
Here's an overview of the apartment:
https://muppet.fandom.com/wiki/Ernie_and_Bert%27s_apartment

Sanjiro68 · 18d
the ceiling is only 4 feet high though.

Figure 1.3: Example of a post on Reddit.

1.2.2 Reddit

Reddit is a discussion network designed to let users post about topics, and thereafter users can comment and respond to each other in a thread. A discussion thus works in much the same way as a regular conversation. People can bring new perspectives into the discussion, address what specific persons have said and express their opinions by *voting* up or down on others' responses. Reddit is divided into subreddits, which can be considered different rooms where a specific topic is discussed. An example of a post in a subreddit is shown in Figure 1.3.

1.3 Eating Disorders and Pro-ED

This section will serve as an introduction to the topic of pro-eating disorder (pro-ED) and its presence in social media. Social media exposes people to a wide range of content, which is not always promoting a healthy lifestyle and may be subject to unfortunate disinformation (Boniel-Nissim and Latzer, 2016).

1.3.1 Eating Disorders

An eating disorder is a mental disorder characterized as a disturbance to a person's eating behaviors. It is a complex illness covering various types of behavior, where the most common expressions are Anorexia Nervosa, Bulimia Nervosa, and Binge Eating Disorder. Although these types of eating disorders deal differently with eating habits, they have in common the use of food as a means to handle emotions and self-control (Polivy and Herman, 2002). Because of its complex nature, it is hard to point out which factors contribute to developing an eating disorder. Polivy and Herman suggest that the presence of body dissatisfaction, along with the need for control and inadequate identity formation, are such factors.

1.3.2 Pro-Eating Disorder

With the introduction of social media into daily life, a new platform for content sharing has emerged. As a consequence, people now tend to search for information online before asking a professional about the given topic (Zhao and Zhang, 2017; Kummervold et al., 2008). Since everyone can publish content on social media, not all information out there is trustworthy or harmless. As mentioned in the introductory section of this chapter, some of the communities that have appeared on social media support and promote an eating disorder lifestyle, so-called pro-eating disorder communities. These communities exist almost everywhere on the internet, including forums, private blogs, and traditional social media, such as Facebook, Instagram, Twitter, and Tumblr.

1.3.3 Pro-ED Content

The pro-ED communities publish content promoting dangerous and unhealthy behavior as a way of living rather than symptoms of a mental illness. Examples of such content are unhealthy weight loss techniques, how to hide symptoms from friends and family, and thinspiration or bonespiration content (content glorifying extreme thinness). Despite being harmful, many users participate in these communities for emotional support and as a place not to be judged, rather than encouraging others to obtain their behavior. In addition, they may feel that people around them do not understand their situation and seek social media to find like-minded who can support them in their struggles (Boniel-Nissim and Latzer, 2016). As previously mentioned, content on social media can be found by anyone. Even though some of the content is not meant to influence others, the risk will be there. One cannot say exposure to pro-ED content causes people to develop an eating disorder; however, findings suggest that exposure can trigger a predisposition to disordered eating or prevent people from recovery (Hilton, 2018).

Several social media platforms have taken action to restrict the publicity of content classified as pro-ED by banning certain hashtags, suspending users, or providing advisory content as a response to search words (Tumblr, 2012; Instagram, 2012; Pinterest, 2012). Examples of tags that are banned or will provide advisory content when searched for are *anorexia*, *bulimia*, *purge*, and *thinspiration.* This has led to the use of lexical variations or abbreviations of the original word to avoid the restrictions: *ana* or *proana* for *anorexia*, *mia* or promia for *bulimia* and *thinspo* or *thinsp0* for *thinspiration.* Apart from the use of hashtags to make the pro-ED content more available, another common feature among users who are considered as pro-ED (see Chapter 5 Page 47 for how the labeling of users is carried out) is the sharing of weight control methods and weight goals. When posting about weight goals, abbreviations as *sw* (start weight), *cw* (current weight), and *gw* (goal weight) are often used.

1.3.4 Pro-Recovery

In contrast to the pro-ED communities, pro-recovery communities have emerged as well. Pro-recovery focuses on helping people out of an eating disorder and towards recovery. They discuss the health challenges of eating disorders, how people can seek help if they need it, and generally serve as support for those who want to recover or those already in a recovery process. The people contributing to pro-recovery communities are often people suffering from an ED themselves and trying to recover, family and friends of people suffering from an ED, or health professionals. As with pro-ED communities, the pro-recovery community also functions as a place for emotional support and a place to share experiences and motivate people struggling.

1.4 Goals and Research Questions

Based on the motivation described in the preceding section, the goal of this Master's Thesis is to switch the focus of detecting pro-ED content from *users*, as previous research did, to *posts.*

Goal *Identify pro-eating disorder posts from various social media platforms by using attention-based models.*

By collecting and annotating a dataset of social media posts, an attention-based model can be fine-tuned to classify pro-ED content. To reach this goal, four research questions are defined to guide the research in a structured manner. The research questions are presented below.

Research Question 1 *How are Twitter and Reddit used by members of pro-eating disorder communities?*

This research question will investigate what an online pro-eating disorder community is and how these communities interact on the social media platforms Twitter and Reddit. In addition, characteristics of the communities' use of Twitter and Reddit will be explored and compared with *regular* users.

Research Question 2 *What criteria should be used in the annotation of pro-eating disorder posts?*

The focus of the second research question is to explore the field of annotation and use the insights from Research Question 1 to make a set of annotation rules. These rules will be used to evaluate if the social media posts can be considered as part of a pro-eating disorder community or not.

Research Question 3 *How can attention-based models be combined to improve the classification of pro-eating disorder posts?*

The third research question considers the task of combining attention-based models to improve the performance of this classification task. An ensemble learner consisting of both a baseline and several attention-based models will be created.

Research Question 4 *How do attention-based models trained on data from one social media platform perform when tested on data from another platform on the task of classifying pro-eating disorder posts?*

The last research question focuses on the performance of attention-based models when tested on data from different social media platforms. Data from Twitter and Reddit will be collected to answer this research question. The motivation behind investigating this research question lies in the possibility to create systems that can detect pro-ED content on social media platforms in general.

1.5 Research Method

To achieve the goal and answer the research questions of this Master's Thesis, several methodologies were used. For both Research Question 1 and 2, a qualitative research method were selected to gather insight about the topics, and the proposal by Jacobsen (2015) for a qualitative research process was selected. First, a literature review was chosen as method for collecting previous research in the field of this study, which is an approach where data is collected and interpreted in order to gather insights about a topic or answer the research questions. Previous research of pro-ED communities and annotation was explored to gather relevant insight into how existing research could be utilized to answer the research questions. Further, a conceptual and theoretical understanding of the data was established. Lastly, a presentation of the research is found in Chapter 4 (Page 37).

The process was highly iterative, as the understanding of relevant concepts often required further collection of previous research.

To answer Research Question 3 and 4 a qualitative research method, a quantitative research method and experiments were used. The qualitative method was similar to the approach for Research Question 1 and 2. The quantitative method included scraping data from the social media platforms Twitter and Reddit, which is explained in detail in Chapter 5 (Page 47). The qualitative research resulted in knowledge about state-of-the-art of natural language processing, while the quantitative research resulted in three datasets. Lastly, three experiments were carried out, and several pre-trained language models were implemented and fine-tuned using the collected datasets.

1.6 Contributions

The contributions from this Master's Thesis are described in the following list:

1. *An overview of how attention-based models are applied to the task of social media text classification.*
2. *A set of criteria for annotating pro-ED and pro-recovery content of social media.*
3. *A manually annotated pro-ED dataset consisting of 16 389 tweets.*
4. *A manually annotated pro-ED dataset consisting of 376 posts from Reddit.*
5. *A semi-automatic annotated pro-ED dataset consisting of 136 846 tweets.*
6. *An ensemble of the attention-based models BERT, DistilBERT, RoBERTa, BERTweet, and ERNIE 2.0, and an SVM, with a majority voter as the meta-classifier.*
7. *An ensemble of the attention-based models BERT, DistilBERT, RoBERTa, BERTweet, and ERNIE 2.0, and an SVM, with a feed-forward neural network as the meta-classifier.*

1.7 Report Structure

The rest of this Master's Thesis is structured in the following manner:

Chapter 2 presents the relevant background theory and technologies used in either this Thesis or relevant work.

Chapter 3 elaborates on the existing research related to the pro-ED community and classification of social media content.

Chapter 4 introduces the datasets and how they were collected, processed, and annotated.

Chapter 5 presents the models and results from the preliminary study to this Master's Thesis.

Chapter 6 explains the architecture of the models used in the experiments.

Chapter 7 contains the details of the experiment, including the experimental plan, setup, and results.

Chapter 8 discusses the research process and evaluates the experimental results in the light of the goal and research questions.

Chapter 9 concludes the discussion of the Thesis, presents the contributions, ethical considerations, and potential future work.

2 Machine Learning for Text Classification

The following chapter will cover the main concepts needed to understand the content concerning machine learning applied to text classification used in this Master's Thesis. The first section will present machine learning concepts and algorithms, followed by an introduction to deep learning and attention-based models. Lastly, will the performance measures used to evaluate the experiments of this Master's Thesis be presented. Most sections in this chapter were written during the preliminary study, and only minor changes are made to these sections. Additional contributions to this chapter are Section 2.1.2, 2.1.5, and 2.2.8 through 2.2.10.

2.1 Machine Learning Concepts

Machine learning is an application of artificial intelligence focusing on systems that can automatically learn and improve from experience without being explicitly programmed. The goal is to make a machine learning model learn from some data using a learning algorithm and then apply it to unseen data to predict a future outcome. In general, a machine learning model *learns* how to treat new instances of data by processing its attributes, which are called **features**. A more formal definition of machine learning is provided by Mitchell (1997, Page 2): *A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.* This definition will be addressed later on in this section with regards to the **task**, the **performance measure** and the **experience**.

2.1.1 Classification and Supervised Learning

Machine learning algorithms can be categorized as supervised or unsupervised, although there are other possible learning paradigms. The categorization depends on how the algorithm **experiences** the data and its features. Whereas unsupervised learning focuses on gaining knowledge from features by looking at the structure of the data, supervised learning requires labeled data representing the correct answer. The model will learn what feature values are typical for each label, and from this, predict the label of new data instances. The term *supervised* comes from the fact that these labels are provided by a supervisor (most often humans), telling the machine what is correct.

Although machine learning can be applied to a variety of **tasks**, this Master's Thesis will focus on the task of supervised text classification. Classification is the process of predicting what class a new instance of data belongs to. The classification task can either be binary, where data instances will belong to one out of two classes, or multiclass, where data instances will belong to one out of three or more classes. Supervised classification is thus predicting the class of new data instances based on the labels given, where each different label represents a class. In the task of classifying pro-ED posts in social media, *pro-ED*, *pro-recovery* and *unrelated* would be the classes, and each post will be labeled as one of these in the dataset.

2.1.2 Overfitting

Overfitting is a common problem related to supervised machine learning that occurs when a learning algorithm has become too well fitted on the training data and thus performs poorly on unseen test data. An overfitted model contains more parameters than can be justified by the data, meaning that the model remembers many examples instead of learning from the feature. When training a supervised learning algorithm, the training and validation error can be used to detect overfitting. If the validation error increases while the training error decreases over time, this may be evidence of overfitting. When a model is overfitted, it will not generalize well to other types of unseen data.

2.1.3 Machine Learning Models

There exist several different machine learning models that will perform and behave differently based on the task given to them. A supervised machine learning model analyses the training data and produces a function that is later used for the classification of unseen data. In this section, the most used machine learning models in the field of supervised text classification are presented.

Naïve Bayes Classifier

The Naïve Bayes Classifier is a probabilistic machine learning model based on the Bayes Theorem. The model computes the probability of each proposed class, given an input, using Bayes Theorem as defined in Equation 2.1, and outputs the class with the highest probability.

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

The model is called *naïve* because of the assumption of conditional independence between the input features. For the task of text classification, Naïve Bayes assumes that each word in the input sequence is independent of the others, which is seldom the case in natural language. Even though the independence assumption makes the Naïve Bayes Classifier a simple and limited model, it is highly scalable and serves as a popular baseline method for many classification tasks.

Support Vector Machines

A Support Vector Machine (SVM) is a supervised learning model commonly used for binary classification and regression tasks. The algorithm aims to separate the data samples in an n-dimensional feature space into two classes by finding a hyperplane separating the classes, as shown in Figure 2.1. The nearest data points to this hyperplane define the support vectors, and the distance between the hyperplane and the support vectors is called the margin. The optimal hyperplane has maximized this margin. After training, only the hyperplane and the support vectors are used to determine the classes of new data samples, making the SVM memory-efficient. In many cases, finding a linearly separable hyperplane for the n dimensional feature space is impossible. This problem is solved using kernel functions, or the *kernel trick* as proposed by Boser et al. (1992). The kernel trick maps the features into a higher-dimensional space where the classes are linearly separable. The kernel itself is essentially a function that defines the similarity between two vectors using the vectors' inner product.

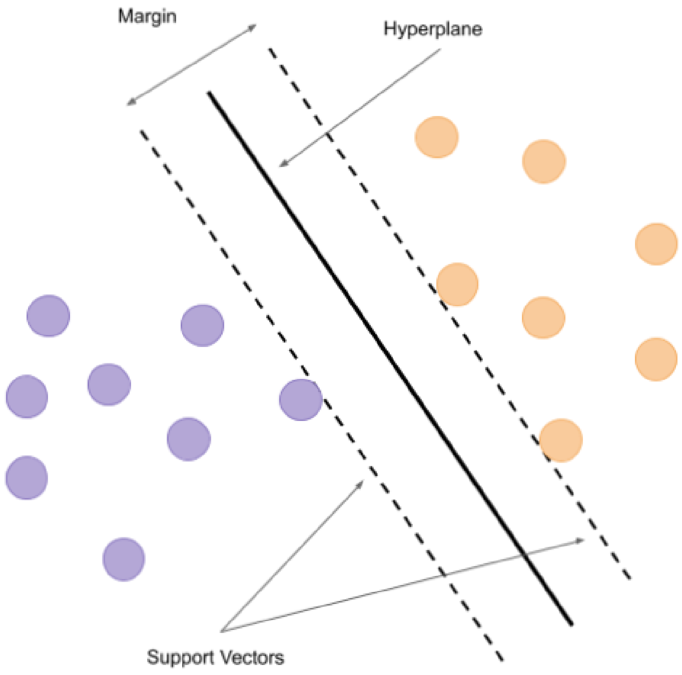


Figure 2.1: A Support Vector Machine in a two-dimensional space.

Logistic Regression

Logistic Regression (LR) is a statistical model often used for binary supervised machine learning classification. LR is a linear method, but the predictions are transformed using the logistic function (or sigmoid function) as described in Equation 2.2, which is an S-shaped curve that maps real-valued numbers into a number in the range $[0, 1]$. The logistic function output shows both the most probable class a data instance belongs to and how certain the model is that the given data instance belongs to the predicted class. Values close to 1 indicates that the model is confident that the instance belongs to the given class, while values close to 0 indicate the opposite.

$$S(x) = \frac{1}{1 + e^{-x}} \tag{2.2}$$

The estimation of parameters in LR is done through the Maximum-Likelihood Estimation, which is an iterative approach used to find the optimal values for the weights in the model by minimizing the error in the probabilities predicted by the model. Although LR is often used for binary classification, it can also be used on a multiclass task. This is done either by using several LR models or using an extension called multinominal logistic regression.

2.1.4 Ensemble Learning

Ensemble learning is a method that combines multiple models to solve a problem and is primarily used to improve the performance of the given task. It works in the same way as a person would ask another doctor for a second opinion. If several doctors give the same diagnosis, the person will trust that the diagnosis is correct. In the case of an ensemble learner, several models make a prediction, and a final decision is made based on a combination of all the predictions.

To fully exploit the benefits of an ensemble system, it should be some diversity between the models. The main contribution of an ensemble is the ability to correct the error of the individual models. If there is diversity between the models, the thought is that each model will make different mistakes, which will be eliminated by the other models' correct predictions. Diversity can be achieved in several ways: either with different models (stacking) or by using different features and training the models on different parts of the data (bagging).

When an ensemble learner is making the final decision of what the outcome should be, there are several ways to combine the models' prediction to produce an answer. The simplest way is to use hard majority voting, which is the method of choosing the class that the majority of the models predicted. Consider that model A predicted class 1, model B predicted class 2, and model C predicted class 1. Then the ensemble classifier would choose class 1 as the final output. Another way of deciding which class the data instance should be predicted is to aggregate each model's predictions, also called soft voting. Lets say model A predicts class 1 with 80 % certainty and class 2 with 20 % certainty [0.8, 0.2], model B predicts [0.4, 0.6] and model C predicts [0.6, 0.4]. The aggregated prediction will then be:

$$[0.8 + 0.4 + 0.6, 0.2 + 0.6 + 0.4] = [1.8, 1.2]$$

and the final prediction should therefore be class 1. There are also other ways to decide the final prediction, e.g., using a neural network on top of the models.

2.1.5 Oversampling and Undersampling

Oversampling and undersampling are techniques used to adjust the distribution of classes in a dataset. When performing an analysis of a dataset, the class distribution may be imbalanced or inadequate for the task at hand. To adjust the ratio, either oversampling and undersampling can be applied, depending on the information available and the nature of the task.

Oversampling is applied when there is a need for more data, usually by sampling data instances to the minority class. Several techniques with different complexity can be used to sample data, where the simplest is random oversampling. Random oversampling duplicates instances from the minority class in the dataset. Undersampling is the technique of underrepresenting the majority class, either by deleting instances or deciding which instances to keep. Random undersampling is the basic technique, which removes samples from the majority class until the desired distribution is present in the dataset.

2.2 Deep Learning

Deep learning is a subcategory of machine learning based on Artificial Neural Networks (ANN), described in Section 2.2.1. In recent years, deep learning techniques have proved great performance on many machine learning tasks. Although deep learning has been around for many years, the recent improvement in computational power and the increase of available data have made it possible to apply this field on a larger scale. This section will mainly focus on the use of deep learning techniques in the field of NLP. To fully understand the more complex state-of-the-art techniques like Attention, described in Section 2.2.3, a basic understanding of the field must be established first.

2.2.1 Artificial Neural Networks

Artificial Neural Networks are a collection of networks consisting of nodes, often called neurons or perceptrons, which are designed using the network of neurons in the human brain as inspiration. The goal of a neural network is to recognize patterns in data. The networks are, in essence, a directed graph of nodes with several layers. Every edge between nodes in an ANN has a weight. The nodes decide their output based on the input and an activation function, which is a non-linear transformation, enabling the network to learn both linear and non-linear functions. This output is multiplied by its weight before it is used in the next layer's sum of inputs. The learning itself is happening when these weights are iteratively adjusted between each node using the backpropagation algorithm. This adjustment is done using gradient descent, an optimization function for finding the weights that minimize a loss function. The loss function is based on the difference between a predicted value and the actual label.

Feed-Forward Neural Network

A Feed-Forward Neural Network (FFNN) is the typical example of deep learning models and is the simplest type of ANNs. FFNNs consist of an input layer, at least one hidden layer, and an output layer. Every node in a layer is connected to all the nodes in both the previous and the next layer. The input layer takes in the value of the input feature vector. The hidden layers are simply layers that are operating on output from other layers. Thus, there could be several hidden layers, making the network deep. When the hidden layers' calculations finally reach the output layer, the network has produced a result. Since data is only passed *forward* to the next layer, there are no loops and feedback circles between layers, hence the name **feed-forward** neural network. Another common term used for Feed-Forward neural networks is Multilayer Perceptron (MLP).

Training an FFNN has three major steps. First, it does a forward pass through the network to predict the input. Second, the model compares the predictions to the ground truth using a loss function, which estimates how bad the prediction is. Last, the model uses this predicted error value when propagating the network backward in an algorithm called backpropagation. In the backpropagation step, the model updates each node's weights based on the gradient of the loss function. This optimization is called gradient descent. The weights are adjusted to minimize the error value from the loss function, i.e., improve the prediction. It is this optimization technique that allows the network to learn.

Recurrent Neural Network

Recurrent Neural Networks (RNN) is a group of ANNs that considers previous output when processing an input in order to make use of sequential information. For sequential data, data instances are related to one another and are likely to influence their neighbors. RNNs take as input a sequence of data, such as a text, where the data instances' position comes in a particular order. When processing such sequences, it may be helpful to consider the context of the sequence. Where FFNNs only feed information forward to the next layer, RNNs add loops to layers, allowing the layers to keep some kind of memory of previous iterations. When a node gets an input and produces an output from the activation function, the output is fed forward to the next layer, as well as kept in memory for use when processing the next input. The node will then concatenate the next input with the value kept in memory before sending it to the activation function. The attribute of storing values in memory is called to keep a hidden state. By introducing the hidden state, the network has the ability to take advantage of previously processed data.

In FFNN and RNN, the gradient of a node is calculated with respect to the gradients from the layer before. Thus, if the gradient from the upstream node is small, the gradient for the node will be small, which causes the gradient to shrink for each layer while backpropagating. Therefore, the first layers of the network do not get the opportunity to learn as much as the layers closest to the output. This phenomenon is called *the vanishing gradient problem* and can be solved by introducing information sharing among nodes.

2.2.2 Encoder-Decoder Architecture

Many NLP tasks can be defined as sequence to sequence learning. Such tasks take a sequence as input and then outputs a sequence. A challenge with these tasks is that the input and output sequences may vary in size, and RNNs need the input sequences to be of equal size. One way of making RNNs handle sequence to sequence tasks is to use the Encoder-Decoder architecture (Sutskever et al., 2014).

The Encoder-Decoder architecture consists of three main parts: the encoder, the decoder, and a hidden vector. The encoder is a stack of multiple RNN cells which read the input sequentially, one element of the input sequence at the time. For each input t, the hidden vector is updated according to the input at that iteration. The RNNs outputs two things: the hidden vector and the output for the given stage. The encoder's output at each step will be of no use because it is only the hidden state that is taken into consideration later on and is therefore discarded. When the encoder has processed the whole sequence, the hidden vector represents the sequence's encoded version. Next, the hidden vector will be passed to the decoder. Each layer in the decoder takes three inputs, the original hidden vector, the hidden vector from the previous layer, and the previous layer's output. For the decoder to understand when the sequence starts and ends, tokens will be added to indicate the start and end of the sequence. The first iteration takes, therefore, only the start token and the hidden vector as input. For the intermediate iterations, the next word is predicted and is fed into the next iteration. Finally, the model predicts the end token, and the output will be passed to a final layer. This layer is a softmax activation function and will produce a probability distribution over the target classes.

2.2.3 Attention

The approaches for solving NLP tasks discussed so far encode the entire input sequence into one hidden vector/state. This could result in a loss of performance if the distance between the word it is trying to predict and the relevant information in the sequence is large. The input sequence's first words would contribute less to the final hidden state for long sequences, and thus information could be lost. To put this in context, consider

English to German translation, where the structure of sentences is quite different.[1] Verbs appear early in an English sentence while often at the end in a German sentence. A hidden state from an encoder in the Encoder-Decoder architecture will hence pay little attention to the English verb when trying to predict the last word of the German translation. One of the methods used to solve these long term-dependencies is the Attention mechanism.

Attention is a mechanism for focusing on particular elements or parts of a sequence when predicting an element. For natural language, words in a sentence are often correlated, and some words will contribute with more information than others when predicting the next word of a sentence. Where previously mentioned RNNs' encoders build a final hidden vector based on the last hidden state, the Attention mechanism makes the hidden state from each iteration available for the decoder. Thus, the decoder can weigh every hidden state individually and decide which words in the sentence it wants to pay attention to when predicting the next.

With the introduction of Attention, the decoder can attend to different parts of the input sequence by examining all the hidden states from the encoder. However, the encoder relies only on the previous iteration's hidden state when encoding the input sequence. The performance loss due to long-term dependencies appears not only between the encoder and decoder but could also appear inside the decoder. For example, consider the sentence *Ernie really likes to try new things, so he decided to write a master about NLP*, here *he* refers to *Ernie*. The hidden state will not necessarily capture this, and if it does, maybe only to some extent. By applying *self-attention*, the encoder will be able to attend to parts of the sequence as the decoder does,[2] and capture dependencies as the one in the given example.

The self-attention mechanism is realized by using three weight matrices - key K, value V, and query Q - which are created during training. Each element in the input sequence is given a key K, value V, and query representation Q, created by multiplying the input with the weight matrices. For calculating the attention scores, the dot product between element n's query representation and the key representations of all the sequence elements is calculated. The dot product is then scaled down by $\sqrt{n}$, where n is the dimension of the hidden state. Further, the softmax over the attention score is used to normalize it. The attention score is then multiplied with each value vector, with the purpose of **keeping** the values of the words to attend to and make the irrelevant words less important. Lastly, all the weighted value representations are summed up. This final vector is the output for element n. This vector now represents n's initial query representation's interaction with the other elements in the sequence.

[1] Sentence and sequence may be used interchangeably. A sentence in this context is a sequence of words. This specialization report covers the field of NLP, and it will be natural to explain the background theory with that in mind.

[2] The term *attend to* means *paying attention to* and is commonly used when talking about the Attention mechanism.

$$Attention(Q, K, V) = softmax(\frac{QK^\intercal}{\sqrt{n}})V$$

2.2.4 Transformers

The Transformer architecture was proposed in Vaswani et al. (2017) and is an Encoder-Decoder architecture based solely on the Attention mechanism. They found that Attention itself was powerful enough, not just to achieve the performance level of the recurrent sequential processing of RNNs, but also to improve it. As the architecture does not use recurrence or convolution, many of the calculations, e.g., the attention scores, can be carried out in parallel while still outperforming the aforementioned architectures.

The encoder is a stack of six identical layers where each layer consists of a multi-head self-attention mechanism and a feed-forward network. To each layer, it is employed a residual connection and a layer normalization. The decoder is composed in the same way as the encoder, with a stack of 6 identical layers and the same components. It also includes a third layer, which performs multi-head attention on output from the encoder. The self-attention layer is modified to make sure the decoder only considers subsequent output positions, i.e., the prediction of output element i only depends on the output of element $1, 2, ..., i - 1$.

The multi-head self-attention layer utilized in the Transformer allows for more complex representations of Attention than the Encoder-Decoder architecture's Attention. In Transformers, the layer consist of multiple instances, or *heads*, of self-attention, each with its own key, value, and query matrices, which are initialized randomly. When each head has its own matrices, it allows for different input treatments, enabling the heads to capture several sub-spaces of the input. To contextualize this, one head will learn to pay attention to pronouns, and another could learn to pay attention to the nouns in the input sequence. The attention score is calculated as explained in Section 2.2.3.

2.2.5 BERT - Bidirectional Encoder Representations from Transformers

After the introduction of the Transformer architecture, several large-scale language models have been introduced. One of the most influential models is the Bidirectional Encoder Representations from Transformer, abbreviated BERT (Devlin et al., 2018). As the name indicates, BERT is based on the Encoder from the Transformer architecture, with a modification that allows for bidirectional encoding of the input sequence. These representations can be pre-trained for general tasks and later fine-tuned by adding one or more additional output layers to create state-of-the-art models for various tasks. Devlin et al. also argue that the current language models have their major limitations by the use of unidirectional encoding, as done by Vaswani et al. (2017). By only paying attention to

Feature	**BERT$_{\text{BASE}}$**	**BERT$_{\text{LARGE}}$**
Layer	12	24
Self-attention heads	12	16
Hidden size	768	1024
Total parameters	110M	340M

Table 2.1: Comparison of the size of the BERT models.

previously seen input elements, lots of useful information may not be captured by the model, which is sub-optimal for sentence-level tasks.

Architecture and Pre-Training Tasks

The main components of the BERT model are layers of fully connected encoders from the Transformer architecture. Devlin et al. (2018) proposed two versions of BERT, BERT$_{\text{BASE}}$ and BERT$_{\text{LARGE}}$, where the difference lies in the size of the model, as shown in Table 2.1.

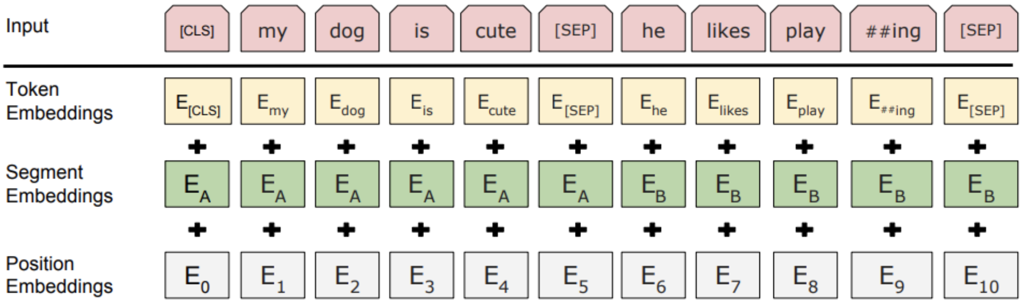


Figure 2.2: BERT input representations. Figure from Devlin et al. (2018), with permission from Jacob Devlin.

Two techniques are introduced to enable the pre-trained deep bidirectional representations. First, the masked language model, where tokens from the input sequence are randomly masked, and the objective is to predict the original element. Second, BERT uses a *next sentence prediction* (NSP) task for pre-training text-pair representations. These two pre-training tasks are performed on a dataset consisting of *the English Wikipedia* (2 500M words) and *BookCorpus* (800M words) (Zhu et al., 2015).

For BERT to handle the techniques mentioned above, it has to be able to represent both a single sentence and sentence pairs. How BERT represents the input are visualized in Figure 2.2. BERT uses WordPiece embeddings with a 30 000 token vocabulary to encode the input. For each input sequence, which may be either a single sentence or a pair of sentences, the model adds a special token [CLS]. If the sequence is a pair of sentences, another special token [SEP] is added between the sentences and at the end of sentence two. To each token, it is also added an embedding which indicates whether the token belongs to sentence A or sentence B. If the input sequence is a single sentence, every token is given the sentence A embedding. Finally, a position embedding is added to indicate the order of the input words.

The previously mentioned masked language model utilizes the token embeddings to allow training a bidirectional representation. 15 % of the input tokens are replaced by a masked token, [MASK], and then the model will predict the real value of the masked token. The prediction will then be based on the mask's context, obtained by fusing the left and right context and, hence, a bidirectional encoder. The drawback of this approach is that the masked token only appears in pre-training and thus creates a mismatch between pre-training and fine-tuning. To solve the mismatch, the masked word is not always replaced with a [MASK] token. 10 % of the time a random token is placed instead, 10 % the token is unchanged, and in the remaining 80% the [MASK] is used.

2.2.6 ALBERT - A Lite BERT

By increasing the model size for pre-trained models like BERT, the performance is often improved. Although higher performance, this increase comes with a cost. The size of models like $\text{BERT}_{\text{LARGE}}$ requires a huge amount of GPU/TPU memory, which are limited resources, and also increase the time to train the model. Lan et al. (2019) introduced A Light BERT - ALBERT - to address these limitations. The main architecture is similar to BERT (Devlin et al., 2018), but Lan et al. propose three innovations: factorized embedding parameterization, cross-layer parameter sharing and inter-sentence coherence loss. With factorized embedding parameterization the embedding parameters (with size E) and the hidden layer size H are decomposed from BERTs $O(V \times H)$ to $O(V \times E + E \times H)$, where V is the size of the vocabulary. By doing so, the number of parameters is significantly reduced when $H >> E$. The second innovation introduced for the means of parameter-reduction is cross-layer parameter sharing. ALBERT basically share all parameters across layers, both the attention parameters and feed-forward parameters. This leads to a significant drop in size, e.g., $\text{ALBERT}_{\text{LARGE}}$ has 18x fewer parameters than $\text{BERT}_{\text{LARGE}}$, without hurting the model's performance. ALBERT also removes BERTs NSP, which were proved unreliable by Yang et al. (2019) and Liu et al. (2019). NSP was supposed to target topic and coherence prediction, and ALBERT is trying to do this in a new way. Topic prediction is partly incorporated by MLM and therefore covered, and Lan et al. (2019) proposes to add sentence-order prediction loss to handle inter-sentence coherence prediction.

2.2.7 DistilBERT

DistilBERT (Sanh et al., 2019) is a distilled version of BERT. The motivation behind the development of the model was that the latest pre-trained language models in the field of NLP tended to become larger and larger, which affected both scalability and computational processing time. DistilBERT was thus developed to maintain the same level of performance as BERT, while at the same time being a much smaller pre-trained model, making it applicable for real-time problems. Where BERT$_{\text{BASE}}$ has 110M parameters, DistilBERT has decreased this by 40 %, to 66M parameters. Despite this radical decrease, DistilBERT retains 97 % of the performance and is 60 % faster than its parent model.

DistilBERT achieves this performance through knowledge distillation and exploiting BERTs knowledge when training. Knowledge distillation is a technique for compressing a large *teacher* model into a smaller compact model, a *student*, without significant performance losses. The student model is trained on the same dataset as the teacher, where the loss function is supervised by the teacher model's training loss, i.e., training the student to generalize the same way as the teacher by matching the output distribution. By this transfer learning approach, the student model can reproduce the teachers' behavior while being both lighter and faster.

2.2.8 RoBERTa - A Robustly Optimized BERT Pretraining Approach

The BERT architecture has shown to be invaluable innovation in the field of NLP, but the process of building these models is constantly improved. Liu et al. (2019) state that BERT is significantly undertrained and proposes RoBERTa, A Robustly Optimized BERT pre-training Approach, which improves the performance on several downstream tasks. Four modifications are suggested to improve the pre-training of the model: (1) training the model longer, with bigger batches, over more data; (2) removing the next sentence prediction objective; (3) training on longer sequences; and (4) dynamically changing the masking pattern applied to the training data.

Longer Training with More Data and Longer Sequences

BERT is pre-trained on *BookCorpus* and *the English Wikipedia*, a 16 GB dataset. Liu et al. (2019) found that BERT was undertrained and an increase of the amount of data in pre-training could improve performance. In addition to *BookCorpus* and *the English Wikipedia*, Liu et al. used *CC-NEWS*, a dataset consisting of 63M news articles; *OpenWebText*, an open-source dataset of web content extracted from URLs mentioned on Reddit; and *STORIES*, a dataset with text similar to the story-like style of Winograd schemas (Levesque et al., 2012). The resulting dataset has a size of 160 GB, in contrast to the 16 GB dataset used by Devlin et al. (2018) for BERT.

A larger batch size when training the model was also used. By increasing the batch size will both the perplexity of the MLM objective increase as well as the accuracy for the downstream tasks. Devlin et al. (2018) used a batch size of 256 sequences while Liu et al. increase the batch size to 8 000 sequences.

Previous models have been trained for 100 000 steps, a number Liu et al. increased significantly. Experiments with both 300 000 steps and 500 000 steps were conducted, where the latter performed best.

Removal of Next Sentence Prediction

The Next Sentence Prediction (NSP) was one of two pre-training techniques introduced in BERT, with an objective to make the model able to predict whether two input segments actually followed each other or were randomly put together. NSP was found to improve the performance of BERT. Liu et al. on the other hand, report that removing the NSP loss from training does not harm the model. A new technique for constructing the input is suggested, where each input is full contiguously sentences from the same document, such that the total length does not exceed the 512 token boundary. If the input is shorter than 512 tokens, the batch size is dynamically adjusted to achieve a similar number of tokens as the previously suggested approaches.

Dynamic Masking

The other pre-training technique introduced by Devlin et al. were masked language model (MLM). MLM masks tokens in the input sequence with the objective to predict what the original element was. The tokens that are chosen to be masked are randomly selected during the pre-processing of the data. To avoid this static selection of mask where the same token in each sequence is used for every training instance, the training data is duplicated so that each duplicated sequence is masked differently. Liu et al. propose a dynamic masking strategy where the masking pattern is generated every time a sequence is feed to the model.

2.2.9 BERTweet - A Pre-Trained Language Model for English Tweets

As BERT and BERT-like models like ALBERT and RoBERTa have revolutionized general NLP tasks, several studies have tried using the BERT architecture to make domain-specific models, as Lee et al. (2019) with BioBERT. They concluded that pre-training BERT on biomedical corpora helped it to understand complex biomedical texts and enhanced the performance obtained by state-of-the-art models on several tasks. Nguyen et al. (2020) did the same thing with the domain of short text on social media and created BERTweet.

BERTweet is a model with the same architecture as BERT (Devlin et al., 2018) and is pre-trained on a corpus of 850M English tweets, using the pre-training procedure of RoBERTa. The idea is that social media texts use informal grammar and are often shorter than traditional texts, e.g., tweets, which have a character limit of 280. Therefore, models like BERT, which is trained on text corpora with formal language, may have trouble when it comes to classifying such texts.

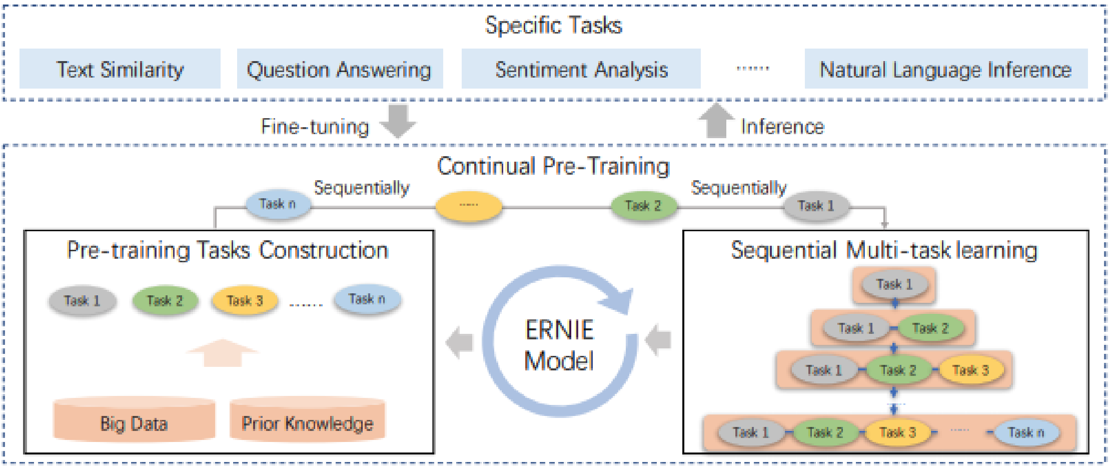


Figure 2.3: The framework of ERNIE 2.0. Figure from Sun et al. (2020), with permission from Yu Sun.

2.2.10 ERNIE 2.0: A Continual Pre-Training Framework for Language Understanding

ERNIE 2.0 is the successor of ERNIE (Enhanced Representation through kNowledge IntEgration) and is a continual pre-training framework taking advantage of multi-task learning to improve its existing knowledge base. Sun et al. (2020) indicate that the pre-training procedures of the state-of-the-art models do not exploit the full potential of the text in pre-training. While most models are trained with simple (as Sun et al. call it) tasks to find co-occurrence of words or sentences, ERNIE 2.0 also aims to grasp other lexical, syntactic, and semantic information in the text. By using two additional kinds of knowledge strategies, phrase-level and entity-level, tasks like named entity recognition and sentence proximity are enabled.

Continual Learning

Continual Learning is the process of training the model sequential with different tasks, with the objective that the model will use the previously learned task when learning the next (Parisi et al., 2019). The model will thus perform well when given a new task because it has some knowledge already. This concept can be thought of as when humans use previously acquired information when confronted with new tasks. ERNIE 2.0 uses continual learning, and combined with multi-task learning, the model can constantly take in new tasks to improve the understanding of lexical, syntactical, and semantic representations in the data. Figure 2.3 shows how continual and multi-task learning is combined. The continual multi-task learning process consists of two steps. First, the pre-training tasks are constructed with big data without the need of human annotators and prior knowledge, which, e.g., can be discourse, named entities, or known phrases.

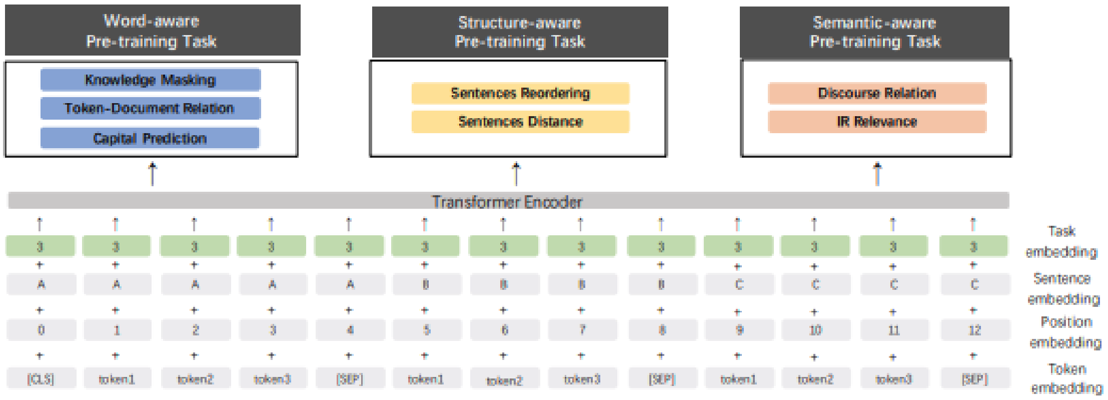


Figure 2.4: Input embeddings and pre-traning tasks for ERNIE 2.0. Figure from Sun et al. (2020), with permission from Yu Sun.

Second, the model is trained efficiently on the tasks from step 1, and the knowledge obtained is stored along with the knowledge learned from the previous steps. Sun et al. (2020) solve these two steps in the following way: When new tasks are feed to the model, it is initialized with the parameters learned from the previous steps. The tasks from the previous step are then combined with the newly arrived tasks, and lastly, the model trains on all the tasks simultaneously.

Pre-Training Tasks

Three tasks are proposed by Sun et al. (2020) to grasp different features in the training data: (1) word-aware, (2) structure-aware, and (3) semantic-aware pre-training tasks. An overview of the tasks and how the input is embedded is visualized in Figure 2.4.

Word-aware The word-aware pre-training tasks are meant to learn lexical information in the data and include the tasks knowledge masking, capitalization prediction, and token-document relation prediction. Knowledge masking comprehend basic-level masking, like the MLM introduced by Devlin et al. (2018) in BERT, phrase-level and entity-level masking. Capitalization prediction is the task of predicting whether the word is capitalized or not. Cased text may have other semantic information than others, e.g., if the word is a name. Lastly, the token-document relation prediction task trains the model to identify the key words of the document.

Structure-aware Structure-aware tasks train the model to capture syntactic information in the data and include the sentence reordering and sentence distance tasks. The former tries to make the model learn the relationships among sentences. The sentence distance task aims to learn the model if two sentences are adjacent, in the same document, or from different documents.

Semantic-aware The last pre-training tasks are the semantic-aware task, including the discourse relation task and the information retrieval (IR) relevance task. The discourse relation tasks aim to predict the rhetorical or semantic distance between two sentences. Lastly, the IR relevance task enables the model to predict the relationship between a query and a title.

2.3 Performance Measures

In general, it is important to measure how well the machine learning model performs on unseen data, as it may be applied to real-life tasks. Therefore it has been developed several measures with regards to **performance**, which in most cases is specific to the task being carried out by the machine learning system. To simulate a real-life situation, it is normal to split the dataset provided for the task into three parts: a training and validation set used for the training part of the task and a test set used in the test part. The training set is usually the biggest dataset and contains the data used for training the model. To continuously improve the model, its performance is measured using the validation set. The result of this evaluation is used to tune the model further. When the model has achieved satisfying results on the validation set, it can make predictions against the unseen test set.

With regards to measuring the performance of a classification model, the terms true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) describe the classifier's predictions. The classifier's prediction of an instance is indicated by the positive or negative terms, while true and false indicate whether that prediction is correct. In other words, a true positive (negative) is an outcome where the model correctly predicts the positive (negative) class, and a false positive (negative) is an outcome where the model incorrectly predicts the positive (negative) class. Table 2.2 shows the confusion matrix describing this relationship for binary classification. For the rest of this section, the performance measures used in this Master's Thesis will be described.

		Prediction	
		Positive	**Negative**
Actual	**Positive**	True positive	False negative
	Negative	False positive	True negative

Table 2.2: Confusion matrix for binary classification.

2.3.1 Accuracy

Accuracy is a simple and common measure for evaluating the performance of a classifier. The accuracy score for a model is the ratio of correct classifications out of the total number of data samples it was tested on, as shown in Equation 2.3. Although accuracy is a good performance measure for models, it has some drawbacks when the class distribution of the dataset is uneven, i.e., if 95 % of the data belongs to some class A, the model could predict every data sample to belong to class A and thus give a false sense of achieving an accuracy score of 95 %.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.3}$$

2.3.2 Precision and Recall

Precision and recall are two correlated performance measures that are popular in the field of information retrieval, natural language processing, and classification. Precision is a measure of how precise your classifier is and is defined as the fraction of how many true positives the model retrieved among all the predicted positives, as defined in Equation 2.4.

$$P = \frac{TP}{TP + FP} \tag{2.4}$$

Recall is a measure for how robust your classifier is and is defined as the fraction of how many true positives the model retrieved among the total number of actual positives, as defined in Equation 2.5.

$$R = \frac{TP}{TP + FN} \tag{2.5}$$

The two scores provide valuable insights into a classification model. If the model achieves high precision, most of the predicted positive instances were classified correctly. On the other side, if the model has a high recall, it has correctly predicted most of the actual positive instances. As mentioned, the scores are highly correlated. One could easily gain a perfect precision score by making the model only retrieve one instance that the system is sure about, but that would give a bad recall score. Similarly, the system could gain perfect recall by simply retrieving all the dataset instances, but that would give a bad precision score. The goal is thus to create a classifier model that scores high for both performance measures.

2.3.3 F1-score

The F_1-score is the harmonic mean of the precision and the recall performance measures, as defined in Equation 2.6. The best possible F_1-score is when both precision and recall are perfect, with the value of 1. Evaluating a classifier towards a measure combining precision and recall gives a more robust result.

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

As precision, recall, and F_1-score are calculated with regards to the positive label, they are suitable for binary classifications. When working with multiclass problems as in this Master's Thesis, there exist different approaches for calculating the average precision, recall, and F_1-score: micro, macro, weighted, and sample averaging. Micro averaging calculates the average on a global scale by calculating the sum of contributions of each class, while macro averaging will calculate the average for each class unweighted. Weighted averaging is similar to macro but is calculated with weights for each label, which can be beneficial when classes are uneven. The sample averaging will calculate the average of the score for each instance.

3 Text Representations and Annotation

This chapter will cover the theory of Natural Language Processing (NLP) and annotation that is not assumed to be known beforehand by the reader. The first section presents NLP and text representations, while the second section covers the concept of annotation. Most sections in this chapter were written during the preliminary study, and only minor changes are made to these sections. Additional contributions to this chapter are Section 3.1.8, 3.1.9, and 3.2.

3.1 Natural Language Processing and Text Representations

Natural language processing (NLP) is the field of programming computers to process, analyze, and understand human natural language data. As computers do not understand written text directly, one of the main challenges when dealing with textual data is numerically representing the text so that machine learning models can interpret it. This section presents relevant methods and concepts used in this report for text representations in NLP.

3.1.1 Processing Written Text

Before applying different text representation techniques to utilize written text as features in a machine learning model, the text may undergo some processing. This is because human written text tends to be grammatically and lexically wrong, especially in terms of social media communication. The most popular pre-processing steps are called stopword removal, tokenization, stemming, and lemmatization.

Stopword removal is a technique where the most frequent terms in a document are removed. This is done because such words usually only introduce noise in the data and do not tell anything about the text's content. Examples of English stopwords are *I*, *a*, and *the*. Tokenization is a technique where the text gets split into a sequence of characters, called tokens. Tokens can also be further processed by converting them into lowercase text, remove certain characters (like punctuation) and do spell-checks. Stemming and lemmatization are techniques where words of different grammatical forms are reduced into a common base form. Stemming is based on specific rules that reduce the word into its root form by removing the end of the word, e.g., converting *walks* and *walking* into the simple form *walk*. This way, all three occurrences of the word *walk* will be treated as the same word instead of three separate words. Lemmatization works a bit differently, as it uses a dictionary to retrieve the base word, called a lemma. An example is that the lemmatization process would convert *am*, *are* and *is* into the base form *be*.

3.1.2 N-grams

N-grams are a popular text processing technique defined as sequences of either n words or characters that appear next to each other in a text document. When $n = 1$, the 1-gram is called a unigram, which essentially is a single word or character. For 2-grams, it is called bigrams, 3-grams is trigrams, and so on. By using bigrams and trigrams, it is possible to find out how often certain words appear together. An example of a unigram representation on word-level from the text *I like being skinny* would be *I*, *like*, *being*, and *skinny*. The bigram representation for the same sentence would be *I like*, *like being*, and *being skinny*. Similarly, a unigram representation on character-level for the same sentence would result in each character being a unigram, while a character-level bigram would be *I*, *l*, *li*, *ik*, and so on.

N-gram models are simple and scalable and have proven to give good results on a variety of NLP tasks. However, one issue when using n-gram language models is out-of-vocabulary (OOV) words. When estimating a language model, the entire observed vocabulary is generally used. However, in some cases, it may be necessary to use a specific fixed vocabulary. In this case, the n-grams containing an OOV word are ignored, and the n-gram probabilities are smoothed over all the vocabulary words even if they were not observed. This can be handled by introducing a special token, such as <UNK>, that replaces the OOV words.

3.1.3 Bag of Words

In NLP, the text is usually stored in documents, and a collection of documents is commonly known as a corpus. The Bag of Words (BoW) model represents a set of every word appearing in the corpus. The model counts how many times each word appears in the text while ignoring the order and grammar of words. The process of creating feature vectors for documents with the BoW model includes creating a vocabulary of all unique terms in the collection of documents and then creating a feature vector per document with the count of occurrences for each term. The BoW model is commonly used in methods of document classification as a feature for training a classifier. An example of how BoW can be used is shown in the two following documents:

Document 1: I just want to be skinny.
Document 2: I started recovery because I was too skinny.

Terms	Document1	Document2
i	1	2
just	1	0
want	1	0
to	1	0
be	1	0
skinny	1	1
started	0	1
recovery	0	1
because	0	1
was	0	1
too	0	1

Table 3.1: Example of Bag-of-Words vector representations with two documents.

As seen in Table 3.1, all terms are listed, and vectors are created for each document. The values of the vectors are the number of times each word appears in the document.

3.1.4 Term Frequency-Inverse Document Frequency

Term Frequency-Inverse Document Frequency (TF-IDF) is a term weighting measurement that reflects the importance of terms in documents and is used in many text classification and information retrieval systems. Term frequency (TF) is a measurement for how often a term t occurs in a document d, defined as $TF(t,d) = \frac{f_{t,d}}{N}$, where N is the total number of documents. A higher TF score indicates that the word is important for the content of that document. Some words, like stopwords, might have a high TF in many documents, but this does not necessarily mean that the stopwords are important to the given document's content. Inverse document frequency (IDF) is a way to handle this, as it measures how much information the term provides. The IDF is based on how many documents that mention a word:

$$IDF = log(\frac{N}{1+n_t}) \tag{3.1}$$

N is the total number of documents, n_t is the number of documents that the term t has appeared in, and the $+1$ is a smoothing constant in case a term does not appear in any documents to avoid division by zero. A term with high IDF means that it appears in a few selected documents and has high discriminative power. If a term has a low IDF value, the term is not unique in the collection of documents and provides little discriminating power. The TF-IDF score is calculated by multiplying the term frequency and the inverse document frequency:

$$TFIDF = TF * IDF \tag{3.2}$$

3.1.5 Word Embedding

Word embedding is a collective term for any NLP technique where words are mapped to vectors of real numbers. The concept is that words with similar meanings will have similar vector representations, inspired by the theory of the distributional hypothesis (Harris, 1954). Each word is mapped from a higher-dimensional space into a vector in a lower-dimensional space. The vector values are learned using a neural network (see Section 2.2.1). This approach overcomes many of the problems that models like BoW have, such as sparseness, high dimensionality, and that similar words do not have similar representations. Words with similar meanings will be closer to each other in the vector space, and both the distance and direction of the vectors can encode semantics in a useful embedding. Another advantage of this technique is that it is possible to do calculations on the vectors. A classic example for describing word embeddings is the equation $king - man + woman = queen$. This represents a gender relationship in the word embeddings, as adding the vectors associated with the words *king* and *woman* while subtracting *man* is equal to the vector associated with *queen*.

3.1.6 Topic Modeling

Topic modeling is an unsupervised statistical model for discovering abstract topics that occur in a collection of documents. A topic is defined as a cluster of words that frequently occur together in the documents. It is unsupervised because it does not require predefined topics; only the number of topics to be computed is defined. A topic model clusters words and expressions that appear most often by detecting patterns like word frequency and distance between words. Topic modeling can measure how much of the document belongs to each topic if several topics are present in the document. It can also discover hidden topics in the text, like semantic structures that are not obvious to humans.

The most popular method for topic modeling is called Latent Dirichlet Allocation (LDA) and is based on the distributional hypothesis, meaning that linguistic items with similar distributions have similar meanings. Each document is described by a distribution of topics, and each topic is described by a distribution of words. LDA ignores syntactic information and treats documents as bags of words while also assuming that all words in the document can be assigned a probability of belonging to a topic. The algorithm first assigns a random topic t to each word w. Then it goes through each document d and determines the probabilities $P_1 = P(t|d)$ and $P_2 = P(w|t)$ for each word in the document. The word is updated with the probability $P_1 * P_2$ for the assignment given, and this process continues for a given number of iterations or until convergence.

3.1.7 WordPiece Tokenization

WordPiece (Schuster and Nakajima, 2012) is a sub-word tokenization algorithm used for training BERT and DistilBERT (among others). The text is split into words, which then is split into word pieces. The approach achieves a balance between the flexibility of characters and the efficiency of words. Special symbols (like '_' to mark the beginning of a word) are added to the word pieces, making it easy to decode the word pieces back to the original text. The approach is data-driven and guaranteed to generate a deterministic segmentation for any possible sequence of characters. Given a training corpus and a certain number of desired tokens D, the optimization problem is to select D word-pieces such that the resulting corpus is minimal in the number of word-pieces when segmented according to the chosen word-piece model. WordPiece has a greedy approach and will, at each iterative step, choose the word-piece pair that will result in the largest increase in likelihood on the data once merged. WordPiece handles OOV words well, as such words will be broken down into sub-word units using this segmentation technique.

3.1.8 Byte-Pair Encoding Tokenization

Byte-Pair Encoding (BPE) tokenization (Sennrich et al., 2015) is another sub-word segmentation algorithm used by several attention-based models and was the inspiration behind the WordPiece algorithm. The algorithm relies on a pre-tokenizer that splits the text into words and counts their frequencies. After this, a base vocabulary is established by splitting all the words into single symbols. BPE then follows a greedy approach by merging the symbol pairs that occur most frequently in an iterative manner until a fixed vocabulary size is reached. The algorithm preserves a balance between representations of both characters and words, making it capable of managing large corpora and OOV words.

3.1.9 Sentiment Analysis

Texts usually have an attitude towards or opinion about something, which is a feature that can be exploited when processing text. Sentiment analysis, or opinion mining, aims to identify and extract this attitude (or sentiment) of a given text or topic. One of the most well-known typed of sentiment analysis is to find the polarity of the text, where the text will be categorized to have either a positive, neutral, or negative sentiment. A sentiment analysis can be performed using a rule-based system, an automatic system, or a combination, where various NLP and machine learning techniques are used, or combined, and applied to the data to perform the analysis.

3.2 Annotation

Supervised learning requires the training data to have some sort of labels or tags that augment the information in each data instance in a desirable way in order to train a model that can predict the label of new data instances (from Section 2.1.1). In cases where the dataset does not come with labels, they need to be extracted from the data instances, a process called annotation. Annotation can be done automatically or manually, or somewhere in between, and which method is chosen often depends on the nature of the task and the resources available.

3.2.1 Automatic Annotation

Automatic annotation is the process where a computer assigns to a data instance the most likely correct label. It is often used when there is a lot of data to analyze or the data is too complicated for humans to understand. To make the computer able to annotate the data, it needs to know how to predict the label of a data instance. One way to teach the computer how to annotate is the case of supervised learning, and another way is to predetermine a set of rules for the computer to follow. Either way comes with a degree of error, and the annotation's reliability depends on the similarity between the training data and the new data to be annotated.

3.2.2 Manual Annotation

Manual annotation is another annotation method that involves humans reviewing the data instances and assigning a label. For classification tasks, the annotator assigns one of multiple predetermined labels based on the annotator's interpretation of the data instance. Since the annotator needs to carefully evaluate each data instance in the dataset, manual annotation becomes expensive and time-consuming. Therefore, manual annotation is often only used to the point where there is enough data for a computer to do the same task with a satisfying performance.

As manual annotation is based on the interpretation of a human being, there is a certain chance of bias and mistakes. To reduce this uncertainty, manual annotation is often done by multiple annotators. If one of the annotators is biased in some way, this will be discovered through different labels for the same data instance. To evaluate the agreement between two or more annotators, inter-annotator agreement metrics can be used. An inter-annotator agreement metric measures to which degree the annotators assign the same label to the same data instance. The measure reveals if the annotators are on the same page regarding the boundaries between the class labels and indicate how reliable the annotated data is.

3.2.3 Cohen's Kappa

Cohen's Kappa is a measure of agreement between two annotators for nominal scales and was introduced by Cohen (1960). Cohen described the simple measure of agreement between two annotators where the proportion of cases in which the annotators agree are counted, and how this does not take into account that the annotators may agree (or disagree) by chance. The following formula was proposed to solve the issue of agreement by chance:

$$\kappa = \frac{p_0 - p_\mathrm{e}}{1 - p_\mathrm{e}} \tag{3.3}$$

where

p_0 is the relative observed agreement between the annotators.

p_e is the probability of agreement by chance.

The Cohen Kappa coefficient ranges from -1 to 1, where 0 represents the agreement of pure chance and 1 represents perfect agreement between the annotators. Values below zero are unlikely and indicate no agreement. Table 3.2 shows an interpretation of the Cohen Kappa coefficient (Landis and Koch, 1977), in terms of the strength of agreement between the annotators.

Kappa	**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*

Table 3.2: Interpretation of Cohen's Kappa (Landis and Koch, 1977).

Although κ is a suitable measurement for inter-annotation agreement, it has limitations. First, it only measures agreement between two annotators. Therefore, another metric has to be used if there are more than two annotators, e.g., Fleiss Kappa.[1] Second, κ tends to underestimate the agreement with the presence of rare categories (Viera and Garrett, 2005). Third, all disagreements are treated the same, and κ will not take into consideration if some of the categories are closer to one another than others.

[1] https://psycnet.apa.org/record/1972-05083-001

4 Related Work

This chapter includes an overview of research related to this Master's Thesis. The first part of this chapter looks at previous research is done on online pro-ED communities, while the second part looks at research in the field of user classification on social media platforms. The third section includes recent research on how attention-based methods have been applied to social media data tasks and how these methods perform against standard machine learning methods. Lastly, research related to collecting and annotating data from social media is presented. The three first sections in this chapter were written during the preliminary study, and are updated with further research for this Master's Thesis. An additional contribution to this chapter is Section 4.4.

4.1 Studies on Online Pro-Eating Disorder Communities

The amount of research related to online pro-ED communities has grown in the last two decades after the growth in popularity of online websites and the use of social media. The studies mainly focus on content analysis (Borzekowski et al., 2010; Branley and Covey, 2017; Cavazos-Rehg et al., 2019), user behavior (Whitehead, 2010; Boero and Pascoe, 2012) and the impact of being exposed to pro-ED content (Wilson et al., 2006; Talbot, 2010; Delforterie et al., 2014; Bardone-Cone and Cass, 2007).

Arseniev-Koehler et al. (2016) investigate pro-ED socialization on Twitter and how users reference EDs. They collected information about 45 pro-ED users regarding profile information, tweets, and followers, which resulted in a total of 4 245 tweets. From this, they developed a codebook to evaluate ED references in tweets and profiles. The codebook consisted of 143 keywords in total, grouped into nine categories. A sample of these categories and words can be found in Table 4.1. Many of the same keywords were found in Cavazos-Rehg et al. (2019), who also examine ED and body image-related content on Twitter, with *thinspiration* or abbreviations of the word (*thinspo*, *thinspos*) being the most popular one. They looked at a random sample of 3000 tweets and found that 65 % expressed a preoccupation with body shape, whereas 51 % of these included an image.

Arseniev-Koehler et al. (2016) also provide three criteria for a user to be defined as pro-ED. A user was defined as pro-ED if one of the following criteria was fulfilled; 1) self-identifying as pro-ED and/or as having an ED and anti-recovery, or 2) expressing a desire for emaciation, or 3) ascribing to a pro-ED event. These criteria were further

Sample domains	Sample keywords
Eating Disorder	anorexic, ana, bulimic, proana, proed, promia, ednos, eating disorder, wannarexic
Body Image & Weight	overweight, obese, fatty, skinnier, skeleton, emaciated, hipbones, backbone, bones, collarbone, thighgap, bikinibridge, hipbones, thighs, hips, thinspo, bonespo, perfection, weight, scale, gw, cw, lw, ugw, bmi, poun
Food and Meals	calorie, food, breakfast, dinner, meals, eating, eat, ate, appetite, starve, hunger, diet, skip, fasted, calorieapril, projectthin, rg, abcdiet, binge, bloate
Compensatory behaviour	laxies, laxatives, vomited, throwup, puke, purge, workout, abs, jog, elliptical, exercise, miles, gym, treadmill

Table 4.1: Sample of ED Reference Codebook from Arseniev-Koehler et al. (2016).

adapted in Giæver (2018) and expanded to include a fourth criteria; 4) encourage extreme weight control methods.

Stewart et al. (2017) studiy lexical and orthographic variation in pro-ED terms on Instagram, following their ban of hashtags like #thinspiration in 2012 (Instagram, 2012). These hashtags grew more popular and more complex, becoming increasingly distant from the original spellings to circumvent the ban. Examples of this include the usages of hashtags like #thygap instead of #thighgap, or #anarexia for #anorexia. Compared to Twitter, which did not introduce a ban of tags (and still have not), the use of orthographic and lexical variations was much higher on Instagram: 51.9 % of the pro-ED Instagram posts contained at least one variant, while on Twitter, only 15.0 % of pro-ED posts did. The study also concluded that committed newcomers in the communities led the change towards increased orthographic variation.

Borzekowski et al. (2010) studiy 180 pro-ED websites and found that 85 % of the websites displayed thinspiration material, while 83 % provided suggestions on how to engage in eating-disordered behaviors, such as fasting and vomiting. Another study collected 222 thinspiration images and posts from Tumblr and found that 97.7 % of the images included a thin woman as the subject and that dieting and losing weight were the most common themes in the text posts (Wick and Harriger, 2018).

Borzekowski et al. (2010) also discovered that approximately one-third of the sites had a statement directed to *wannabes*. These statements could both be polite and hostile but were made with the same goal of protecting the community and chase away site visitors that were not *true* anorexic. The findings are supported in Boero and Pascoe (2012), where they looked at 14 online discussion groups on MySpace. They defined these *wannabes* as *wannarexic* - people who want to participate in the community but whose credibility as eating disordered is in doubt. They found that the pro-ED users consider wannarexics a treat to the community and thus tries to expose them. The community would then act aggressively when they discovered such persons. The study found that certain rituals were conducted to protect the community: check-ins (including statistics and food reports), posting pictures, and group activities (fasts and surveys). Another study describes the pro-ED community as a form of collective identity for the people in the community and that the members create boundaries by stating that the websites are for those who already have an eating disorder, not the ones who are trying to *catch* one (Whitehead, 2010).

Several studies focus on the difference between pro-ED and pro-recovery content. For example, Branley and Covey (2017) compare how people communicate about EDs on Twitter and Tumblr, gathering 12 000 tweets and 73 000 tumblogs. They looked at three categories (pro-ana, anti-ana, and pro-recovery) and found that pro-ana posts were more common on Twitter than Tumblr, while Tumblr had more anti-ana and pro-recovery posts. Other findings from the research showed that pro-ana posts displayed EDs as a lifestyle choice, with users sharing motivational material like images and quotes such as *keep going, nothing tastes as good as skinny feels.* In contrast, pro-recovery material included posts from people who underwent recovery themselves, sharing their recovery progress and struggles. Other pro-recovery users offered support in the form of empathy, compassion, and understanding.

Yom-Tov et al. (2012) investigate why pro-recovery users post their content and if this facilitates the recovery of pro-anorexia users. The study suggests that pro-recovery users employ similar words as those used by pro-ana users to describe their photos so that their content would more likely end up in the *feed* of a pro-ana user. Pro-recovery users also comment on pro-ana content, but the findings suggest that this was counterproductive, as it entrenches pro-ana users in their stance. The findings of Fettach and Benhiba (2019) support this: they looked at topics used by pro-ED and pro-recovery communities on Reddit and found that the two share common topics like binge eating and thinspiration. Chancellor et al. (2016a) examine the effect that pro-recovery content had on users and found that only half of the cohort they studied was estimated to experience recovery in approximately four years.

Several studies investigate the impact of watching pro-ED content. Bardone-Cone and Cass (2007) gathered 235 female undergraduates and displayed either pro-ana websites or control group websites related to either female fashion or home decor to the participants. They found that participants exposed to the pro-anorexia websites had a more significant negative effect, lower social self-esteem, and lower appearance self-efficacy post-website than those who viewed either of the comparison websites. Pro-anorexia website viewers also perceived themselves as heavier post-website than viewers of the other websites. Participants also reported that viewing the websites made them more likely to exercise and think about their weight in the near future. Another study asked patients and the families of patients to look at pro-ED websites and found that 96 % reported learning new weight loss or purging techniques, while 69 % reported using the new techniques as a result of viewing the websites (Wilson et al., 2006).

Harper et al. (2008) conduct a similar study on how viewing pro-ED websites would affect body dissatisfaction and eating disturbance, where 1 575 women looked at either pro-ED websites, professional sites containing clinical information about EDs, or sites unrelated to EDs (control group). Their findings show that individuals who frequented pro-eating disorder sites had higher body dissatisfaction and eating disturbance levels than the control group. Despite this, there was limited evidence that those who view pro-ED sites differ from individuals who view professional sites offering information regarding eating disorders. On the other hand, Delforterie et al. (2014) conduct a similar study but found that women who viewed the pro-ana websites did not differ from those in the control group, when looking at the measures affect, appearance self-efficacy, and body satisfaction. One body satisfaction measure even showed that pro-ana viewers tended to be more satisfied with their bodies than home decoration viewers. The study explained that this could result from downward social comparison processes, where individuals compare themselves to someone they believe is worse off than themselves, which has been found to affect body satisfaction in women positively. Further, they discuss that women might view these websites because of pre-existing body dissatisfaction. Thus, it may be essential to focus on targeted prevention on these sites in the future.

4.2 Classification of Eating Disorder Users in Social Media

This study is based on the findings of Giæver (2018) and Nornes and Gran (2019), who both created systems to classify pro-ED users on Twitter. Giæver (2018) investigates several machine learning models, using unigrams of tweets, bigrams of tweets, emojis, Twitter username and display name, and Twitter biographies (bio) as features. The findings show that unigrams and bigrams from tweets were the most influential features but that all the features affected the score in general. Another finding showed that bios could include important information, as many pro-ED users fill their bio with references to body shape and weight. Examples of this are *sw* (start weight), *cw* (current weight), *gw* (goal weight), and *tw* (trigger warning, to warn other users of their content). The machine learning algorithms explored were the Support Vector Machine, Naïve Bayes, Logistic Regression, and Random Forest. The best model on the test dataset from this experiment was the SVM and a voting classifier consisting of all four models, using weighted feature groups on a binary classification task (pro-ED or non pro-ED).

Nornes and Gran (2019) continue the work of Giæver by exploring more features on the same but slightly more processed dataset. Their findings include using a machine learning model trained on The Big 5 Personality Traits (Tupes and Christal, 1961) as a feature, along with unigrams, bigrams, and topic modeling. Just like Giæver, the study also emphasizes that the Twitter bio could be a useful feature, as pro-recovery users often state that they are pro-recovery and where to get help in their bio. Several machine learning algorithms were tested, with Multilayer Perceptron achieving the best score when personality was included as a feature, thus improving the state-of-the-art system introduced by Giæver.

One relevant shared task for the same subject is the 2018 *Early risk prediction on the Internet* (eRisk 2018) task by Losada et al. (2018). The study involved creating a system to find users at risk of developing anorexia based on their Reddit posts over time. One of the participanting systems worth mentioning is that by Ramírez-Cifuentes et al. (2018), who further establish SVM as a great model for social media text classification. The study utilizes TF-IDF vectors based on n-grams, topic modeling, and linguistic information as features. Their findings show that the SVM outperforms both Logistic Regression, Random Forest, and Multilayer Perceptron models.

The 2019 edition of the same workshop (eRisk 2019) introduce the same task of early detection of anorexia in Reddit data, along with detection of self-harm and the severity of depression (Losada et al., 2019). The participants that achieved the highest F_1-score on the anorexia task were Mohammadi et al. (2019). They applied several neural sub-models and user-level attention to create a user representation, which they then fed into an SVM classifier to perform the classification.

4.3 Attention-Based Classification of Social Media Text

With the introduction of Attention and Transformer (see Section 2.2.3 and 2.2.4), several studies have been conducted to use such methods for NLP tasks on social media texts. There are no studies showing how attention-based techniques perform on the task of classifying pro-ED content in social media. Therefore, this section will cover relevant studies for the use of attention-based techniques on various classification tasks using social media text.

One of the more relevant contributions to social media text classification in recent times is the 2019 *Social Media Mining for Health Application* (SMM4H) shared task (Weissenbacher and Gonzalez-Hernandez, 2019). They had 34 teams look at four different tasks related to adverse drug reactions in tweets. The shared task's results show that systems implementing BERT achieved the highest score on all of the four tasks.

The first task was a binary classification task focusing on detecting whether a tweet mentions adverse drug reactions or not. The best scoring study achieved an F_1-score of 0.6457, and used the tweets as input to a BERT model, and then applied a softmax layer to classify the tweet (Chen et al., 2019). It is also worth mentioning that all top five systems for this task utilized BERT in some part of the system. Another interesting finding is that the same task was used in the 2018 *Social Media Mining for Health Application* (SMM4H) shared task (Gonzalez-Hernandez et al., 2018). This was around the time that BERT was released and before it was widely adopted in the NLP community. None of the participants tried BERT at the time, but Attention showed its strengths either way, with the best system being based on multi-head self-attention, achieving an F_1-score of 0.5220 (Wu et al., 2018). When comparing the results from the 2018 SMM4H to the 2019 SMM4H, the difference in F_1-score indicates that BERT performs good on social media Twitter data.

Another interesting and relevant task from the 2019 SMM4H is the fourth task, which investigated classification of personal mentions of health in tweets. The aim of this task was to create the most generalized system, as the system did training on one domain (influenza) and testing on another (Zika virus). The best scoring system was proposed by Ellendorff et al. (2019) and included an ensemble of different fine-tuned BERT models (BERT-$_{\text{BASE}}$ uncased).

Also, the 2020 *Social Media Mining for Health Application* shared task reports interesting results (Klein et al., 2020). Task 1 was to classify tweets mentioning medications, which also appeared in the 2018 SMM4H (Gonzalez-Hernandez et al., 2018), but then with a balanced dataset. The 2020 edition used the previously mention dataset from SMM4H 2019, which was highly unbalanced (Weissenbacher and Gonzalez-Hernandez, 2019). Nine out of the top ten teams used BERT, RoBERTa or domain-specific versions of BERT, and the three best model outperformed the baseline attention-based model from the 2019 SMM4H, which had a F_1-score of 0.788. Dang et al. (2020) were ranked as number one with a F_1-score of 0.85 and solved the task with an ensemble of BERT-like models, one of them an extension of the domain-specific Bio-BERT. The runner-up achieved a F_1-score of 0.80 with a pre-trained RoBERTa that was fine-tuned on corpora suited to the task (Casola and Lavelli, 2020).

Another study looked closer at the performance of attention-based models pre-trained on both specific domains and specific data sources. Guo et al. (2020) conducted a benchmark experiment where attention-based pre-trained models were tested on 25 social media text classification datasets. The models that were included in the research was RoBERTa (Liu et al., 2019), BERTweet (Nguyen et al., 2020) and ClinicalBioBERT (Alsentzer et al., 2019), which is a BERT model pre-trained on biomedical and clinical text. Their experiments found that RoBERTa$_{\text{BASE}}$ and BERTweet performed better than the ClinicalBioBERT, even on health-related tasks. BERTweet achieved the highest accuracies on 16 out of 25 datasets, including health and non-health-related datasets from both Twitter, Facebook, Reddit, and YouTube. This suggests that BERTweet can learn universal characteristics of social media languages by pre-training on tweets only. However, there were no significant differences between RoBERTa and BERTweet on most datasets in the experiments, which shows that RoBERTa can perform well on social media tasks. They concluded that for social media datasets built for health-related tasks, it might be better to choose a source-specific pre-trained model (like BERTweet for social media) rather than a domain-specific one. They also suggest that incorporating data from multiple social media platforms in the pre-training would possibly further improve the performance of BERTweet.

The 2019 *Natural Language Processing for Internet Freedom: Censorship, Disinformation, and Propaganda* shared task (NLP4IF) (Feldman et al., 2019) also has interesting findings. The task's goal was to classify sentences in news articles and whether or not they contained propaganda. For this, nine out of ten submissions applied BERT, using either the model itself or the embeddings, stating that this is one of the go-to approaches for solving recent NLP tasks. The highest F_1-score on the sentence level classification task was achieved by

Mapes et al. (2019), who substituted the final layer of their BERT model with a linear real-valued output neuron from a layer of softmaxes.

González-Carvajal and Garrido-Merchán (2020) conduct an extensive experiment to compare BERT against classical machine learning approaches on four different NLP tasks; IMDB movie review sentiment analysis, RealOrNot tweet classification, thematic foreign language (Portuguese) news classification, and Chinese hotel reviews sentiment analysis. In all four experiments, BERT was the superior model. Although not all of these tasks include classification of textual data from social media, the study is still relevant in showing that BERT outperforms classical machine learning models on several tasks.

Isaksen (2019) explores how BERT performs on a multiclass task of hate speech detection using two datasets from Twitter. The study examines four different BERT models, where two of them are further pre-trained on domain-specific data. The findings concluded that the pre-trained BERT model performed better than other machine learning algorithms like SVM on the same task. Another study also examines the use of BERT for hate speech detection in social media (Mozafari et al., 2019). Their best performing model profits the syntactical and contextual information embedded in different transformer encoder layers of the BERT model.

Another study that is highly related to hate speech detection is Davidson et al. (2020). They investigated how a BERT and DistilBERT model perform on the task of automatically identifying incivility in social media using data from both Reddit and Twitter. The study first pre-trained both models on 3 million Reddit posts and then fine-tuned them both for classification. The results show that the DistilBERT model achieves the highest F_1-score of the two. They use this DistilBERT model further to automatically classify a large collection of Reddit posts, creating synthetic data, used for training of a logistic regression model based on TF-IDF vectors created from unigram, that showed competitive results with the BERT and DistilBERT model.

The Second Workshop on Trolling, Aggression and Cyberbullying presented the task of *aggresion identification on social media* in English, Bengali, and Hindi, where the provided dataset was gathered from YouTube comments, 5 000 instances for each language (Kumar et al., 2020). Risch and Krestel (2020) submitted an ensemble of BERT models based on bagging (bootstrap aggregation), which outperformed all of the 19 participating teams on the task of identifying aggression in social media for English text. Risch and Krestel report that the motivation behind creating an ensemble was the instability of the classification performance across different fine-tuning runs of the same model. The ensemble consisted of several BERT models that were trained with different random weight initializations and dataset splits and found that this increased the performance substantially when the number of models in the ensemble exceeded ten. Devlin et al. also report variance in performance during fine-tuning, where variations between 84 % and 88 % for the same pre-training checkpoints are observed.[1]

[1] Link to findings on Google's Github `https://tinyurl.com/bertgithub`

Dodge et al. (2020) also address the matter of instability during fine-tuning of attention-based models. They fine-tuned BERT hundreds of times while varying the random seed of both the final classification layer of the fine-tuning process and the order of the training data. They found that trying different configurations of these two contributed comparably to the variance of the performance. They also found that for small datasets, many fine-tuning trials diverge part of the way through training.

4.4 Collection and Annotation of Data from Social Media

A characterization study on online eating disorder communities was conducted by Choudhury (2015), and a dataset concerning pro-anorexia when collecting the dataset was a manual exploration of Tumblr posts mentioning common eating disorder terms. This exploration was based on snowball sampling and resulted in 28 terms related to eating disorders. Further, a crawl of one week of Tumblr posts including these terms was conducted. From these posts, words that frequently co-occurred with the 28 eating disorder terms were stored, and after filtering out generic terms, like *fat* and *food*, 72 words (or tags) remained. These tags were then used to collect posts for the dataset. A separation between posts considered as pro-anorexia and pro-recovery was carried out. By examining the tags used to collect the dataset and the words that co-occur with them, a new set of tags considered as pro-recovery was made and used to distinguish pro-anorexia posts from pro-recovery posts.

Choudhury did also feature in Chancellor et al. (2016b), a study of content moderation and lexical variations in pro-eating disordered communities on Instagram. The same approach as in Choudhury (2015) was used for collecting data for the research.

Wang et al. (2017) conducted a study of homophily among pro-ED communities on social media and collected a dataset consisting of 3 380 Twitter users that are considered having a positive attitude towards eating disorders. To find users who self-identify as being diagnosed with an eating disorder, Wang et al. (2017) also used a similar approach as Choudhury (2015). Wang et al. focused on Twitter users who included terms which are semantically close to *eating disorder*, like *anorexia*, *proana* or *thinspo*. The resulting set of terms included several generic words like *food* and *fat*, and abbreviations of eating disorders (*ed* may denote the past-tense suffix of verbs), which were removed to avoid noise in the dataset. Thereafter, a search with the keywords through the Twitter Public API was conducted, and all words that had a high co-occurrence with the ED-terms were added to the set of keywords. As users who are considered pro-recovery also use many of these keywords, another constraint was added. To be considered an ED-positive user, the profile biography should additionally include biological information such as body weight or weight goals.

The same approach for data collection was later used when the same research group (Wang et al.) explored social interactions in online eating disorder communities (Wang et al., 2018). After the collection, 1 000 users were selected for annotation to ensure the quality of the data. 95.2 % of the users were considered highly likely to have an eating disorder.

Giæver (2018) conducted a study with the goal of automatic detection of users taking part in pro-ED communities and collected a dataset from Twitter in order to do so. The procedure for collecting the data was based on a set of search codewords and criteria. In order to identify pro-ED users, common pro-ED hashtags and terms were used in a search on Twitter to find relevant users and tweets. For each user collected, the username, display name, biography, location, date of account creation, and up to 3200 of the user's most recent tweets and retweets were collected, along with the publishing time and date for each tweet.

To establish the characteristics of a pro-ED user, Giæver defined four criteria. The first three of these criteria were inspired by Arseniev-Koehler et al. (2016), while the fourth was a contribution of her own. A user was considered pro-ED if they:

1. Included a self-identification as pro-ED, or
2. Expressed a desire for emaciation, or
3. Ascribed a pro-ED event, or
4. Encouraged extreme weight control methods

In order to label a user as pro-ED, satisfying *one* of the criteria in either the profile information, tweets, or retweets was enough. The annotation is therefore considered to be performed on **user-level**, meaning that if a user was labeled as the class *pro-ED*, all tweets written by that user would be labeled as the same class automatically.

When deciding whether a user was pro-recovery or unrelated, Giæver used another approach. Users that discussed eating disorders or related topics with a recovery or health-related focus, and did not satisfy any of the criteria mentioned above, were considered pro-recovery in the dataset. All remaining users at this point were labeled unrelated.

Nornes and Gran (2019) used the dataset collected by Giæver in their study on classification of pro-ED users on Twitter. They decided to do another round of pre-processing and re-annotation for their Master's Thesis. This was done due to minor errors found in the dataset and because most of the data was only annotated by one person. Nornes and Gran divided the dataset with Twitter users in two, where the sets had 1 000 overlapping users, and randomly collected 200 tweets from each user in the dataset. As Giæver previously had annotated the whole dataset, 1 000 users were now annotated by three annotators and the rest by two annotators (either Nornes or Gran, and Giæver). Cohen's Kappa (see Section 3.2.3) was used to calculate the inter-annotator agreement between Nornes and Gran (2019), and gave a score of 0.98. Fleiss' Kappa was used for the inter-annotator agreement for the 1 000 users annotated by the three annotators and gave a score of 0.96.

Table 4.2 includes an overview of the dataset and the different classes after re-annotation and another round of pre-processing conducted by Nornes and Gran. The result of these processes is the version of the dataset which was used in the preliminary study for this Master's Thesis (see Chapter 6 Page 77). For simplicity, this dataset will from now on be referred to as the Giæver dataset, as it was Giæver who collected the data and established the annotation criteria in the first place.

Label	**Number of Users**	**Number of Tweets**
Pro-ED	2 293	2 384 068
Pro-recovery	675	1 019 780
Unrelated	3 856	6 417 259
Total	**6 824**	**9 821 107**

Table 4.2: Distribution of users in pro-ED dataset from Nornes and Gran.

When annotating data, the annotators can disagree upon the label of the data they are annotating. Voutilainen (1999) investigated the inter agreement between two linguists who analyzed and tagged texts and established three reasons for why the linguistics in the research had different analyzes of the texts they disagreed upon. These three reasons are described below:

1. Inattention on the part of one linguists (as a result of which a correct unique analysis was jointly agreed upon).
2. Joint uncertainty about the correct analysis (both linguists feel unsure about the correct analysis).
3. Conflicting opinions about the correct analysis (both linguists have a strong but different opinion about the correct analysis).

Voutilainen (1999) reported an agreement of 99.8 % for the analyzed text and found that the main reason for disagreement was "slip of attention". One of the linguists admitted that "most of the job seemed too much of a routine to keep one mentally alert enough".

5 Data

In order to explore how attention-based models perform on the task of identifying pro-ED posts on social media, it is essential to have relevant and sufficient data available. In this study, a new pro-ED dataset was collected from Twitter, and a smaller Reddit dataset was collected for testing purposes. These two datasets were manually annotated. Additionally, a larger semi-automatically annotated Twitter dataset was established. The first part of this chapter discusses the motivation behind collecting new pro-ED datasets, while the second part introduces the annotation criteria that were established. The next sections explain how the datasets were collected, pre-processed, and annotated. Lastly, this chapter will cover characteristics found in the new datasets and a comparison between them.

5.1 Motivation for Data Collection

The following section will introduce the motivation behind collecting a new pro-ED dataset. The preliminary study (presented in Chapter 6) unveiled that the dataset discussed in Section 4.4 was not ideal for the experiments that were to be conducted in this Master's Thesis. It was therefore decided to collect new data, and the motivation is mainly based on three arguments.

The first argument considers the discussion about annotations on **user-level** versus **post-level**. As mentioned in Section 4.4, to label a user as pro-ED in the Giæver (2018) dataset, satisfying *one* of the criteria in either the profile information, tweets, or retweets was enough. With this approach, there is a theoretical possibility that a user has one single tweet that is considered pro-ED, while the rest of the tweets could be unrelated. This could negatively impact the machine learning model, as it is mostly trained on unrelated text, but it is annotated as being pro-ED. To avoid this, it was considered that a dataset annotated on post-level would give a more precise result and a better training basis for the model.

The second argument for collecting new data was the motivation for using data from other social media platforms than Twitter, such as Reddit. It was considered that the posts from these platforms would differ in various characteristics, such as length of posts, the formality of language, and usage of slang, hashtags, emojis, and URLs. These differences could be used to create a more general model for classifying pro-ED posts than previous models trained on data from Twitter only. As it was decided that annotations on post-level were preferable in this Master's Thesis, there would be challenges related to using the Giæver (2018) dataset in combination with new data from Reddit. For instance, there would be a difference in which pre-processing steps the data went through. If the datasets are not comparable, it would prove difficult to analyze the results of models trained on the different datasets and compare them to each other.

The third argument concerns that the data should be fully exploited by the attention-based models explored in this Master's Thesis. The preliminary study showed that when aggregating all tweets from one user into one long text string per user, most of the text was truncated during tokenization in the attention-based models. This was because the models usually have a fixed sequence length. As can be seen in Table 6.4 (Page 81), $97 - 98\%$ of the users in the Giæver (2018) dataset had text truncated during tokenization when the maximum sequence length was set to 512 for the long text string. With a new dataset annotated on post-level, the data loss from truncation of large text strings will be avoided because the aggregation of data is unnecessary.

5.2 Annotation Criteria

Collecting and manually annotating a dataset for the purpose of supervised learning requires a set of annotation criteria. For this study, the data was annotated into three classes; *pro-ED*, *pro-recovery* or *unrelated*. The understanding of what is to be considered as a pro-ED post might be different from one annotator to another. The pro-ED communities usually post content that glorifies disordered eating habits, encourages fasting or other weight loss control methods, and idolizes thinness. The reason why users participate in these communities may vary; some are looking for inspiration, motivation, and support, while others are looking for like-minded users to interact with. This leads to the fact that two posts that both show a positive pro-ED attitude may include quite different content.

The definition of what is to be considered as a positive pro-ED post is based on four criteria. The criteria are almost identical to the criteria established by Giæver (2018) (see Section 4.4), with minor differences due to the fact that annotations would be performed on post-level instead of user-level. One difference is that Giæver considered a user as pro-ED if one of the criteria was fulfilled in *either* the tweets or the profile information of the user, such as username, display name, or biography. On the other hand, in this Master's Thesis, only the text from each post is considered during annotation.

The four annotation criteria for categorizing a post as pro-ED are defined as:

1. **Include a self-identification as pro-ED**
   The first criterion is satisfied if the post either states that the user is pro-ED, or states that the user has an eating disorder while being negative to recovery. One important thing to consider for this criterion is that some users do not self-identify as pro-ED themselves while still sharing pro-ED content. Because of this, the

criterion also includes if a post uses a pro-ED hashtags in a supportive manner.

2. **Express a desire for emaciation**
   The second criterion is satisfied if a post states an aspiration to become extremely thin. This includes if a post is sharing thinspiration or bonespiration content, refers to having protruding bones, describes symptoms as a consequence of emaciation, or romanticizes hospitalization.

3. **Ascribe a pro-ED event**
   An online pro-ED event is usually a collective happening where users take part in a diet or fast together, posting calorie intake in a competitive manner. The goal of these events is to minimize the calorie intake or to keep the calorie intake below a given threshold. The third criterion is satisfied if a post stated taking part in such pro-ED events, either by posting daily calorie intake or by using an event-hashtag in a supportive manner. One example of such event-hashtags is *Skinny4Xmas*, where participants are encouraged to have a daily intake of below 400 calories in the weeks before Christmas.

4. **Encouraged extreme weight control methods**
   The last criterion is satisfied if a post encourages weight control methods that were considered extreme. This included sharing extreme diets or fasts, encouraging abnormal food restrictions, or behaviors such as vomiting, misuse of laxatives, or over-training.

Alongside defining what characterized a pro-ED post, it was important to define which posts belonged to the pro-recovery class. As discussed in Section 4.1, distinguishing pro-recovery content from pro-ED content might not be trivial, as pro-recovery posts often include pro-ED words (Yom-Tov et al., 2012). For a post to be considered pro-recovery, the post had to concern eating disorders, or topics related to eating disorders, while not fulfilling any of the pro-ED criteria mentioned above. These were usually posts that either talked about eating disorder recovery in a positive manner or included pro-recovery hashtags in a supportive way. This means that any post concerning eating disorders in any way would end up being either pro-ED or pro-recovery.

Pro-recovery posts were often written by organizations or health personnel, with the goal of spreading awareness about the dangers of eating disorders. Some pro-recovery posts were also posted by users who underwent recovery and shared the experience to motivate others and themselves. One important thing to notice is that it was not enough to mention recovery alone in a post to be annotated as pro-recovery, the recovery had to be linked to eating disorders. Recovery from other things like cancer, surgery, or other mental illnesses is common. Annotating these other recovery-themed posts as pro-recovery and not as unrelated would potentially introduce errors in the dataset.

For a post to be annotated as unrelated, it would simply fail to fulfill any of the pro-ED criteria or not contain eating disorder content related to recovery. This was the case for most of the social media posts in the dataset.

5.3 Data Collection

For the purpose of this study, data was collected from two different social media platforms, namely Twitter and Reddit. The following sections will explain the collection procedures that was established for both platforms. There are differences in the collection procedures due to the different usage and nature of the social media platforms.

5.3.1 Twitter Data Collection

As eating disorder content only contributes to a small fraction of all the posts on Twitter in total, there had to be established a way to target the pro-ED and pro-recovery communities specifically. To do this, some well-known keywords and tags were listed (see Table 5.1). These tags were formed based on previous work in collecting relevant pro-ED and pro-recovery data from both Giæver (2018) and Wang et al. (2018). In addition, lexical variations of well-known pro-ED tags were also added, according to the work of Chancellor et al. (2016b). Some of the tags were also added after empirical studies of pro-ED posts on Twitter.

There was also established a set of keywords and tags for collecting unrelated posts. These were based on top trending tags on Twitter during February 2021. However, these tags were never used to collect tweets, as the first Twitter data extraction provided enough unrelated examples for the purpose of this study.

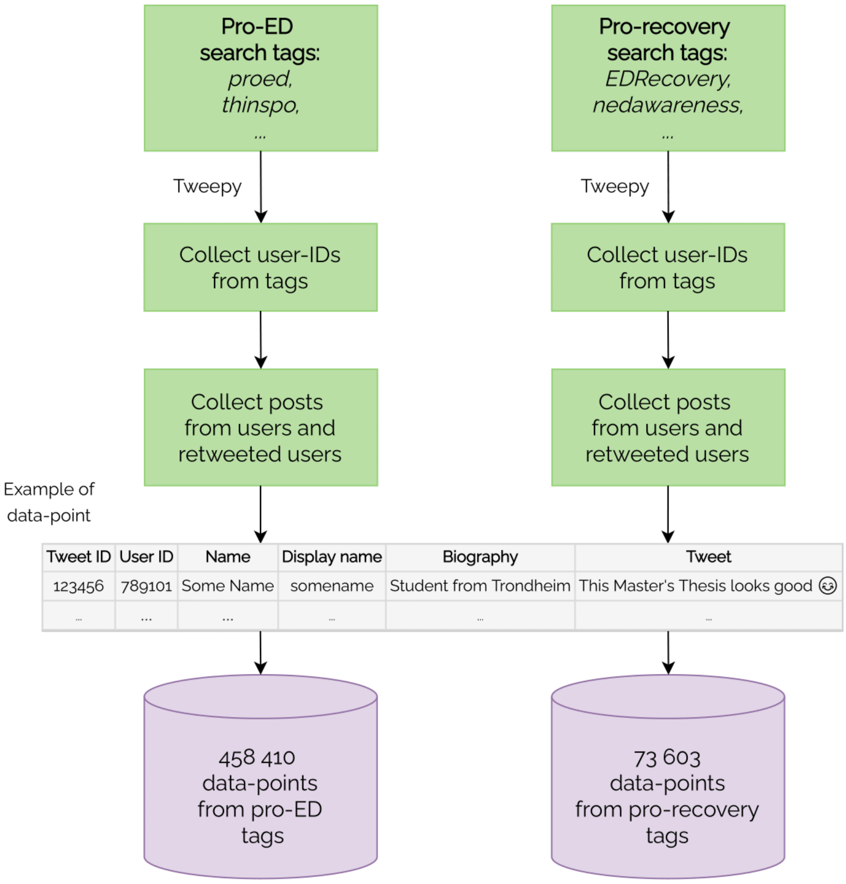
Domain	Keywords and tags
Pro-ED	*UGW, thin, legspo, eatingdisorder, proed, edproblems, hipbones, proana, anorexia, thinspo, edfamily, anatwt, meanspo, thinkthin, skinny, edprobs, edtwt, edlogic, miatwt, ricecaketwt, thinspo, thighgap, bonespo, thinspiration, promia, bulimia, skinny4xmas, mia, ana, thinsp0, bonespiration, proED, ednos, ed*
Pro-recovery	*EDRecovery, RecoveryWarriors, BeatED, anorexiarecovery, EatingDisorderRecovery, anarecovery, nedawareness, eatingdisordersupport, Eating Disorder Recovery,*

Table 5.1: Keywords and tags used for data collection on Twitter.

To collect data from the Twitter API, a Python library called Tweepy was used.[1] A complete overview of the data collection procedure can be found in Figure 5.1. First, an initial search on the established tags was provided, and Twitter user IDs were collected among the search results. The search was conducted on the 4th of March 2021, with a time span of six months, meaning that the program found users who had used these tags from that date and six months back in time (i.e., until the 4th of September, 2020). This

[1] `https://www.tweepy.org/`

initial search resulted in 6 899 user IDs fetched from the pro-ED tags, while 140 user IDs were fetched from the pro-recovery tags.


Pro-ED
search tags:
proed,
thinspo,
...
Pro-recovery
search tags:
EDRecovery,
nedawareness,
...
Tweepy
Tweepy
Collect user-IDs from tags
Collect user-IDs from tags
Collect posts from users and retweeted users
Collect posts from users and retweeted users
Example of data-point
458 410 data-points from pro-ED tags
73 603 data-points from pro-recovery tags

Tweet ID	User ID	Name	Display name	Biography	Tweet
123456	789101	Some Name	somename	Student from Trondheim	This Master's Thesis looks good 😉
...	...	...	...	...	...

Figure 5.1: Data collection procedure using the Twitter API.

Having established a base of possible relevant users, the Twitter user IDs were used in a new search to collect posts. For each user, 20 of the most recent posts were collected. If a post among these 20 posts was a retweet, the username from which the tweet was retweeted was collected, but the retweeted post itself was skipped. Then, 20 posts from the retweeted user were collected. This procedure aimed to find potentially relevant users that were not collected in the first search. However, no additional retweet usernames were collected from the posts of these retweeted users. Since there was a considerable difference in the number of pro-ED users and pro-recovery users from the initial user ID search (6 899 pro-ED users versus 140 pro-recovery users), the 50 most recent posts were collected from both the pro-recovery users and the retweeted users instead of 20. For each post, the tweet ID, user ID, username, display name, biography, and the tweet text itself were collected. All the posts were collected in the same time span as the user IDs in the initial tag search.

The result from this data collection was 458 410 posts fetched from the pro-ED user IDs and 73 603 posts fetched from the pro-recovery user IDs. This is the initial dataset prior to the pre-processing and annotation steps.

5.3.2 Reddit Data Collection

Only a small dataset was collected from Reddit, as this was to be used as a test set. The data collection procedure was almost identical to the procedure mentioned in Section 5.3.1, with small adjustments to suit the decreased amount of data to be gathered.

A brief exploration of Reddit was conducted before gathering data by manually searching for typical pro-ED keywords from Twitter. The exploration showed that the keywords, displayed in Table 5.1, also were used on Reddit. It was noticed that the resulting posts tended to include more pro-recovery content than a similar exploration on Twitter. A subset of the pro-ED tags was therefore chosen to gather data from Reddit, as it would give both pro-ED and pro-recovery posts. The tags are shown in Table 5.2.

Domain	**Keywords and tags**
Pro-ED	*pro-ED, proana, promia, thinspo, bonespo , hipbones, ugw*

Table 5.2: Keywords used for data collection on Reddit.

The exploration of pro-ED content on Reddit also showed that Reddit had banned certain subreddits where people with an eating disorder were active because the subreddits were *encouraging physical harm.* When trying to access the subreddit, the following message is displayed:

> ***r/****** has been banned from Reddit***
> *This subreddit was banned due to a violation of our Content Policy, specifically, the posting of content that encourages physical harm. If you or someone you know is struggling with an eating disorder, there are resources that can help. Visit the National Eating Disorders Association website or contact their telephone helpline at 1-800-931-2237 for more information.*

New subreddits used by the pro-ED community on Reddit have emerged after the ban. Often the subreddits come with rules for how to behave or what type of content is disallowed to post in the subreddit.

To collect data from the Reddit API, a Python library called PRAW was used.[2] A search on the established tags was conducted, and the top 100 posts from each tag were collected. The post-ID, subreddit name, the title of the post, and the content itself were collected for each post. Only the *submission* was collected, not the comments on the submissions. This resulted in 587 posts. In case the dataset would be highly unbalanced, 2 000 unrelated

[2] Python Reddit API Wrapper: `https://praw.readthedocs.io/en/latest/`

posts were gathered from Reddit's Hot-list. Posts on the Hot-list are the posts that have gotten the most upvotes and comments recently on Reddit in general.

5.4 Data Pre-Processing and Filtering

Before the datasets could be annotated and later used for machine learning purposes, a pre-processing and filtering procedure was conducted. The following section will introduce each of the pre-processing steps applied to remove noise and resolve inconsistencies in the textual data and the filtering process where non-English and uninterpretable posts were discarded. These steps will be presented separately for the Twitter and Reddit platforms, as there were minor differences in the procedures.

5.4.1 Twitter Pre-Processing and Filtering

The pre-processing procedure was done in two stages. The first initial processing was done during collection, while the second step was a filtering process where some posts were dropped if they did not fulfill certain requirements. An illustration of the pre-processing procedure can be found in Figure 5.2.

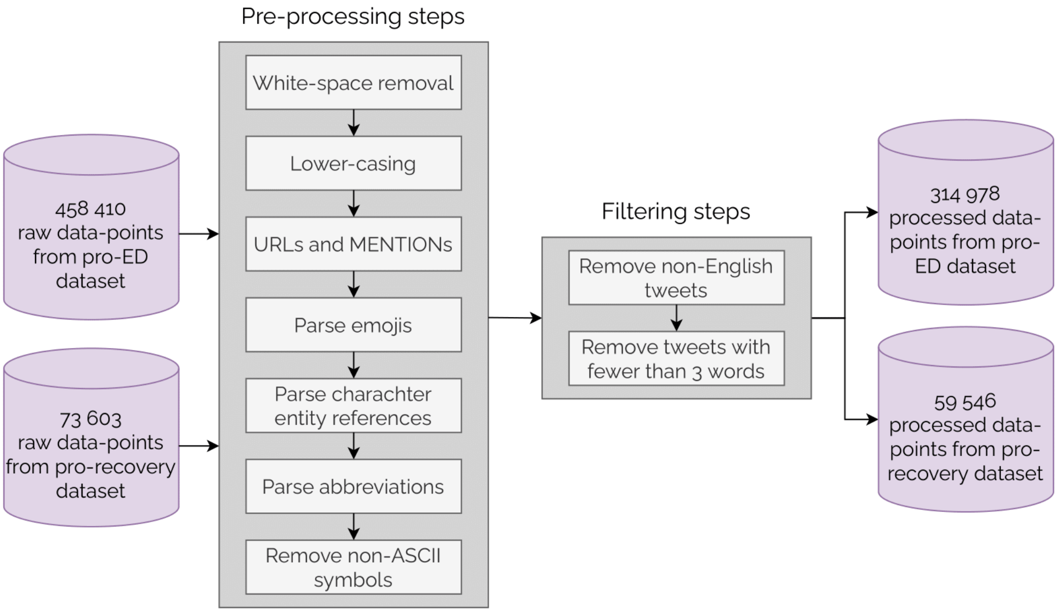

Pre-processing steps
White-space removal
Lower-casing
URLs and MENTIONs
Parse emojis
Parse charachter entity references
Parse abbreviations
Remove non-ASCII symbols
458 410 raw data-points from pro-ED dataset
73 603 raw data-points from pro-recovery dataset
Filtering steps
Remove non-English tweets
Remove tweets with fewer than 3 words
314 978 processed data-points from pro-ED dataset
59 546 processed data-points from pro-recovery dataset

Figure 5.2: Data pre-processing and filtering pipeline.

Each of the steps conducted in the initial data pre-processing is explained in the following list:

1. **White-space removal**
   This step included removing white-spaces such as line shifts and tabs from both the tweet and the biography and replace them with a single space.

2. **Lower-casing the text**
   In this step, every word in the tweet is lowercased. This is done to ensure consistency in the dataset. We do not want a tokenizer to store several versions of the same word in the corpus, i.e., one version with a capital letter and one without. For example, the words *HELP*, *Help* and *help* would all be changed to *help.* This step is essential for social media data, as users often have spelling errors and do not mind correcting things like casing in a word.

3. **Handle URLs and mentions**
   This step included replacing all URL links and user mentions with the tags *URL* and *MENTION*, respectively. For this study, the content of the URLs or the person who was mentioned was not important and would not provide any additional information to the classification task, so such a mapping to a common keyword was beneficial.

4. **Handle emojis**
   In this step, each emoji in a tweet or biography was parsed to a predefined word. All emojis have a descriptive term associated with them which defines what kind of emoji it is. Some examples are *smiling-face* representing a smile emoji, or *red-heart* representing a red heart emoji. Each emoji was prefixed with the term *EMOJI*, split on the "-" symbol to separate words, capitalized, and then concatenated with its descriptive term, e.g., *EMOJIRedHeart.* To do this, a Twitter pre-processor tool called Tweet-Preprocessor was used to detect emojis,[3] and then the Emoji library for Python was used to parse an emoji into its descriptive term.[4]

5. **Handle character entity references**
   Symbols like "<", ">" and "&" were not parsed correctly when fetched from the Twitter API. Instead, they were represented as "&lt;" (less than), "&gt;" (greater than) and "&amp;" (and, &). This step parsed these representations to the correct symbol. One of the more common emoticons in social media is the "<3", which represents a heart. Therefore, a parsing from "<3" to *EmojiRedHeart* was also included in this step.

6. **Remove punctuation and numbers**
   Another common step in data pre-processing is removing punctuation and numbers. Because these posts were to be manually annotated, some punctuation was kept in the text to provide readability for the annotators. The punctuation processing removed all special symbols except: ".,?!-’". Numbers, in general, do not give our machine learning model any useful information and were not considered essential

[3] https://pypi.org/project/tweet-preprocessor/

[4] https://pypi.org/project/emoji/

to provide good readability for the annotators. They were therefore removed.

7. **Handle abbreviations**
   This step replaced some common abbreviations or slang in social media text with the word or words that the abbreviations were meant to represent. Examples of these are *ppl* into *people*, *idk* into *i don't know*, *rly* into *really*. A complete list of which abbreviations and slang words that were substituted can be found in Appendix A.1. This step was conducted on both the tweets and the biographies.

8. **Remove non-ASCII symbols**
   In this final step, all non-ASCII symbols were removed from both the tweets and the biographies, as non-ASCII symbols were considered as noisy data.

The initial data pre-processing was followed by a filtering process. First, all non-English tweets were discarded from the dataset. Even though the data collection from the Twitter API was set to *English mode*, some of the tweets were in foreign languages. To do this, a language detection tool for Python called LangID was used to identify which language the tweet belonged to.[5] This tool calculated for each post the probability that the post belonged to a certain language. If the most probable language for a post was not English, it was discarded from the dataset.

The second filtering step included removing all posts with a word length of less than three. Both the emojis and the *MENTION* and *URL* placeholders were ignored in this check, meaning that the post needed at least three words excluding mentions, URLs, and emojis to avoid being discarded from the dataset. The reason for filtering out very short posts was that they would not provide enough useful information for annotators to label them properly. In such a case, all short tweets would end up as belonging to the unrelated class, which could give the impression that all short tweets are unrelated.

After this filtering process, both the pro-ED and the pro-recovery datasets were considerably smaller. A complete overview of the datasets before and after the filtering is shown in Table 5.3. Removing non-English posts contributed to a reduction of 15 % and 8 % for pro-ED and pro-recovery, respectively, while the removal of short posts contributed to a further reduction of 19 % and 12 %. The pro-ED dataset now consisted of 314 978 posts, while the pro-recovery dataset consisted of 59 546 posts.

The different pre-processing steps are up for discussion. The emoji language is ever-evolving, adding new emojis frequently to the vocabulary.[6] One problem that occurred during the parsing of emojis was the different skin tones that can be used. Some emojis can be used with up to five different skin tones, and the descriptive term for that emoji will depend on the skin tone, which is added as a suffix. For instance, a thumbs-up emoji with skin tone number 2 will be named *thumps-up-medium-light-skin-tone*. The emoji parsing tool used in this study had problems with this kind of emojis and instead parsed them as two separate emojis: *thumps-up* and *medium-light-skin-tone*. This might

[5] `https://github.com/saffsd/langid.py`

[6] See the complete list of Unicode emojis: `https://unicode.org/emoji/charts/full-emoji-list.html`

	Pro-ED dataset	**Pro-recovery dataset**
Size before filtering process	458 410	73 603
Size after removing non-English posts	391 008	67 762
Size after removing short posts	314 978	59 546

Table 5.3: Size of dataset before and after the filtering process.

introduce some confusion in the dataset. The skin tones will be seen as a separate emoji in the dataset (e.g., *EMOJIMediumLightSkinTone*), and the occurrence of these skin tone emojis might be high as they are common for a larger collection of the emojis in general. This must be considered during the data analysis. However, the fact that the skin tones are separated from the original emoji itself is not a huge problem for this study, as it is the core emoji that is interesting for us in the data analysis.

Another problem with the emoji parsing was emojis that were put together by other emojis. One example of this is the *family* emojis, which can consist of any combination of the *man*, *woman*, *boy* and *girl* emojis. One example is the emoji *family-man-woman-girl-girl*, which our emoji parser would split into five separate emojis. This separation introduced tweets in our dataset that contained more symbols than allowed in a regular tweet, as one emoji could be parsed into five separate words.

The removal of some abbreviations is also up for discussion. Many abbreviations used in social media could have been included, but only a handful was chosen in this pre-processing procedure. One of the benefits of this step is that possible out-of-vocabulary words that pre-trained models may not have seen before are removed. On the other hand, if the abbreviations had been kept as they were, the model might learn to recognize and understand when and how the abbreviations were used and thus resulting in a more robust model for social media data.

It is also worth mentioning that the usage and understanding of abbreviations are highly subjective. Some people might use the abbreviation *idk* and read it as *idk* instead of *i don't know*, and a casting from *idk* to *i don't know* might therefore be viewed as incorrect for that person. The same example can be used for the well-known abbreviation *lol*, meaning *laughing out loud*. This abbreviation was not included in the pre-processing step because most people read it as *lol* and not as *laughing out loud*. Casting this abbreviation might introduce tweets that are strange to read. Defining the line between what should be cast and what should not might prove hard, and the easier choice would be to ignore this step.

5.4.2 Reddit Pre-Processing and Filtering

The pre-processing and filtering procedure explained in Section 5.4.1 was also applied to the Reddit dataset, with a few exceptions. First of all, the pre-processing steps were not applied to the posts during the collection itself, as was done for the Twitter dataset, but conducted on the non-processed collection of data afterward. Secondly, Reddit posts consist of both a title and the content of the post itself. The title was considered important for the classification task and was therefore concatenated with the post's content into one single text string. After this, all pre-processing steps were carried out before filtering the posts. This process resulted in a dataset consisting of 587 posts.

The same procedure was later applied to the 2 000 posts gathered from Reddit's Hot-list, but this resulted in reducing the number of posts to 107. The reason for the considerable decrease in number of posts was revealed after an exploratory analysis of the posts. This showed that the majority of the posts contained either only a title, or title and a picture, or a link. These posts did not survive the filtering step, which required the post content to be longer than three words.

5.5 Annotations

In order to ensure a high-quality pro-ED dataset for supervised learning, a part of the collected data was sampled and manually annotated. The annotation were conducted in three stages for the Twitter dataset and once for the Reddit test dataset. Additionally, a larger pro-ED Twitter dataset was created using semi-automatic annotation techniques.

5.5.1 Annotation Procedure

All annotations were conducted by the authors of this Master's Thesis, who are two male students in their twenties. The annotators are Norwegian and have English as their second language, and neither had domain knowledge about eating disorders nor pro-ED communities on social media before the preliminary study to this Thesis.

Twitter

The whole annotation pipeline for the Twitter dataset is illustrated in Figure 5.3. For the first step in the annotation procedure, it was decided to annotate 10 000 posts manually. As explained in Section 5.4.1, two datasets consisting of pro-ED and pro-recovery data were collected and pre-processed. To annotate 10 000 posts, the first 5 000 posts from each of the datasets were selected and merged into one smaller dataset. This dataset was then shuffled and split into two, as there were only two annotators. Each annotator received a dataset consisting of 5 500 posts, where 1 000 were overlapping to check inter-annotator agreement.

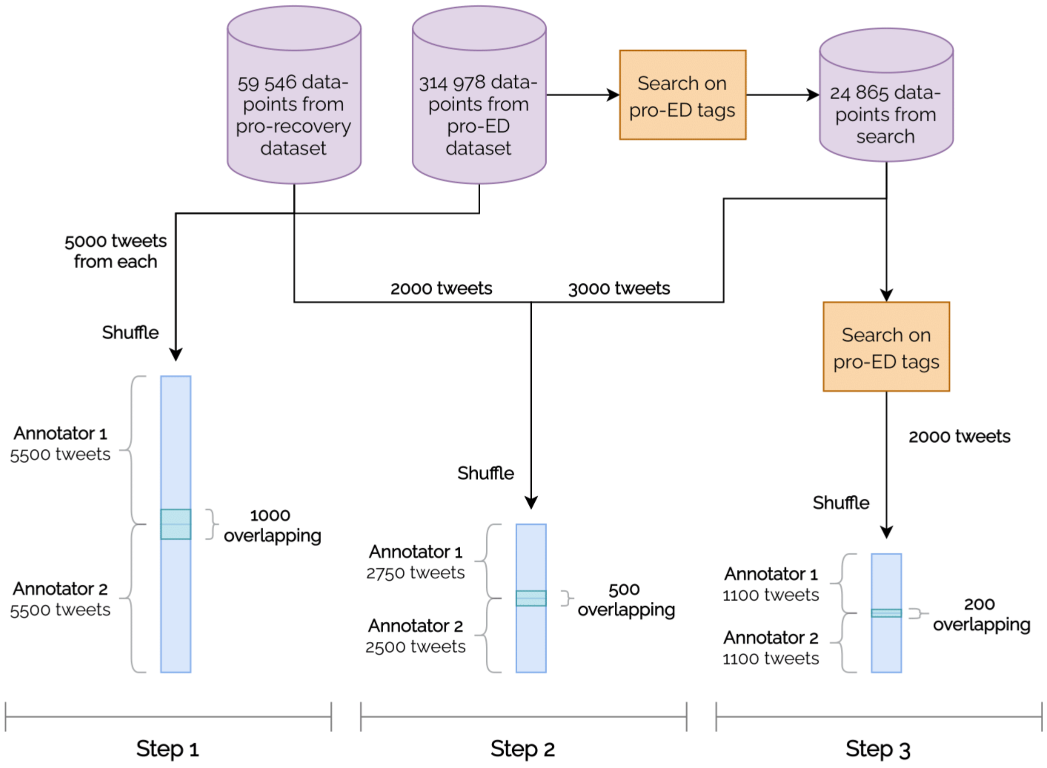


Figure 5.3: Annotation procedure pipeline for the Twitter dataset.

To conduct the annotations, a program called Prodigy was used.[7] Prodigy provides software to both manually and automatically annotate text for classification or other tasks like Named Entity Recognition, as well as model training. The program received a dataset as a .csv file, with the requirement that one of the columns was named *text*. The content of this text field was shown to the annotator, and the annotator had to select the label *unrelated*, *pro-ED*, or *pro-recovery* based on the criteria from Section 5.2. It was also possible to ignore a post during annotation. It was decided that if a post included another language than English or if the content included unreadable symbols, it would be ignored from the annotated dataset. The use of Prodigy led to a swift annotation process. In general, deciding the label for one post would take five to ten seconds on average. A decision for the unrelated posts was usually easier to establish within two to five seconds, as it became clear early on if the post had nothing to do with eating disorders. However, some pro-ED and pro-recovery posts were especially challenging to label and could take up to a minute to label. Both annotators shared this experience.

After annotating 10 000 posts, it was found that the annotated dataset was highly unbalanced, with way more unrelated posts than pro-ED or pro-recovery post. In order

[7] `https://prodi.gy/`

to cope with this problem, it was decided to annotate another 5 000 posts. This time, the pro-ED posts would be targeted more specifically before they were added to the annotation dataset. Instead of choosing random posts from the original pro-ED dataset (with 314 978 posts) to annotate, a search was conducted. The search included finding tweets that used certain tags, collecting the user ID for those tweets, and then creating a small pro-ED sub-dataset based on the tweets from these users. The tags that were used in this search were *proana*, *thinspo*, *meanspo*, *bonespo*, *edtwt*. These tags were chosen because they are strongly connected to a pro-ED attitude. The result of this search was a new dataset consisting of 24 865 tweets that were highly likely to be labeled as pro-ED.

In this second step in the annotation procedure, 3 000 posts from the new pro-ED sub-dataset and 2 000 posts from the original pro-recovery dataset were sampled, merged, and shuffled, totaling 5 000 posts. Each annotator received a sample of 2 750 posts each from this collection, where 500 of the posts were overlapping to check for inter-annotator agreement.

After this second step of annotations, the dataset was still somewhat unbalanced. The dataset consisted of 83 % unrelated posts, 9 % pro-ED posts, and only 7 % pro-recovery posts. To further increase the share of pro-ED posts in the dataset, another search targeting pro-ED posts including even more keywords were included. The keywords used in the search were *proana*, *thinspo*, *meanspo*, *bonespo*, *edtwt*, *proed*, *promia*, *binge*, and *cals*. From these keywords, the third step in the annotation procedure was conducted. Here, 2 000 additional posts were sampled and manually annotated, with 200 overlapping posts used to calculate the inter-annotator agreement score.

In total, 17 000 Twitter posts were reviewed by the annotators. During the annotations, 145 of these were ignored and removed from the dataset. Additionally, a post-processing step was added to remove potential duplicate tweets. 466 tweets were duplicates. With both ignored and duplicated posts removed from the dataset, the final annotated dataset consisted of 16 389 posts. From this, 10 % of the posts would be withdrawn and become the test dataset used in the experiments, leaving 14 750 posts in the training set.

As a result of the unbalance that was found in the annotated Twitter dataset, a balanced version of the dataset was constructed using an undersampling technique (as was explained in Section 2.1.5 Page 15). The undersampling procedure was quite simple and involved removing a certain number of random posts from the majority class (unrelated). The share of pro-ED posts in the balanced dataset was increased from 15 % to 25 %. This would lead to removing 5 774 unrelated posts. Optimally, the distribution would have been even more balanced, but doing further undersampling would result in removing a substantial amount of data, which was not desirable. The distribution of the balanced manually annotated dataset will be further discussed in Section 9.1.1 (Page 107). This undersampling was conducted on the training set of 14 750 posts and resulted in a more balanced dataset with 8 976 posts, where 64.3 % of the posts were labeled unrelated, 25 % labeled pro-ED, and 10. % labeled pro-recovery.

Domain	Keywords and tags
Pro-ED	*proana, thinspo, bonespo, edtwt, proed, thinspiration, thinsp0, anatwt, thighgap, promia, ricecaketwt, skinny, meanspo, pro ana, pro ed, pro mia, thygap, ugw, edproblems, tw*
Pro-recovery	*edaw, nedaw, edrecovery, recoverywarriors, beated, eatingdisorderrecovery, anorexiarecovery, anarecovery, nedwareness, eatingdisordersupport, ed recovery, recover from eating, ed awareness support eating disorder recovery*

Table 5.4: Keywords and tags used for semi-automatic annotation of the Twitter dataset.

Additionally, a larger semi-automatically annotated Twitter dataset was created. There were two reasons for creating this dataset: (1) a lot of data was already collected, and until now, not used, and (2) in order to compare models fine-tuned on manually annotated data versus models fine-tuned on computer annotated data. The semi-automatic annotation process was based on the presence of certain tags and keywords. First, the remaining posts from the initial data retrieval process from both pro-ED and pro-recovery tags were merged. This resulted in a merged dataset consisting of 358 135 tweets where the already manually annotated tweets were excluded. An algorithm was developed to iterate through the posts in the merged dataset and label each post as either pro-ED, pro-recovery or unrelated. To establish the annotation keywords, a selection of popular pro-ED and pro-recovery tags was fetched from both the tags used for the initial data collection (as explained in Section 5.3.1, Table 5.1), and words that were learned during the manual annotation process to be used frequently by the pro-ED and pro-recovery communities. An overview of these keywords can be found in Table 5.4. The annotation algorithm would first check if the given tweet included any of the established pro-ED keywords, and then label the tweet accordingly. If none of the pro-ED keywords were present, the algorithm checked for the presence of pro-recovery keywords in the post and labeled it as pro-recovery upon a match. If the tweet still was not labeled at this point, it was automatically labeled as unrelated.

The chosen annotation algorithm is considered effective for annotating large amounts of data, yet it may be too simple and unfair. Firstly, it may favor the pro-ED label and thus lead to a disproportionately high number of pro-ED labeled tweets. Secondly, the occurrence of one of the pro-ED keywords does not automatically imply that the post should be labeled pro-ED, as it ignores the context for that keyword. This will further be discussed in Chapter 9 (Page 107).

After running the algorithm on the merged dataset, the distribution of the three classes was very unbalanced; 98 % of the data was labeled unrelated. To cope with this problem, the dataset described in Section 4.4 collected by Giæver (2018) was used to access more pro-ED and pro-recovery data. Thus, the algorithm iterated through the posts labeled as pro-ED and pro-recovery in the Giæver dataset. A stopping criterion was added to the algorithm to guarantee a desirable label distribution; the algorithm would stop if it had labeled 25 000 pro-ED posts and 12 000 pro-recovery posts. The threshold for the number of unrelated posts was set to 100 000. The purpose of this stopping criterion was to create a semi-automatically annotated dataset with a similar distribution as the manually annotated dataset. A post-processing step was also added to remove potential duplicate tweets. The result of this process was a semi-automatically annotated Twitter dataset consisting of 136 846 tweets.

Reddit

As the Reddit dataset is only meant to be a small test dataset, it was decided to only annotate the 284 posts that were left after the pre-processing and filtering of the data collection from Reddit, as explained in Section 5.3.2 and 5.4.2. The dataset was then shuffled and split in two. The annotators received a dataset consisting of 157 posts each, where 30 of these were overlapping in order to check the inter-annotator agreement.

When annotating the posts gathered from Reddit's Hot-list, all were labeled as *unrelated* without any further exploration. The pro-ED communities are niche communities on social media platforms, and the authors found it highly unlikely that pro-ED or pro-recovery posts would appear on Reddit Hot. An empirical study was later conducted to ensure that the posts did not belong to any subreddit known to discuss eating disorders.

5.5.2 Dataset Overview and Challenges

This sub-section includes an overview of the three datasets that were created in this Master's Thesis. The main focus will be on the manually annotated Twitter dataset. Additionally, a discussion about the challenges associated with the annotation process is provided.

Twitter

After the manual annotation process had been conducted, the posts and labels were fetched from the Prodigy database and merged with the full dataset (including tweet ID, biography, screen name, etc.). The final annotated dataset consisted of 16 389 posts. For the rest of this Thesis, this dataset will be referred to as **Dataset T**. The label distribution can be found in Figure 5.4, and shows that 78.3 % of the posts were categorized as unrelated, 15.2 % as pro-ED, and 6.5 % as pro-recovery. As mentioned in Section 5.5.1, the dataset ended up being highly unbalanced, despite the measures that were initiated after both step one and two of the annotation process to cope with this problem. Another thing to keep in mind is that this amount of unrelated data came from the datasets collected with pro-ED and pro-recovery tags, and not from a dataset collected based on unrelated tags alone.

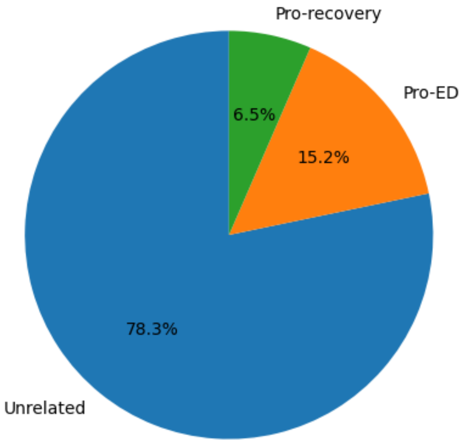


Figure 5.4: Label distribution of **Dataset T**.

The first plausible explanation for why the dataset is unbalanced is simply that pro-ED and pro-recovery data contribute only to a small amount of all the posts on Twitter and other social media platforms in general, and that the annotated data reflects this in its label distribution.

The second explanation is related to how the collection procedure was conducted. As explained in Section 5.3, only 20 posts per user were collected when extracting posts from the pro-ED user IDs. One can define these users as *primary* pro-ED users, as they were highly likely to post pro-ED content. When a retweet occurred, the retweeted user was collected, and 20 posts were collected from them as well. These users can be defined as *secondary* pro-ED users. One problem with this method was that the secondary users might not post tweets with relevant pro-ED data. In a case where *all* of the 20 posts from a primary user included a retweet, this would result in $20 * 20 = 400$ posts from secondary users. The ratio between posts that were collected from primary users and secondary users should have been in favor of primary users, and not a ratio of $1 : 1$. A better collection strategy could be to collect, e.g., 200 posts from primary users and 20 posts from secondary users, and then restrict how many secondary users to collect posts from. This will be further elaborated on in Section 9.1.1 (Page 107).

Another possible reason why the dataset became unbalanced is that annotating at post-level is different than on user-level. Giæver (2018) achieved a dataset where 63 % of the tweets were unrelated, 24 % were pro-ED and 12 % were pro-recovery when annotating based on user-level (this annotation procedure was further explained in Section 4.4). When annotating on post-level, each post had to include an explicit reference to eating disorders in some way in order not to be labeled as unrelated. Table 5.5 shows a list of example tweets from the dataset that show the difficulties when annotating on post-level

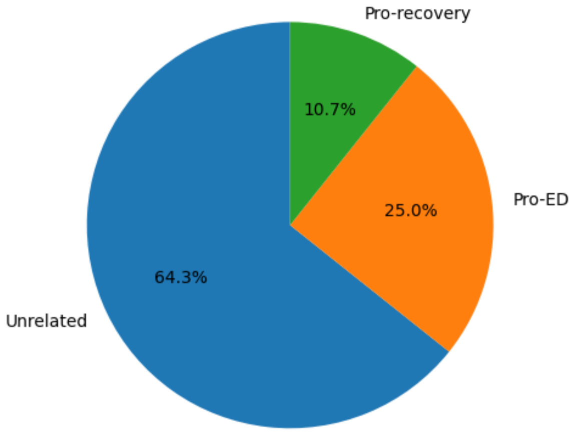


Figure 5.5: Label distribution of **Dataset T***.

ID	Tweet	Label
1	"im at a bday party and decided to wear a dress all of my friends are so skinny and i look so big"	unrelated
2	"i shouldnt have had that banana and the chocolate fuck"	unrelated
3	"just a reminder that you don't have to hop on the diet train or lose weight this year or any other year. you are enough just the way you are and you don't need to do anything at all. just be yourself and nourish that amazing body of yours! #nourishtoflourish"	unrelated
4	"help your clients, and you by building your reputation as someone who can support people struggling with their body image. URL #rdchat #bodypositive #haes URL"	unrelated

Table 5.5: Sample tweets that could be related to eating disorders but ends up being unrelated during annotation.

and why many posts end up being unrelated with the defined criteria.

The first post in the table is written by a user who might struggle with their body image. Knowing the context of this Master's Thesis, one could imagine this being a user with eating disorders. Despite this, such assumptions cannot be used to conclude what class the post should belong to during annotation. As the post itself does not fulfill any of the pro-ED criteria, nor the pro-recovery criteria, it ends up being unrelated. If the dataset was annotated at user-level, this tweet might have been included in the pro-ED class because the user could fulfill the criteria in one of their other tweets. The same applies to the second post. Knowing the context of this study, the user is probably struggling with an eating disorder, but the post itself does not explicitly fulfill the pro-ED criteria. The user who wrote this post could also possibly be a healthy person having a bad reaction to the food they just ate. The third and fourth posts are related to body image, dieting, nutrition, and weight loss but do not mention anything about eating disorders or recovery and are therefore annotated as unrelated.

Semi-Automatically Annotated Twitter Dataset

The semi-automatically annotated Twitter dataset consisted of 136 846 tweets in total. This dataset will be referred to as **Dataset S** for the rest of this Thesis. The label distribution can be found in Figure 5.6 and displays a similar distribution as **Dataset T**. As mentioned in Section 5.5.1, most of the pro-ED and pro-recovery data was fetched from the Giæver dataset. A possible challenge with this is that the pre-processing procedure conducted by Giæver has minor differences from the pre-processing procedure conducted in this Master's Thesis.

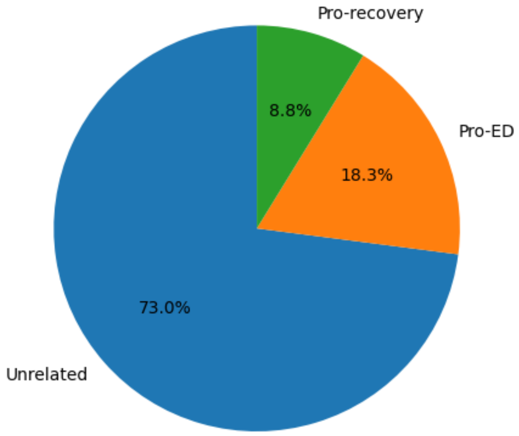


Figure 5.6: Label distribution for **Dataset S**.

Reddit

After the annotation process had been conducted, the posts were fetched from the Prodigy database alongside the labels. This data was then merged with the original dataset to obtain a fully annotated dataset. At this point, the annotated dataset consisted of 276 posts, which indicated that eight posts were ignored during annotations. This dataset will be referred to as **Dataset R** for the rest of this Thesis. The dataset consisted of 35.9 % pro-ED, 31.9 % pro-recovery and 32.2 % unrelated. As discussed, pro-ED is a niche community, and this distribution is skewed towards both pro-ED and pro-recovery, in contrast to **Dataset T** and social media in general. Therefore, 100 of the posts gathered from Reddit's Hot-list were labeled as unrelated and added to the dataset. This resulted in a Reddit dataset consisting of 376 posts in total. The label distribution for this dataset can be found in Figure 5.7.

One of the challenges that were encountered during annotation of **Dataset R** was that a post could be both pro-ED and pro-recovery at the same time. This was a challenge that was rarely encountered during annotation of **Dataset T**. The limit for posts on Reddit is 40 000 characters,[8] which is 166 times the maximum length allowed for tweets. This leads to users that express themselves in a more verbose way on the Reddit platform. A post can include a story of how a user has a desire to become extremely thin and the methods used to get there (which is considered pro-ED), and further, write about how the user battles to get the disorder under control and how others can reach out for help. In cases like this, it is hard to decide the post's label, as it fulfills the criteria for multiple classes. The annotators reached an agreement to label the posts with the label that was the most dominant for the post.

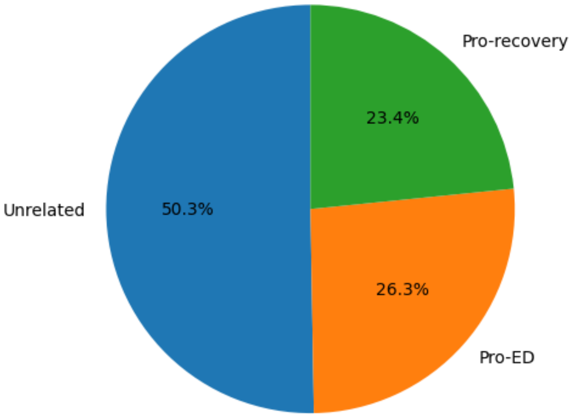


Figure 5.7: Label distribution for **Dataset R**.

5.5.3 Inter-Annotator Agreement

Although the annotation criteria are made to provide guidance for how to annotate the data, it is not possible to provide definitions that cover **all** conditions and varieties of posts. Therefore, the two annotators were annotating 1 700 of the same tweets and 30 of the same Reddit posts to be able to find the inter-annotator agreement. The inter-annotator agreement is a measure of the reliability of the annotation process and is measured using Cohen's Kappa for this study (described in Section 3.2.3 Page 34).

As the main focus of this study is to identify pro-ED posts on social media platforms, the inter-annotator agreement was calculated both for the multiclass case with three labels but also for the binary case where the classes were pro-ED and **not** pro-ED. In the binary case, the **not** pro-ED class consisted of both unrelated and pro-recovery posts.

[8] https://www.reddit.com/r/redesign/comments/ahxbva/why_is_the_character_limit_for_posts_-to_my/

When the annotators disagreed on the label of posts, the final label could not be decided by using majority voting as there were only two annotators. Therefore, the annotators manually inspected these posts once more to agree on the final class label. During the manual inspection of the posts, the annotators disagreed upon, each of the posts was tagged with a number, 1, 2, or 3, based on the reasons of disagreement that was constructed by Voutilainen (1999) (explored in Section 4.4). The reasons are revisited below:

1. Inattention on the part of one annotator (as a result of which a correct unique analysis was jointly agreed upon).
2. Joint uncertainty about the correct analysis (both annotators feel unsure about the correct analysis).
3. Conflicting opinions about the correct analysis (both annotators have a strong but different opinion about the correct analysis).

Twitter

As mentioned earlier, **Dataset T** was annotated by the two authors of this Master's Thesis only, with 1 700 overlapping Twitter posts. Additionally, there were 170 unintentional duplicates. These were tweets with a different tweet ID, but their content was the same. The inter-annotator agreement score calculated by using Cohen's Kappa can be found in Table 5.6. The score for both the multiclass and the binary inter-annotator agreement is close to 0.85.

	Multiclass	**Binary**
Annotators	0.848	0.849

Table 5.6: Cohen's Kappa agreement between the two annotators for the Twitter dataset.

Out of the 1 870 overlapping posts in the dataset, the annotators only disagreed upon the label of 122. In total, 102 of the 122 conflicting posts were tagged with reason 1, 16 were tagged with reason 2, and only four were tagged with reason 3. Inattention was thus the main reason why the labels had been different for most of the posts, and the final labels for these posts was quickly agreed upon by the annotators. The posts tagged with reason 2 and 3 were discussed more thoroughly. Table 5.7 shows examples of posts that were tagged with reason 2 and 3. With three possible classes to label the posts into, it was found that for 11 of the posts, the annotators both disagreed with themselves and chose the final third class label option instead, discarding the other two classes that the post was initially labeled as by the annotators. Two of these posts can be found in the table with ID 3 and 4. In the end, a common label had been agreed upon for all the posts and was added to the final dataset.

ID	Tweet text	Annotator 1	Annotator 2	Final label	Reason tag
1	i fucking hate my cousin she never eats she never fucking eats it makes me want to bash my head into a wall	Pro-ED	Pro-recovery	Pro-ED	2
2	hmmmm kind of sad i got my initial ed diagnosis at <NUMBER>* and i just turned <NUMBER>literally have spent <NUMBER> of my life obsessed w food ... thats fucked	Pro-ED	Pro-recovery	Pro-recovery	2
3	what are examples of being negatively affected by diet culture weight stigma? - apologizing for your body or what you've eaten - pressure on new moms to 'get their body back' - inaccessible seating - medical care withheld - weigh-ins in schools - compulsive foodbody thoughts	Pro-recovery	Pro-ED	Unrelated	3
4	starving ltc starving pse a hot mess. MENTION and i wrote a statement about it, read it here URL	Pro-recovery	Pro-ED	Unrelated	2
5	are you looking for person-centred counselling in #preston? URL is here to help! book now for minimal cost #counselling and #talkingtherapies with one of our URL counsellors and start your journey of recovery and healing today! EMOJISmilingFaceWithSmilingEyes URL URL	Unrelated	Pro-recovery	Unrelated	1

*the <NUMBER>placeholder was added by the authors to provide readability, as numbers are removed during the pre-processing

Table 5.7: List of example tweets that were disagreed upon by the annotators during the annotation process.

Reddit

Dataset R had 30 overlapping posts, which does not sound much, but it is the same share of the total as for **Dataset T**. Table 5.8 shows Cohen's Kappa for both multiclass and binary dataset. The score is lower than for **Dataset T**, and falls under the category *substantial* when considering the strength of agreement between the annotators (described in Section 3.2.3).

	Multiclass	Binary
Annotators	0.704	0.631

Table 5.8: Cohen's Kappa agreement between the two annotators for the Reddit dataset.

Out of the 30 overlapping posts in **Dataset R**, the annotators disagreed upon the label of six. This is a much higher fraction than for **Dataset T**, but since the size of the overlapping Reddit data is small, it may not generalize the same way as the overlapping Twitter data. Therefore, the two annotators manually inspected the six posts to decide the final class labels, as done with **Dataset T**. Five of the six posts were tagged with the first (number 1) of Voutilainen's reasons of disagreement, and the last post was tagged with number 3.

5.6 Data Characteristics

In order to answer research question number one, which concerns the usage of Twitter and Reddit among the members of pro-ED-communities, a brief analysis of data characteristics was conducted. **Dataset T**, consisting of 16 389 tweets, and **Dataset R**, consisting of 376 posts, were used for the analysis, which covers the presence of internet terms, use of emojis, length, and the sentiment of the posts. These features can provide useful information about what data to consider when building the classification model.

5.6.1 Internet Terms

Social media texts often include specific terms, or unigrams, that do not exist in traditional text. Among these unigrams are URLs and platform-specific terms like Twitter's user mentions and retweets. Emojis have also become a common way for people to express themselves on social media.

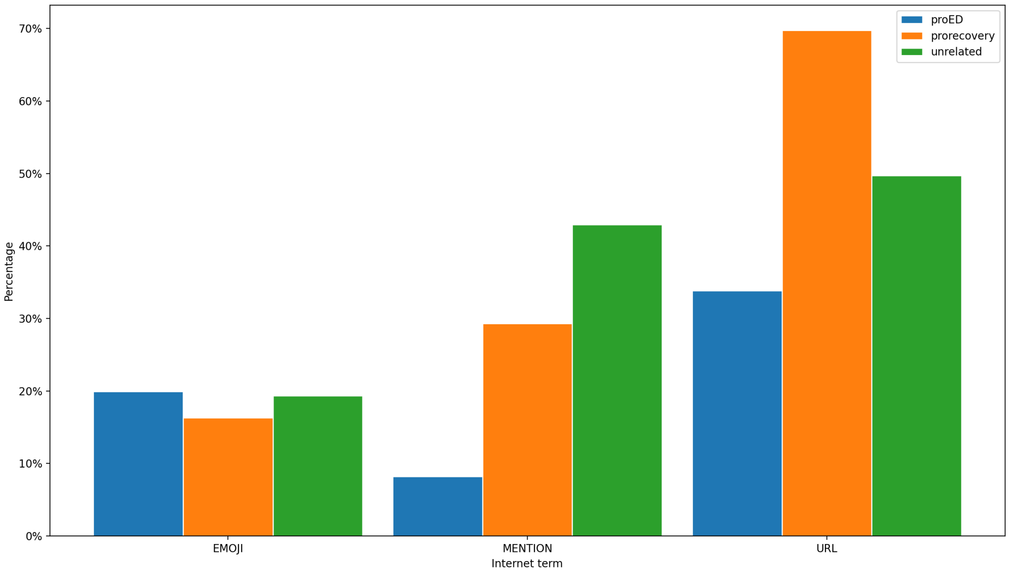


Figure 5.8: Distribution of Internet terms in **Dataset T**.

Twitter

Figure 5.8 shows the distribution of the unigrams in **Dataset T**. A noteworthy observation is that pro-ED tweets tend not to include *MENTION* as much as the other tweets. While

30.5 % of tweets labeled as pro-recovery, and 32.8 % of tweets labeled as unrelated use *MENTION*, only 8.3% of pro-ED labeled tweets include this term. Another observation worth mentioning is the frequent use of URLs among pro-recovery users. Tweets labeled as unrelated use URL twice as often as pro-ED labeled tweets. The use of emojis does not vary much between the classes.

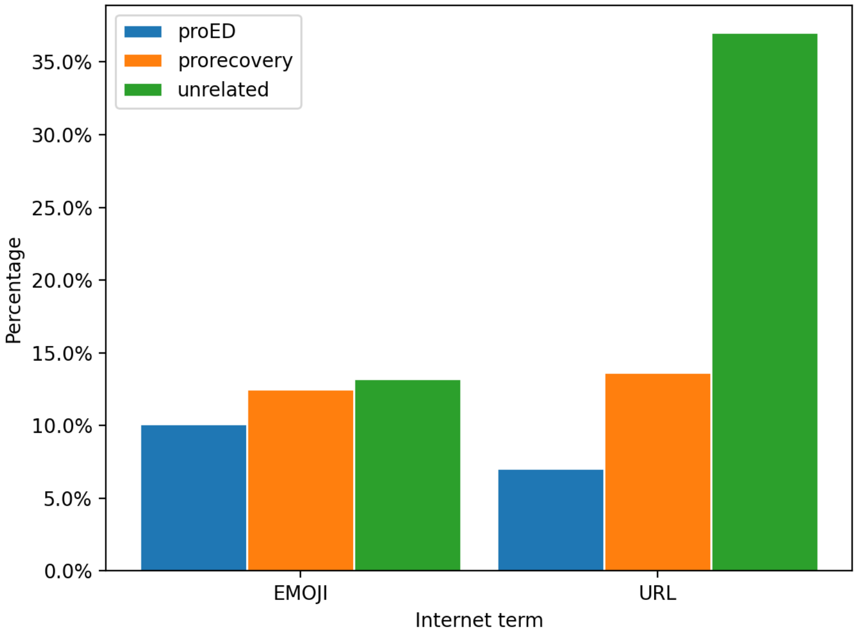


Figure 5.9: Distribution of Internet terms in **Dataset R**.

Reddit

The distribution of internet terms between the labels for **Dataset R** is displayed in Table 5.9. Emojis are used less frequently in Reddit posts than in tweets, but there are no notable differences between the labels, nor between Reddit and Twitter. For URLs, on the other hand, there are substantial differences. URLs appear in 7.1 % of posts labeled as pro-ED compared to almost 35 % of tweets with the same label. 13.6 % of Reddit posts labeled as pro-recovery include URLs, which is over five times less than the equivalent tweets. From the annotation, it was observed that pro-recovery Reddit posts often were stories from people who had suffered an eating disorder, while the tweets often referred to podcasts-episodes and whom to contact if somebody needed help. This could be an explanation for the major difference in the use of URLs among pro-recovery posts. Unrelated posts on Reddit also tend to use URLs a lot, but once again, not as much as on Twitter. The use of URLs among unrelated posts is lower than the equivalent label on Twitter. An exploration of Reddit's Hot-list shows that a common structure of a Reddit post is a title and an URL as the content. As the pre-processing excludes all posts with content shorter than three words, these post will not be a part of the dataset. Hence, the proportion of Reddit posts labeled as unrelated may be artificially low compared to the real world.

5.6.2 Emojis

Emojis are used to supplement textual language with the emotional cues that disappear when a conversation is not face-to-face or vocal. There exists a wide range of emojis, and as seen in Table 5.8 around 20 % of all tweets in the dataset use emojis, and almost 15 % of the Reddit posts include one or more emoji, as shown in Table 5.9.

Twitter

The most popular emojis in pro-ED tweets for **Dataset T** are displayed in Figure 5.10, where the *red heart* emoji is by far the most used emoji by all categories in the dataset. The second most popular emoji for pro-ED tweets is *loudly crying face*.

Twitter's custom emojis count to 3 245 emojis, which makes it easy to understand that the choices are many, and the average occurrence of an emoji in a tweet is therefore relatively low. Figure 5.10 shows this clearly. Even though the given emojis are far more present in pro-ED tweets than for the other classes, most of them are just present in about 1 % of the tweets labeled pro-ED and will therefore not be examined further.

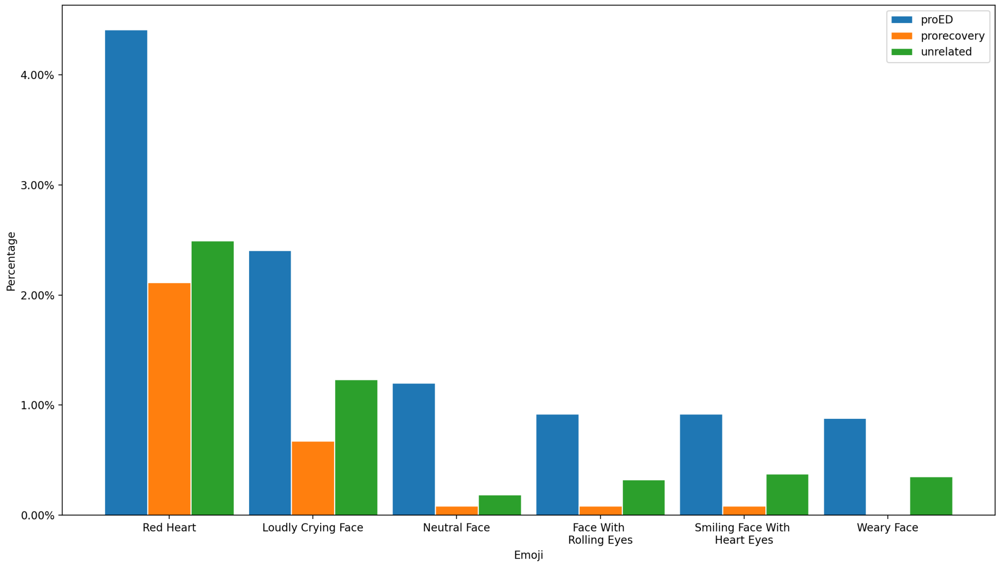


Figure 5.10: Distribution of the most popular emojis in pro-ED labeled tweets for **Dataset T**.

Reddit

As the Reddit dataset only consists of 376 posts, the presence of emojis is not sufficient and shown to be random. The distribution of emojis is therefore not visualized in this Thesis. However, it is worth mentioning that the *red heart* emoji is the most used emoji in Reddit posts as well.

5.6.3 Post length

The third feature to be analyzed was the post length, both the number of characters in a post and the number of words were examined. As seen in Section 5.4, some of the pre-processing steps included modification of the posts. One of the steps included the encoding of emojis to a string starting with *EMOJI*, i.e., Unicode character U+1F602 will be encoded as *EMOJI_loudly_crying_face*.[9] Another step including the encoding of URLs to the string *URL*. Both these pre-processing steps may have an impact on the length of the posts.

Twitter

Twitter has an upper limit of 280 characters or Unicode glyphs.[10] The encoding of emojis will, therefore, potentially increase the number of characters in a tweet substantially. In the example from the previous paragraph, the tweet would gain 23 characters which is an 8.2 % increase if the tweet has used all available characters. To avoid this, each emoji has been replaced with a single character, *e*, to imitate the original length of each emoji as only one character.

MENTION is another term generated in the pre-processing step, but it is not possible to retrieve the original length of the username that was replaced. Twitter usernames can range from four to fifteen characters (which gives a mean of 9.5 characters), and the replacement token is six characters. Therefore, the total tweet length will be slightly affected.[11] As seen in Figure 5.11, some tweets exceed the limit of 280 characters, which is the result of the aforementioned pre-processing step.

A third modification of the tweets that may have affected the length is the encoding of URLs to the string *URL*. As mentioned in Section 5.4, the content of the URL is in itself uninteresting, and as they randomly vary in length, they do not provide any useful information.

Figure 5.11 visualizes the number of characters in the tweets. A clear trend is that the pro-recovery labeled tweets are considerably longer than pro-ED and unrelated labeled tweets, which have a pretty similar distribution, with an early peak. The distribution for pro-ED peaks at a slightly higher number of characters than unrelated labeled tweets.

[9] https://www.compart.com/en/unicode/U+1F602

[10] https://developer.twitter.com/en/docs/counting-characters

[11] https://help.twitter.com/en/managing-your-account/twitter-username-rules

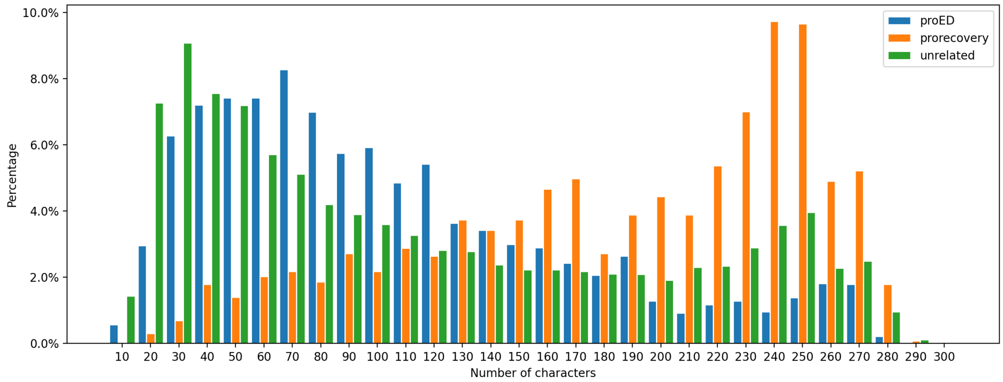


Figure 5.11: Distribution of number of characters in tweets for **Dataset T**.

Another observation is that for both pro-ED and unrelated labeled tweets, the frequency of tweet length gets a boost after 200 characters.

The same patterns are seen for word count and is displayed in Figure 5.12. The pro-ED and the unrelated labeled tweets have a similar distribution, with pro-ED peaking at a slightly higher word count before declining, while pro-recovery labeled tweets, in general, have far more words than the others. This time only unrelated labeled tweets get a boost at a high word count.

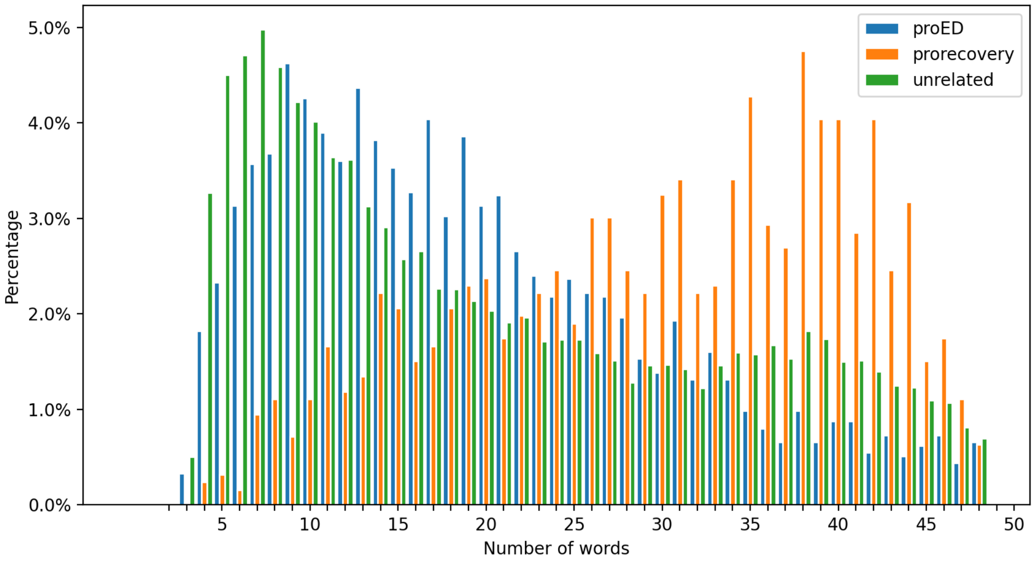


Figure 5.12: Distribution of word count in tweets for **Datset T**.

Reddit

As Reddit has a 40 000 character limit and the posts are quite long, the encoding of emojis will not have the same impact on the length of the posts. Figure 5.13 shows the length of posts in **Dataset R**, and with an average of 1 021 characters, the pre-processing steps will not change the length of a post in the same way as with tweets. For comparison reasons, the same approach as with **Dataset T** has been taken. Figure 5.14 shows the length of posts with regards to word count.

The figures show that pro-ED posts are skewed towards left when comparing them to the other labels, with a mean that is only 66 % of the mean of pro-recovery and unrelated posts. When comparing **Dataset R** to **Dataset T**, there is a substantial difference in the mean of both word length and number of characters, as seen in Table 5.9.

	Word count (mean)	**Number of characters (mean)**
Twitter	21	124
Reddit	196	1021

Table 5.9: Comparison of average post length between Twitter and Reddit posts.

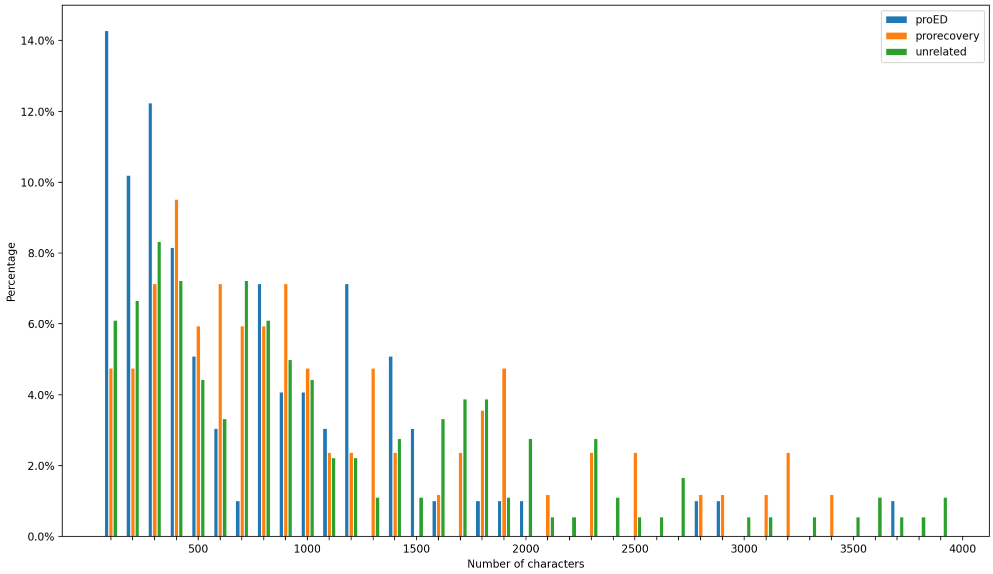


Figure 5.13: Distribution of number of characters in posts for **Dataset R**.

5.6.4 Sentiment Analysis

A sentiment analysis of the posts was also carried out. The posts were categorized into three sentiment categories: *positive*, *neutral* and *negative*. When computing the sentiment

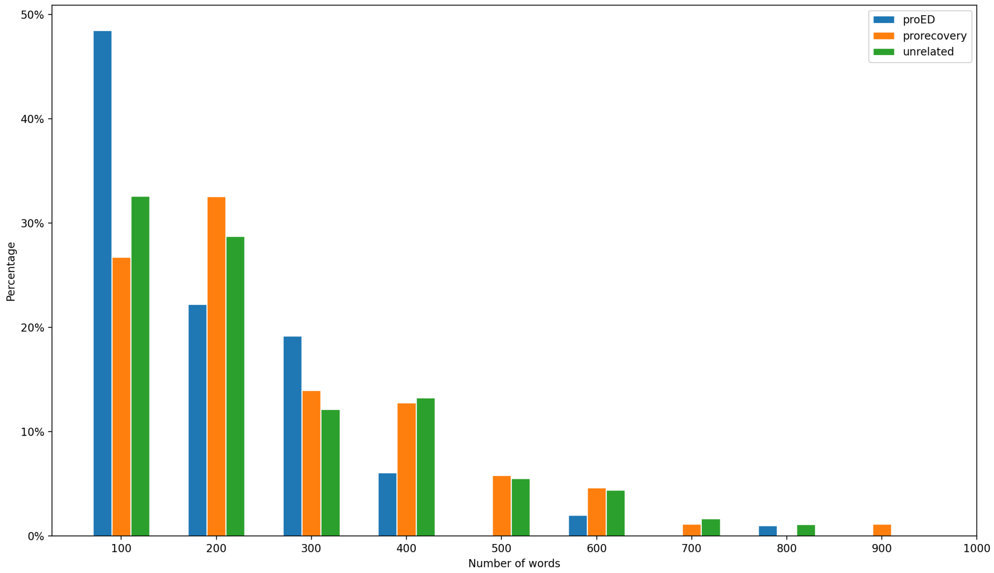


Figure 5.14: Distribution of word count in posts for **Dataset R**.

category for each post, a normalized, weighted composite score is used. The *compound* score, defined by Hutto et al. (2014), is useful when a single uni-dimensional measure of sentiment is needed. Table 5.10 shows the thresholds for the *compound* score when classifying the sentiment of the post, as given by Hutto et al. (2014).

Sentiment	**Compound score**
Positive	$\geqslant$ 0.05
Neutral	<0.05 and >-0.05
Negative	$\leqslant$ -0.05

Table 5.10: Threshold for sentiment score.

Twitter

Figure 5.15 visualizes the distribution of sentiment for each label in **Dataset T**. Tweets labeled as pro-recovery tend to have a more positive sentiment, with most of the tweets categorized as either positive or neutral. The same pattern is applicable for tweets labeled as unrelated but with a higher share of tweets with negative sentiment. For pro-ED labeled tweets, the distributions between the sentiment categories are quite equal with no tendency.

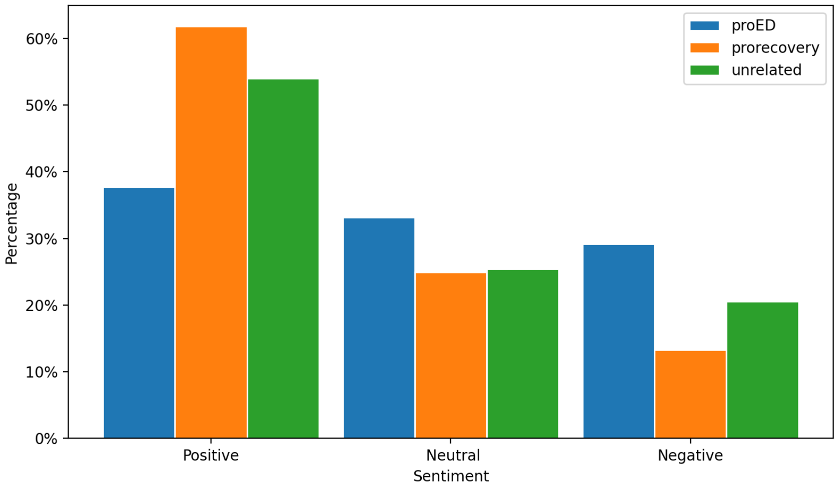


Figure 5.15: Sentiment in tweets for **Dataset T**.

Reddit

The sentiments of Reddit posts are shown in Figure 5.16. All three labels have over 95 % of their posts categorized as either positive or neutral sentiment. The share of positive sentiment versus neutral sentiment is almost equal, with positive slightly higher than neutral sentiment. When comparing **Dataset R** to **Dataset T**, the major difference is with the low amount of negative sentiment among Reddit data. The pro-ED posts also tend to be more positive on Reddit than on Twitter.

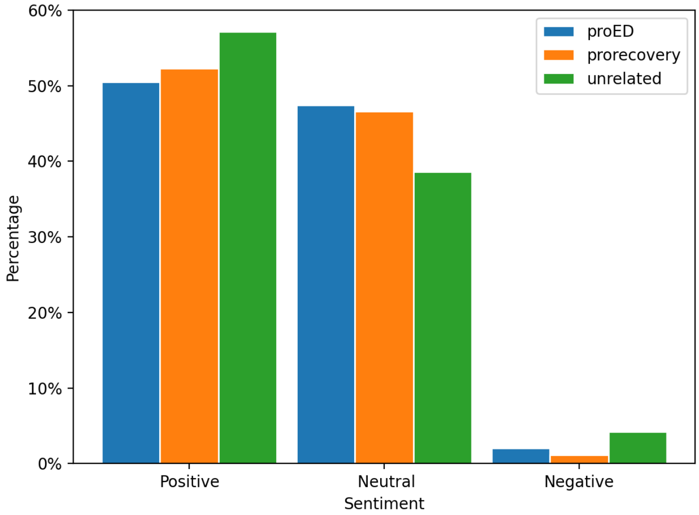


Figure 5.16: Sentiment in posts for **Dataset R**.

6 Preliminary Study

During the fall of 2020, a preliminary study prior to this Thesis was carried through, and this chapter will present the main contributions. The purpose of the study was to conduct an analysis for building a classification system for the detection of pro-ED users on Twitter, comparing the current state-of-the-art and newly developed techniques in the field of deep learning. The dataset was collected and annotated by Giæver (2018), and further pre-processed and annotated by Nornes and Gran (2019), and was presented in Section 4.4. The first section of this chapter will present the architecture of the models that were used, and the second section will present the results from the conducted experiments.

6.1 Architecture

This section will present the system architecture and the models used in the preliminary study. First, the overall system architecture is presented, and then the architectures of the attention-based models are described, along with the architectural choices.

6.1.1 System Architecture

The overall system architecture is shown in Figure 6.1 and describes the process from the input dataset to the evaluation of the models. As mentioned, the dataset used in the preliminary work was the dataset gathered by Giæver (2018), which was later reviewed by Nornes and Gran (2019). The dataset was first split into a test and training set and further into tweet documents and class labels. The tweet documents were tokenized in order to be used as input to the models. The train dataset was used to train both a Support Vector Machine (see Section 2.1.3 Page 13) and three attention-based models (further described in Section 6.1.2). After training, the models were tested on the test dataset to evaluate the different models' performance.

6.1.2 Attention-Based Architecture

The attention-based models used in the preliminary work were BERT, ALBERT and DistilBERT, described in Section 2.2.5 (Page 19), 2.2.6 (Page 21) and 2.2.7 (Page 21) respectively. The architectures used in the experiments were the same as suggested by Devlin et al. (2018) for BERT, Lan et al. (2019) for ALBERT, and Sanh et al. (2019)

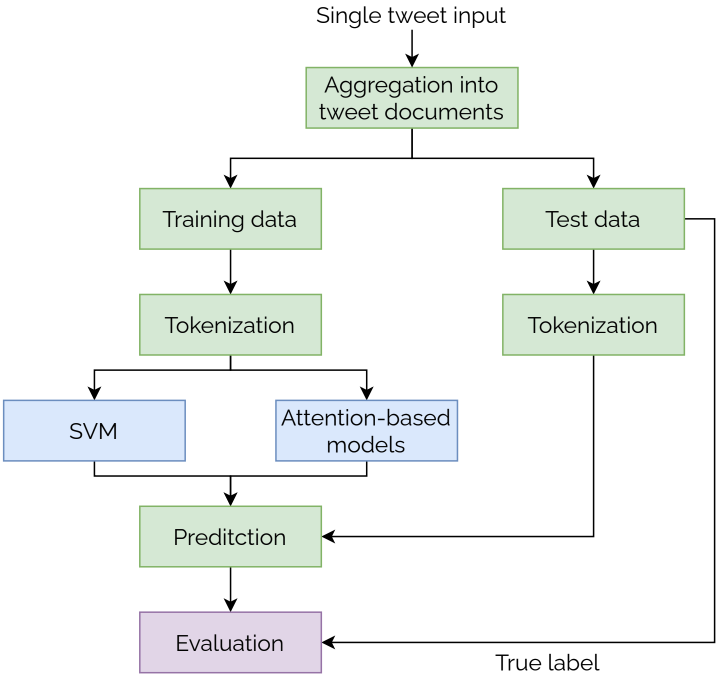


Figure 6.1: Overall system architecture.

for DistilBERT. As the problem was a textual multiclass problem, a model architecture specific for sequence classification was chosen.[1] Figure 6.2 shows the neural network layer composition of the sequence classification models. The first layer included the pre-trained models. The dropout layer was included to prevent the models from overfitting, while the dense layer was used for classification purposes. The DistilBERT model also included a dense pre-classifier layer using the ReLU activation function in the model architecture. The fine-tuning process updated the node weights in these layers based on the supervised input data.

BERT and ALBERT come in different sizes, where the base version is the standard version of the models. Devlin et al. (2018) and Lan et al. (2019) also created larger versions, with a more complex architecture to improve the performance. A comparison between $\text{BERT}_{\text{BASE}}$ and $\text{BERT}_{\text{LARGE}}$ can be found in Table 2.1 (Page 20). As the preliminary study only wanted to *explore* the performance of attention-based models, the base versions were selected for the extensive part of the experiment to keep it simple and to have the possibility to explore several batch sizes without running out of memory. Additionally, configurations beyond the default ones were tested. This included implementing $\text{BERT}_{\text{LARGE}}$, DistilBERT cased, and DistilBERT uncased with

[1] Proposed by Huggingface, `https://huggingface.co/transformers`

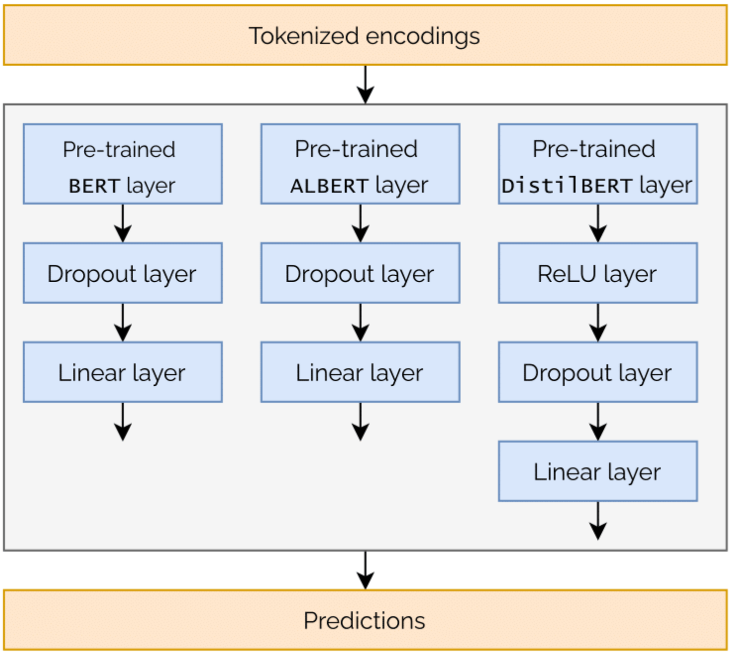


Figure 6.2: Huggingface's model architecture for BERT, ALBERT and DistilBERT.

larger sequence length. The motivation behind implementing these was to explore if they achieved any results that could be investigated in future work. An overview of the model sizes, and parameter choices suggested by Devlin et al. and Lan et al. is found in Table 6.1.

Devlin et al. (2018) state that the uncased models usually yield better performance unless it is known that case information is important. A model that is uncased has only been trained on lowercase text, and since the dataset obtained from Nornes and Gran (2019) only contains lowercase text, the uncased models were preferred.

Parameter	**BERT$_{\text{base}}$**	**ALBERT$_{\text{base}}$**	**DistilBERT**
Hidden layers	12	12	6
Hidden size	768	768	768
Attention heads	12	12	12
Max Sequence Length	512	512	512
Learning rate	5e-5, 3e-5, 2e-5	1e-5	-
Batch size	16, 32	32	-

Table 6.1: Parameters used by Devlin et al. (2018) and Lan et al. (2019).

6.2 Experimental Results

This section will introduce the results from the conducted experiments in the preliminary study. The performance of the SVM (which was a recreation of the experiment conducted by Nornes and Gran, 2019) and the performance of the attention-based models will be presented separately. The system's performance was evaluated by the metrics introduced in Section 2.3 (Page 26), namely precision, recall, F_1-score, and accuracy.

6.2.1 Experiment 1 - Results

The results of Experiment 1 are displayed in Table 6.2, which shows the performance of both the experiment from Nornes and Gran (2019) and the reproduced model. The two models differ by 0.001 for recall and F_1-score while being similar for precision.

Model	**Precision**	**Recall**	$\mathbf{F_1}$
$\text{SVM}_{\text{Nornes and Gran}}$	0.971	0.971	0.970
$\text{SVM}_{\text{reproduced}}$	0.971	0.970	0.969

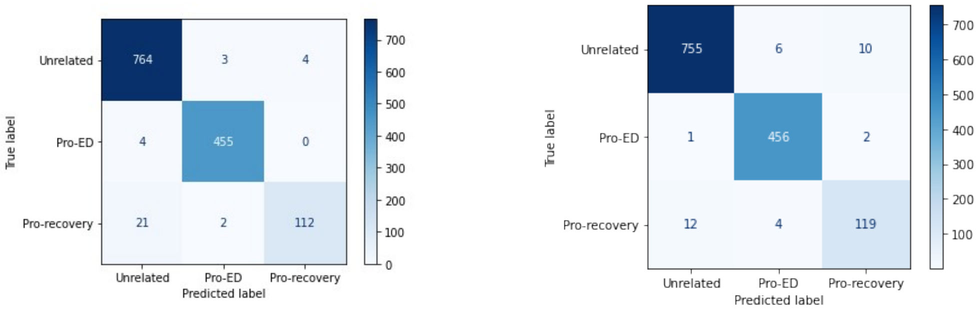
Table 6.2: Results from Experiment 1.

		Unrelated			**Pro-ED**			**Pro-recovery**			**Total**	
Models	**Batch size**	**P**	**R**	**F**	**P**	**R**	**F**	**P**	**R**	**F**	**Weight. Avg F1**	**Acc**
SVM	-	0.968	0.991	0.979	**0.991**	0.911	**0.991**	**0.957**	0.830	0.889	0.975	0.974
BERT	16	0.565	**1**	0.722	0	0	0	0	0	0	0.565	0.408
	32	0.955	0.988	0.971	0.978	0.978	0.978	0.954	0.763	0.848	0.963	0.961
	64	0.958	0.986	0.972	0.82	0.974	0.978	0.940	0.815	0.873	0.965	0.964
	128	0.966	0.968	0.967	0.950	**0.998**	0.973	0.901	0.741	0.813	0.955	0.954
$\text{BERT}_{\text{LARGE}}$	32	0.950	0.988	0.969	0.991	0.969	0.980	0.921	0.778	0.843	0.961	0.960
ALBERT	16	0.565	**1**	0.722	0	0	0	0	0	0	0.565	0.408
	32	0.565	**1**	0.722	0	0	0	0	0	0	0.565	0.408
DistilBERT	16	0.980	0.977	0.979	0.981	0.987	0.984	0.889	0.889	0.889	0.971	0.971
	32	0.983	0.979	0.981	0.979	0.993	0.986	0.908	0.881	0.895	0.974	0.974
	64	0.983	0.973	0.978	0.985	0.987	0.986	0.908	0.881	0.895	0.974	0.974
	128	**0.984**	0.960	0.972	0.974	0.991	0.983	0.836	**0.904**	0.868	0.965	0.965
DistilBERT*	32	0.973	0.994	**0.983**	**0.991**	0.978	0.985	0.936	0.867	**0.900**	**0.976**	**0.976**
DistilBERT**	32	0.967	0.982	0.974	0.982	0.976	0.979	0.865	0.807	0.835	0.963	0.962

Table 6.3: Results from Experiment 2 conducted in the preliminary study. DistilBERT* is fine-tuned using a sequence length of 1024, while DistilBERT** is the cased version of DistilBERT.

6.2.2 Experiment 2 - Results

Experiment 2 investigated how different attention-based models perform compared to the SVM from Experiment 1, and the results are displayed in Table 6.3. These findings showed that the SVM model achieved the best performance overall with regards to both F_1-score and accuracy. DistilBERT with sequence length 1 024 achieved the best results, outperforming both the SVM and the DistilBERT model with sequence length 512. However, the SVM was better than the attention-based models at classifying *pro-ED* users. Figure 6.3 shows the confusion matrices for the two best classifiers, the SVM (Figure 6.3a) and DistilBERT (Figure 6.3b). Both models found it harder to classify the *pro-recovery* users than the other classes. Another result worth mentioning is that $\text{BERT}_{\text{LARGE}}$ achieved a higher F_1-score than its equivalent $\text{BERT}_{\text{BASE}}$, when classifying pro-ED users. The BERT model fine-tuned with batch size 16 and the ALBERT models fine-tuned with batch size 16 and 32 performed badly on the dataset by classifying all users as *unrelated* in this experiment. The fine-tuning of ALBERT with batch sizes higher than 32 resulted in an Out-Of-Memory error in this experiment.



(a) Confusion Matrix for SVM classifier.

(b) Confusion Matrix for fine-tuned DistilBERT.

Figure 6.3: Confusion matrices.

As the text documents processed by the tokenizer were quite large, an analysis of the number of truncated tokens was conducted, and the result is presented in Table 6.4. The average number of tokens is 53 times larger than the sequence length the models were trained on. When running the tokenizer for BERT and DistilBERT only 1.76 % of the users' tweet documents were not affected by truncation, and 2.44 % were not affected by the truncation by the ALBERT tokenizer.

Tokenizer	**Average number of tokens**	**Users with number of tokens >512**
BERT & DistilBERT	28 111	98.24%
ALBERT	26 812	97.56%

Table 6.4: Number of tokens and tokens removed by truncation.

7 Architecture

The following chapter will describe the different model architectures that are used in the experiments of this study. The first section will cover how a Support Vector Machine baseline was implemented, while the second section will include an overview of the different attention-based model architectures. The last section proposes two stacked ensemble learner architectures with different meta-classifiers on top to perform the final prediction. This chapter will only contribute an overview of the systems in this study, while the actual implementation and tuning of hyperparameters will be further presented in Chapter 8.

7.1 Baseline Support Vector Machine Architecture

To create a baseline text classification model, a Support Vector Machine (SVM) was implemented. The baseline would serve as a good reference point when evaluating the attention-based transformer models. The SVM was chosen as the baseline as it had proved to give good results when classifying eating disorder data before (Giæver, 2018; Nornes and Gran, 2019; Ramírez-Cifuentes et al., 2018). The SVM was also used in the preliminary study for this Master's Thesis and achieved better results than some attention-based models when classifying pro-ED users on Twitter (as explained in Section 6.2.2).

To create the baseline SVM classifier, features from **Dataset T** had to be extracted and prepared for learning. Choosing the correct feature groups for training a machine learning model is an important part of creating a well-functioning classifier. An illustration of how the SVM classifier was created can be found in Figure 7.1. The feature groups investigated for creating the SVM baseline are inspired by the work of Giæver (2018) and Nornes and Gran (2019). Some of the feature groups were also chosen based on the findings from the dataset analysis in Section 5.6. The feature groups that are explored for the baseline classifier are listed below:

1. **N-grams of tweet text**
   Both Giæver (2018), Nornes and Gran (2019) and the preliminary study found that calculating the TF-IDF score for n-grams of tweets was the most influential feature when classifying pro-ED data.

2. **N-grams of biographies**
   The biography of the Twitter user was collected together with the tweet during

data collection. Again, the studies of Giæver (2018) and Nornes and Gran (2019) showed that n-grams of the biography text could provide useful information during classification.

3. **Character N-grams of names**
   Both usernames and display names were collected for each tweet. Giæver (2018) found that many eating disorder-related Twitter accounts had body image references in their names and could therefore be investigated as a useful feature. Here, n-grams on character-level were used.

4. **Tweet word length**
   Section 5.6.3 shows that there are notable differences between the post length across the three classes, especially posts that are labeled as pro-recovery seem to be longer when it comes to word and character count. This feature was tested to see if the model can benefit from this information during classification.

5. **Sentiment analysis score from tweets**
   Section 5.6.4 shows that the sentiment of the tweet texts varies across the three classes; pro-recovery and unrelated posts are more positive in general compared to pro-ED posts. Therefore, a sentiment score in the interval [-1,1] was stored for each post in the dataset and used during classification.

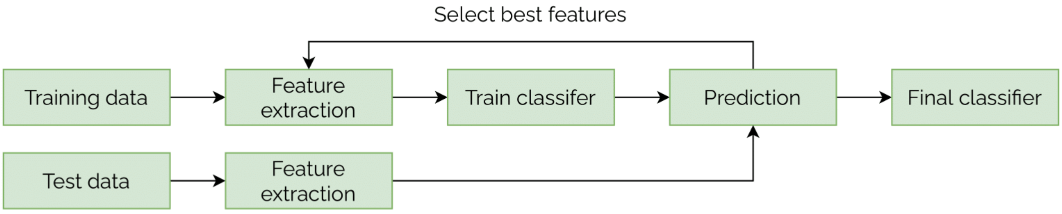


Figure 7.1: Support Vector Machine classifier pipeline.

7.2 Attention-Based Architectures

For the purpose of this study, five different pre-trained attention-based models are fine-tuned on the task of classifying pro-ED posts from social media. The five models are BERT (Devlin et al., 2018), BERTweet (Nguyen et al., 2020), DistilBERT (Sanh et al., 2019), ERNIE 2.0 (Sun et al., 2020), and RoBERTa (Liu et al., 2019). These were chosen based on the findings from related research (see Section 4.3 Page 41). ALBERT (Lan et al., 2019) was discarded due to the problems encountered in the preliminary study of this Thesis. This section provides an overview of all the model architectures as well as which hyperparameters that are used during the fine-tuning process.

7.2.1 Model Architectures and Layers

All of the attention-based model architectures are essentially based on or extensions of the original BERT architecture. In order to fine-tune the models on this specific text classification task, additional layers are added on top of the pooling output of the already pre-trained layers in the deep neural network architectures of these models. This composition of pre-trained layers combined with linear, fully-connected layers on top is proposed and made available through specific sequence classification versions of the models.[1]

An overview of the layer composition used for each model during fine-tuning is found in Figure 7.2. Both the BERT model and the ERNIE 2.0 model for sequence classification use a dropout layer in combination with a linear classification layer on top of the already pre-trained layer. The DistilBERT model uses a pre-classifier with a ReLU activation function before the dropout and linear classification layer. The RoBERTa model for sequence classification uses the pre-trained RoBERTa layer together with a custom classification head consisting of two dropout layers, one hidden layer with the hyperbolic tangent activation function, and a linear layer on top. The BERTweet model consists of the pre-trained BERTweet layer together with the same classification head as RoBERTa.

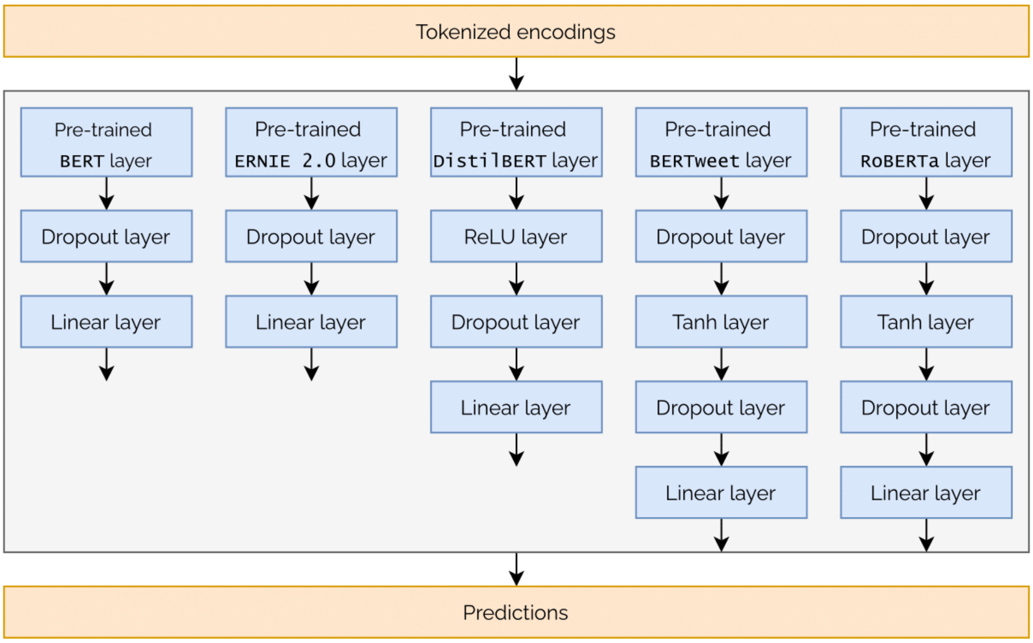


Figure 7.2: Overview of the attention-based model layer compositions used for fine-tuning.

An overview of how the fine-tuning process of the attention-based models is conducted

[1] Proposed by Huggingface, `https://huggingface.co/transformers`

can be found in Figure 7.3 (using BERT as an example), and is explained as follows: Each post in the dataset is treated as a sequence, and an encoded representation of the text is created using the tokenization algorithm used by the different models. Both BERT, DistilBERT, and ERNIE 2.0 use the WordPiece tokenization (Schuster and Nakajima, 2012), while RoBERTa and BERTweet use Byte-Pair Encodings tokenization (Sennrich et al., 2015). The first token of every sequence is the special classifier token [CLS]. Each sequence is then fed through the pre-trained language model, which produces the pooled output sent through the additional linear layers on top. The pooled output is the hidden state output of the [CLS] token. The weights of the pre-trained layer and the linear layers are then trained together using the cross-entropy loss function. The output sizes of these linear layers corresponds to the number of labels for the given task, i.e., three for the multiclass task and two for the binary task. This output is then used to make a prediction for each post in the dataset.

7.2.2 Hyperparameters for Fine-Tuning

When fine-tuning BERT for different NLP tasks, Devlin et al. (2018) proposed keeping the default hyperparameters for the BERT model. The only exception was that the batch size, number of epochs, and the learning rate should be adjusted to the purpose of the task. The experiments in this study will follow the proposal from Devlin et al. (2018), as Liu et al. (2019) and Nguyen et al. (2020) also did when they conducted their fine-tuning. The learning rate optimizer that will be used is the Adam optimizer (Kingma and Ba, 2014). A full list of the hyperparameters and other characteristics of the attention-based models can be found in Table 7.1. All models will be trained using learning rates in the range (5e-5, 2e-5, 1e-5), with batch sizes (16, 32) for four epochs.

Parameters	**Hidden Layers**	**Hidden Size**	**Attention Heads**	**Maximum seq length**	**Learning Rate**	**Batch Size**	**Epochs**	**Tokenization Algorithm**
BERT$_{\text{BASE}}$	12	786	12	512	5e-5, 2e-5, 1e-5	16, 32	4	WordPiece
BERT$_{\text{LARGE}}$	24	1024	16	512				
ERNIE$_{\text{BASE}}$	12	786	12	512				
ERNIE$_{\text{LARGE}}$	24	1024	16	512				
DistilBERT	6	768	12	512				
RoBERTa$_{\text{BASE}}$	12	786	12	512				Byte-Pair Encodings
BERTweet	12	786	12	128				

Table 7.1: Hyperparameters used for fine-tuning the attention-based models on the task of classifying pro-ED posts.

7.3 Meta-Classifier Architectures

Research Question 4 is directly linked to how attention-based models can be combined to create the best possible classifier. In order to create the best possible classifier to detect pro-ED posts in social media, two stacking ensemble architectures using a meta-classifier were developed. Both ensemble models consisted of several fine-tuned

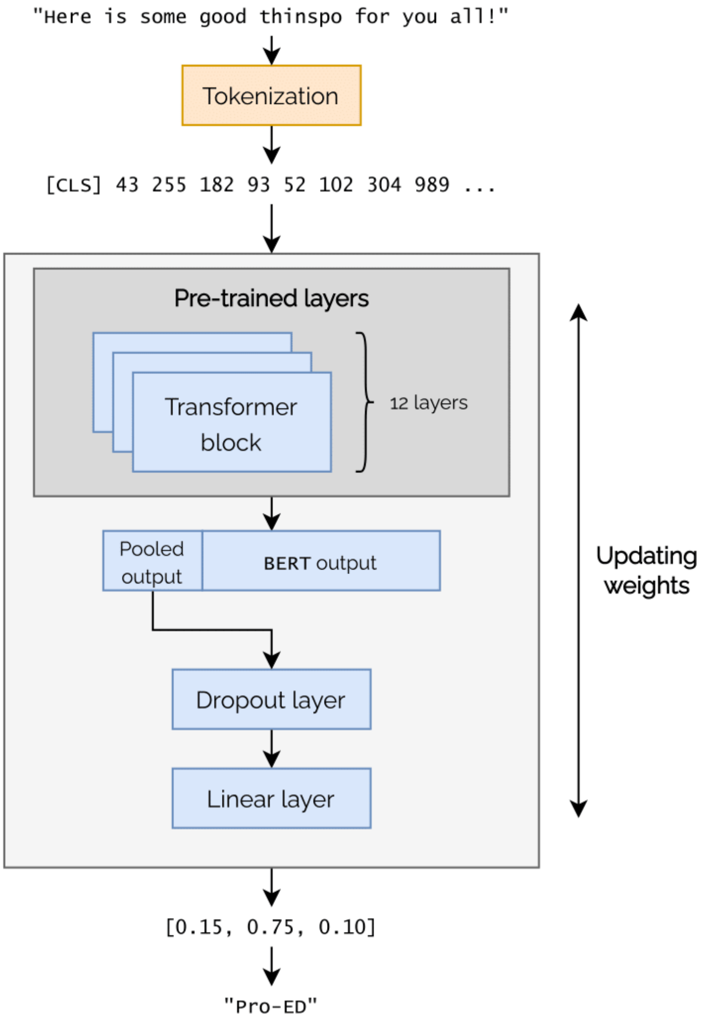


Figure 7.3: The fine-tuning pipeline from raw data input to final prediction.

attention-based models, but the meta-classifier on top was different. The first meta-classifier is a simple voting classifier, while the second is a feed-forward neural network trained on the predictions from the attention-based models.

7.3.1 Voting Classifier Architecture

The stacked voting classifier architecture included the best baseline SVM learner, along with the best resulting fine-tuned versions for each of the attention-based models presented in Section 7.2. An illustration of the architecture can be found in Figure 7.4. All level-0 models were trained on the tweet from the same training dataset, while the SVM was also trained using the biography of the Twitter user. The predictions from these level-0 models was then fed to the voting classifier. The voting classifier was implemented with both a *hard* and *soft* schema. The *hard* option would simply collect the predicted label from each model and choose the predicted label with the most votes as the final prediction. If multiple classes get the highest number of votes, the first occurring class is chosen. The *soft* option receives the probability for each class from each model and summarizes these probabilities. The class with the highest sum is chosen as the final prediction. The final prediction $\hat{y}$ can be defined as:

$$\hat{y} = argmax_j \sum_{i=1}^{n} p_{i,j},$$

where $p_{i,j}$ is the probability for class label j predicted by the i-th classifier (out of n classifiers).

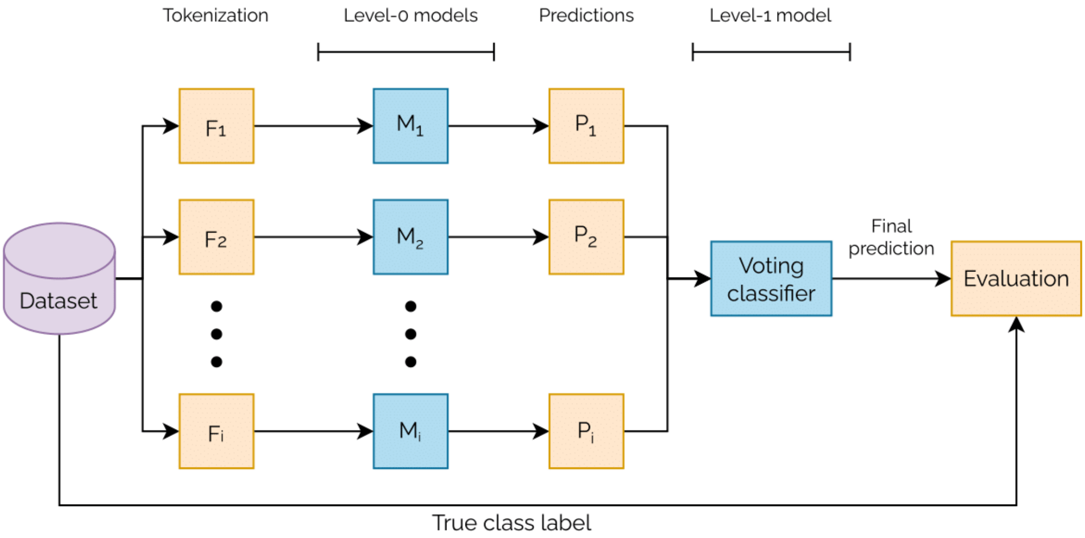


Figure 7.4: Ensemble architecture using a voting classifier as the meta-classifier.

7.3.2 Neural Network Meta-Classifier Architecture

The stacked neural network meta-classifier architecture is similar to the voting classifier architecture, but instead of using a voting classifier as the level-1 model, a neural network is trained on the predictions from the level-0 models to output a final prediction. An overview of this architecture can be found in Figure 7.5. The neural network was first trained using the predictions from the training dataset of the level-0 models as input. Then, the level-0 models made their predictions on the test dataset. The output of these predictions was used as input to the trained neural network to obtain the final predicted label. The input dimension of the network can be described as $num\_stacked\_models * num\_classes = input\_dimension$. The first hidden layer in the network is a dense layer with the ReLU activation function, followed by a dropout layer, another dense layer with a ReLU activation function, and finally, a softmax activation output layer. The output of the softmax layer is an array with the probability that the text belongs to a given label. The label with the highest probability is chosen as the final class prediction from the neural network.

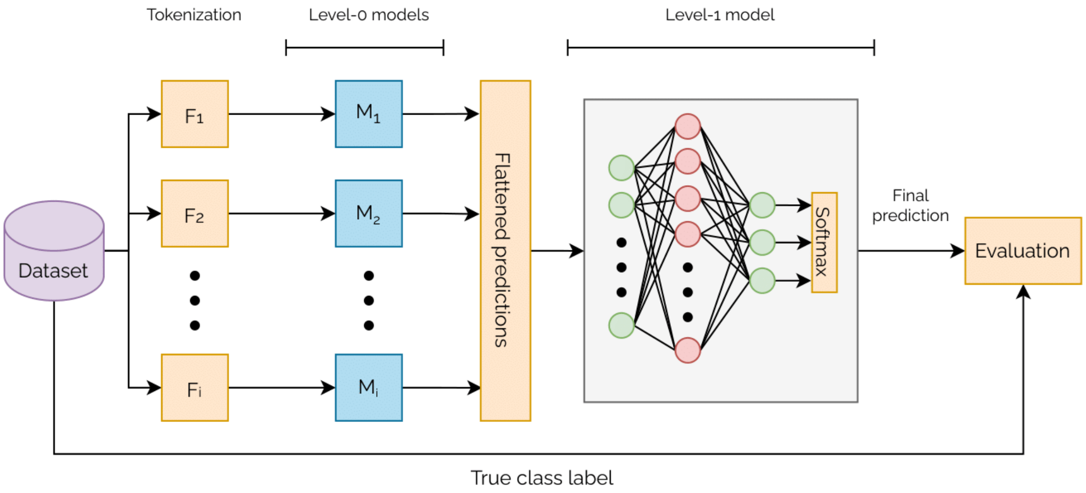


Figure 7.5: Ensemble architecture using a neural network as the meta-classifier.

8 Experiments and Results

The following chapter will cover all of the conducted experiments and their results. The first part of the chapter will include the experimental plan, where the experiments are presented, and what Research Question the experiments are designed to answer. The second part includes the experimental setup and will cover the technologies used. The last part will present the results of all the experiments conducted in this study. These results will be further discussed in Chapter 9.

8.1 Experimental Plan

To keep the experiments structured, an experimental plan is developed. The research's experimental plan consists of three parts. The first part will describe the creation of the ensemble model, which is trained and tested on the manually annotated Twitter dataset (**Dataset T**). The second part will present the experiment where the ensemble from part one is tested on the manually annotated Reddit dataset (**Dataset R**). Lastly, the plan for the experiment with the ensemble model trained on the semi-automatically annotated Twitter dataset (**Dataset S**) will be presented. All of the datasets used are described in Chapter 5.

8.1.1 Experiment 1 - Ensemble Model for Classification of Pro-Eating Disorder Users on Twitter

The first experiment aims to answer Research Question 4, presented in Section 1.4 (Page 7), by fine-tuning, combining and testing an SVM and five pre-trained attention-based models, which are described in Section 7.2. These models will then be used to create the ensemble model presented in Section 7.3. All models will be trained on the two versions of the manually annotated dataset from Twitter, namely **Dataset T** and the **Dataset T***. **Dataset T*** is included to see if the distribution of labels in the dataset has any impact on the models' performance. The experiment will be conducted both as a multiclass problem with labels *unrelated*, *pro-ED* and *pro-recovery*, and as a binary classification problem with the labels *non-pro-ED* and *pro-ED*, where the *pro-recovery* and *unrelated* posts will belong to the *non-pro-ED* class.

Fine-Tuning and Hyperparameter Optimization

An SVM and five attention-based models will be trained and fine-tuned for this experiment. The SVM will be tested with six features and different combinations of these. The features are: username, screen name, biography, tweet length, the tweet itself, and tweet sentiment. A grid search with 5-fold cross-validation will be performed to optimize the hyperparameters of the SVM and find the best combination of parameters to use when training the model.

The five attention-based models are presented in Section 2.2 and the architectures are described in 7.2. The models are BERT, DistilBERT, RoBERTa, BERTweet, and ERNIE 2.0. For BERT and ERNIE 2.0, both the base version and the large version will be tested. The experiment will include hyperparameter optimization for all models, both when fine-tuning the models with the training subset of **Dataset T**, and when fine-tuning with the training subset of **Dataset T***. The training datasets will be divided into a training and validation set with a 95/5-split to keep as much data as possible for training. Devlin et al. (2018) suggested a set of parameters when experimenting with BERT, as seen in Table 7.1. Experiment 1 will try the different batch sizes suggested: 16 and 32, and the learning rates of 5e-5 and 2e-5. Additionally, a learning rate of 1e-5 will be included in the hyperparameter optimization, as it was used by Nguyen et al. (2020) for BERTweet. For the other hyperparameters, the default for each model will be chosen. Lastly, the models will be trained for four epochs, except for BERTweet where Nguyen et al. (2020) reported increased performance for larger epochs, and will therefore also be trained with 10 and 30 epochs to investigate if this impacts the performance of the classifier.

Testing

A test dataset based on **Dataset T** will be created by splitting the dataset from Section 5.3.1 (Page 50) with a stratified 90/10-split. By using a stratified split, the original label distribution will be kept in both the test and train datasets. The models will be tested on two tasks: a multiclass task with the classes *pro-ED*, *pro-recovery*, and *unrelated*, and a binary task with the classes *pro-ED* and *unrelated*. Precision, recall, accuracy, and F_1-score (described in Section 2.3 Page 26) will be used as performance measures. Both the macro average and macro weighted average F_1-score is calculated and used to check performance.

Building the Ensemble

When all the models have been tested, two ensembles will be made by combining the best performing version of each model. In other words, the BERT model with the hyperparameter settings that performed best will be used, the SVM with the best combination of features, etc. The two different ensemble architectures consist of one ensemble with a voting classifier as the final prediction layer and the second with a neural network as the final predictor. The voting classifier will only be tested with the soft voting configuration. The best parameters for the neural network will be found using a 5-fold cross-validation grid search, which will experiment with the learning rate, batch size, number of epochs, dropout rate, and size of hidden layers. Further, ensembles with a different number of participating models will also be implemented. First, the ensemble will consist of seven models: the best performing SVM, the best performing of each attention-based model trained on **Dataset T**, and the best performing model trained on **Dataset T***. Further, the attention-based models with the poorest and second poorest performance are removed to create a new ensemble with six and five models, respectively. Lastly, the baseline SVM will be removed. A comparison between the four different sizes will be conducted to find the optimal number of models in the ensemble. The SVM is removed last because it is trained on other types of features than just the post text, which potentially can be a fine addition to an ensemble learner.

8.1.2 Experiment 2 - Ensemble Model Trained on Twitter Data and Tested on Reddit Data

Experiment 2 aims to answer Research Question 4 and will explore how a model trained on data from one social media platform performs when classifying data from another social media platform. In this experiment, the model is trained on **Dataset T** and tested on **Dataset R**. Once again, an ensemble model will be made by combining the best performing version of each model presented in Section 7.3. The same dataset and the same collection of models that were used in Experiment 1 will be used in this experiment, i.e., the best performing configurations for each model in Experiment 1 will also be participating in the ensemble of Experiment 2. The procedure for building the ensemble presented in Section 8.1.1 will also be conducted for Experiment 2. **Dataset R** was described in Section 5.5.2 (Page 64). The ensemble will only be tested on the multiclass task. The performance measures used in Experiment 1 will also be used in this experiment.

8.1.3 Experiment 3 - Models Trained on Semi-Automatically Annotated Twitter Data

The last experiment is conducted in order to investigate if the amount of data is more important than the quality of the data when creating a pro-ED classifier. The best model hyperparameters from Experiment 1 will be used in this experiment, except that the models will be trained on **Dataset S**. These models will be tested on both the test set from **Dataset T** and **Dataset R** to compare the performance to the results from Experiment 1 and 2. No ensemble learner will be developed for this experiment.

8.2 Experimental Setup

This section will introduce the experimental setup with the aim of making the experiment reproducible. The experimental design will cover the implementation details, including configurations, tools, libraries, and hyperparameters. The source code for the experiment is available on GitHub.[1]

8.2.1 SVM Implementation

The SVM model is based on the baseline SVM model created in Nornes and Gran (2019) and in the preliminary study (see Chapter 6). The implementation was done using the Linear Support Vector Classification model from the machine learning library Scikit-learn with default parameters.[2] This model applies a linear kernel to the SVM algorithm. A grid search revealed that the model performed better when the *class_weight* hyperparameter was set to 'balanced'. The text documents are tokenized using the TfidfVectorizer from Scikit-learn,[3] which takes care of both creating n-grams and removing stopwords. The TfidfVectorizer was initialized with the following parameters for the tweet texts and the biographies:

- ngram_range=(1, 2)
- stopwords='english'
- max_features=8000

For both the username and the screen name, the following parameters were used for the TfidfVectorizer:

- ngram_range=(2, 4)
- analyzer='char'
- max_features=8000

The sentiment score for each tweet was calculated using the Natural Language Toolkit (NLTK) sentiment package.[4] Tweet length was calculated on word-level.

8.2.2 Attention-Based Implementation

The implementation of the attention-based models was done using the Transformers library from Huggingface and Tensorflow Keras API.[5] The Transformers library includes

[1] `https://github.com/eirikdahlen/MSc-Computer-Science-2021`

[2] `https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html`

[3] `https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html`

[4] `https://www.nltk.org/api/nltk.sentiment.html#module-nltk.sentiment`

[5] `https://www.tensorflow.org/api_docs/python/tf`
`https://huggingface.co/transformers/`

several out-of-the-box pre-trained BERT-like models, and in the experiments in this Master's Thesis the Tensorflow models for sequence classification were used.[6] The models were trained using the Adam optimizer from Tensorflow,[7] and by using the default loss functions for the sequence classifiers, which was a sparse categorical cross entropy loss. The training data is split into a training and validation set with a 95/5-split, allowing the models to train on as much data as possible. The tokenizers for the respective models were used to create the tweet feature encodings.[8]

8.2.3 Stacked Ensemble Implementation

The two stacked ensemble learners were implemented with the already fine-tuned models but with different meta-classifiers on top. The first meta-classifier was a voting classifier that simply collected the predictions from each model and chose the label with the highest sum of votes. The second meta-classifier was a neural network created using the Sequential model with custom-made layers from the Tensorflow and Keras API. [9] A grid search was conducted with a 5-fold cross-validation to perform hyperparameter optimization. The following parameters were included in the grid search:

- Batch size: (16, 32)
- Learning rate: (0.01, 0.001)
- Epochs: (25, 50)
- Number of neurons in layer 1: (512, 1024)
- Number of neurons in layer 2: (256, 512)
- Dropout rate: (0.2, 0.3)

After the grid search was completed, the best parameters was chosen for the final meta-classifier. During training, a learning rate scheduler from Tensorflow was used, with a

[6] https://huggingface.co/transformers/model_doc/bert.html#transformers.TFBertForSequenceClassification
https://huggingface.co/transformers/model_doc/distilbert.html#tfdistilbertforsequenceclassification
https://huggingface.co/transformers/model_doc/roberta.html#robertaforsequenceclassification
https://huggingface.co/nghuyong/ernie-2.0-en
https://huggingface.co/vinai/bertweet-base

[7] https://www.tensorflow.org/api_docs/python/tf/keras/optimizers/Adam

[8] https://huggingface.co/transformers/model_doc/bert.html#berttokenizerfast
https://huggingface.co/transformers/model_doc/auto.html#autotokenizer
https://huggingface.co/transformers/model_doc/roberta.html#robertatokenizer
https://huggingface.co/transformers/model_doc/distilbert.html#distilberttokenizerfast

[9] https://www.tensorflow.org/guide/keras/sequential_model

custom made time-based decay.[10] The decay was defined as

$$decay = \frac{initial\ learning\ rate}{number\ of\ epochs},$$

and each new learning rate (lr) was calculated as

$$lr_{i+1} = \frac{lr_i}{1 + decay * epoch_i}.$$

8.2.4 Environment and Resources

The fine-tuning of the attention-based models was conducted on NTNU's high performance computing cluster IDUN (Själander et al., 2019), a cluster that enables the use of NVIDIA Tesla V100 and P100 GPUs. The fine-tuning of the regular base-models typically lasted for 2-3 hours, while the fine-tuning of the larger versions lasted for 5-8 hours.

8.3 Experimental Results

This section will introduce the results from the conducted experiments in this Master's Thesis. The performance of the models which were fine-tuned on both the manually annotated Twitter dataset (**Dataset T**) and the balanced version of this dataset (**Dataset T***) will be presented first. Thereafter, the results of the same models tested on the Reddit dataset (**Dataset R**) will be examined. Lastly, the results from the models fine-tuned on the semi-automatically annotated Twitter dataset (**Dataset S**) are presented. The systems' performance will be evaluated by the metrics introduced in Section 2.3 (Page 26), namely precision, recall, F_1-score, and accuracy.

8.3.1 Experiment 1 - Results

For Experiment 1, each model was trained over several runs with different hyperparameters and feature groups. The results that are displayed here will only include the best version of each model on the multiclass task, both when trained on **Dataset T** and **Dataset T***. The best performing models and their hyperparameters are presented in Table 8.1. The performances on the binary classification task will also be presented.

Models Trained on Dataset T

An overview of the best performing models trained on **Dataset T** can be found in Table 8.2. Here, SVM_1 was trained with the TF-IDF scores of unigrams and bigrams of biographies and tweets, while SVM_2 was trained with the TF-IDF scores of unigrams and

[10] `https://www.tensorflow.org/api_docs/python/tf/keras/optimizers/schedules/LearningRateSchedule`

Models	Batch Size	Learning Rate
BERT$_{\text{BASE}}$	32	5e-5
BERT$_{\text{LARGE}}$	32	1e-5
BERTweet	16	1e-5
DistilBERT	32	5e-5
ERNIE 2.0$_{\text{BASE}}$	32	2e-5
ERNIE 2.0$_{\text{LARGE}}$	32	2e-5
RoBERTa	16	1e-5

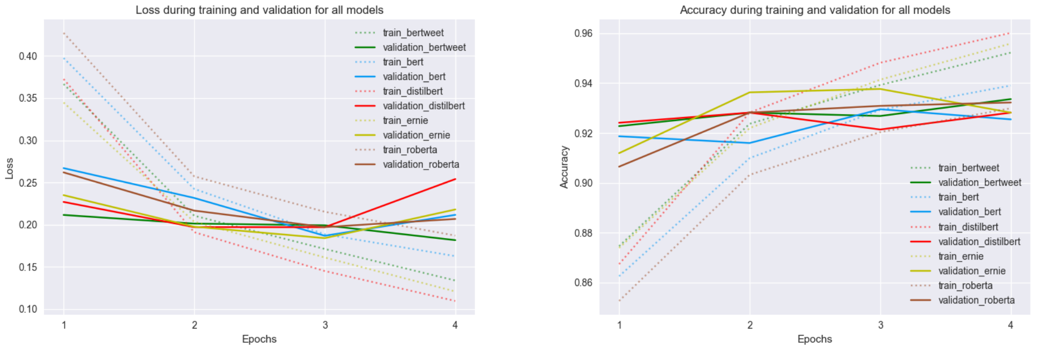
Table 8.1: Overview of hyperparameters for the best models fine-tuned on Dataset T.

Models	Unrelated			Pro-ED			Pro-recovery			Macro	Weight.
	P	R	F1	P	R	F1	P	R	F1	Avg F1	Avg F1
BERT$_{\text{BASE}}$	0.968	0.953	0.961	0.814	0.863	0.838	0.714	0.748	0.731	0.843	0.927
BERT$_{\text{LARGE}}$	**0.976**	0.939	0.957	0.758	**0.932**	0.836	**0.828**	0.759	0.792	0.862	0.928
BERTweet	0.968	0.959	**0.963**	0.837	0.888	**0.862**	0.808	**0.875**	**0.796**	**0.874**	**0.937**
DistilBERT	0.971	0.955	**0.963**	0.818	0.867	0.842	0.717	0.757	0.736	0.847	0.930
ERNIE 2.0$_{\text{BASE}}$	0.966	0.956	0.961	0.840	0.819	0.829	0.683	0.804	0.738	0.843	0.927
ERNIE 2.0$_{\text{LARGE}}$	0.960	**0.962**	0.961	**0.858**	0.803	0.830	0.708	0.794	0.749	0.846	0.927
RoBERTa	0.968	0.950	0.959	0.774	0.851	0.811	0.776	0.769	0.772	0.847	0.924
SVM$_1$	0.947	0.954	0.951	0.789	0.843	0.816	0.790	0.598	0.681	0.816	0.913
SVM$_2$	0.948	0.947	0.948	0.784	0.831	0.807	0.766	0.673	0.716	0.824	0.911

Table 8.2: Best performing models trained on **Dataset T** from Experiment 1.

bigrams of tweets only. The highest weighted macro average F$_1$-score was 0.937, produced by the BERTweet model. The same model also had the highest F$_1$-score for all three classes individually. BERT$_{\text{LARGE}}$ achieved the highest recall value for the pro-ED class overall. There seemed to be no obvious optimal batch size or learning rate for this task, as all variations of batch sizes and learning rates that were investigated were represented among the best models. Naturally, the models struggled more with the pro-ED and the pro-recovery classes, as these were underrepresented in the dataset. The training and validation losses and accuracies were tracked during the fine-tuning procedure for all models and are illustrated in Figure 8.1. For some models, the loss increased, and accuracy decreased on the validation data between epoch 3 and 4. This indicates that the optimal number of epochs may vary across models and supports the findings of Devlin et al. (2018) who recommended fine-tuning the models for either three or four epochs. Another explanation could be that the size of the validation dataset was only 5 % of the training set and that for some epochs, this would lead to random jumps in the metric values.

Another interesting finding was that DistilBERT, which only has 66M parameters, outperformed both BERT$_{\text{LARGE}}$ and ERNIE 2.0$_{\text{LARGE}}$ (both with 340M parameters) with regards to weighted macro average F$_1$-score. This supports the findings from the preliminary study, where DistilBERT outperformed BERT on the task of classifying



(a) Training and validation loss.

(b) Training and validation accuracy.

Figure 8.1: Loss and accuracy during training and validation on Dataset T.

pro-ED users on Twitter. The worst performing attention-based model was RoBERTa, which was outperformed by BERTweet with a difference of 0.13 in weighted macro average F_1-score. However, all the attention-based models performed better than the two baseline SVM models. SVM_2 obtained the highest weighted macro average F_1-score of the two, while SVM_1 achieved a better macro average F_1-score because it was slightly better at classifying pro-recovery tweets. Both are listed because the model trained on both tweets and biographies was be used in the ensemble learner.

One finding that is not directly related to the performance measures for each model is how much information was lost during tokenization. As mentioned in Section 6.2.2 (Page 80), 98 % of the texts in the preliminary study were truncated during tokenization, resulting in a substantial loss of data. This was due to the maximum sequence length of 512 for the attention-based models. The same overview for this experiment can be found in Table 8.3, and show that doing classification on post-level instead of user-level (as in the preliminary study) results in no data loss at all for the Twitter posts. This will be further discussed in Section 9.1.2 (Page 111).

Tokenizer	**Average number of tokens**	**Posts with number of tokens >512**
WordPiece	34	0 %
Byte-Pair Encoding	35	0 %

Table 8.3: Results from tokenization of posts in Dataset T.

The rest of the results from Experiment 1 can be found in Appendix B.1. For the attention-based models, all possible combinations of batch sizes (16, 32) and learning rates (5e-5, 2e-5, 1e-5) were tried. However, some of them resulted in the models classifying all tweets as the majority *unrelated* class, meaning that the models did not converge. The results from these runs are not included in the tables.

Because BERTweet was the best performing model, another training round for this model was conducted out of curiosity. Nguyen et al. (2020) reported that they fine-tuned BERTweet for 30 epochs to achieve their results. Therefore, two models were trained for 10 and 30 epochs, respectively, to investigate if drastically increasing the number of epochs had any impact on the model's performance. The results can be found in Appendix B.1, together with a figure displaying the loss and accuracy for each training epochs. The BERTweet model that was fine-tuned for 30 epochs achieved a weighted macro average F_1-score of 0.932, which was a strong, but not better than the already highest performing BERTweet model. Experimenting with a higher number of training epochs was thus discarded for the rest of the experiments.

Models	**Batch Size**	**Learning Rate**
BERT$_{\text{BASE}}$	32	2e-5
BERTweet	32	1e-5
DistilBERT	16	2e-5
ERNIE 2.0$_{\text{BASE}}$	32	2e-5
RoBERTa	16	2e-5

Table 8.4: Overview of hyperparameters for the best models on Dataset T*.

Models Trained on Dataset T*

The same models were also trained on **Dataset T*** with the same exploration of hyperparameters. The best performing configuration for each model is presented in Table 8.4, while the results can be found in Table 8.5. The highest scoring model from this training was DistilBERT, with a weighted macro average F_1-score of 0.926. This model also achieved the highest F_1-score for both the pro-ED and the pro-recovery classes. In addition, the DistilBERT fine-tuned for **Dataset T*** outperformed the best RoBERTa model that was fine-tuned for **Dataset T**. In contrast to the findings from training the models on **Dataset T**, BERTweet achieved the worst performance of the attention-based models, also surpassed by the best baseline SVM. In other words, BERTweet did not seem to perform well when given a smaller amount of data on this task. None of the large model versions were explored for this part of the experiment. A complete list of the result from training on **Dataset T*** can be found in Appendix B.1.

Ensemble Learner Results

After fine-tuning the models, they were put together in an ensemble learner with two different meta-classifiers on top. The voting classifier was tested with several models using a soft voting approach, while the feed-forward neural network (FFNN) experimented with batch size, dropout rate, number of epochs, learning rate, number of neurons in layer 1, number of neurons in layer 2, and number of models. All models are marked with a number n (like FFNN$_n$), where n is the number of models in the ensemble. When $n = 7$,

Models	**Unrelated**			**Pro-ED**			**Pro-recovery**			**Macro**	**Weight.**
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**Avg F1**	**Avg F1**
BERT$_{\text{BASE}}$	0.966	**0.952**	**0.959**	**0.800**	0.851	0.825	0.727	0.748	0.737	0.840	0.924
BERTweet	0.957	0.928	0.942	0.712	0.823	0.764	0.710	0.710	0.710	0.805	0.900
DistilBERT	0.969	0.948	0.958	0.777	0.884	0.827	0.792	0.748	**0.769**	**0.852**	**0.926**
ERNIE 2.0$_{\text{BASE}}$	**0.976**	0.937	0.956	0.788	**0.896**	**0.838**	0.685	**0.794**	0.736	0.843	0.924
RoBERTa	0.970	0.924	0.946	0.697	**0.896**	0.784	**0.808**	0.741	0.773	0.834	0.910
SVM$_1$	0.958	0.931	0.945	0.736	0.863	0.795	0.730	0.682	0.705	0.815	0.906
SVM$_{\text{BASE}}$	0.961	0.917	0.938	0.713	0.867	0.783	0.679	0.710	0.694	0.805	0.899

Table 8.5: Best performing models trained on **Dataset T*** from Experiment 1.

all attention-based models trained on **Dataset T** was included in the ensemble, together with the baseline SVM and the best DistilBERT trained on **Dataset T***. When $n = 6$, the best RoBERTa model was removed from the ensemble, while the best DistilBERT trained on **Dataset T*** was removed when $n = 5$. Lastly, the baseline SVM model was removed when $n = 4$. A grid search was conducted for the FFNN for each n-value, finding the optimal hyperparameters for each composition of models in the ensemble. Table 8.6 show an overview of the optimal hyperparameters for each of the FFNN ensembles.

Models	**FFNN$_4$**	**FFNN$_5$**	**FFNN$_6$**	**FFNN$_7$**
No. Models in Ensemble	4	5	6	7
Epochs	50	25	25	25
Batch Size	16	32	16	32
Learning Rate	0.001	0.01	0.01	0.01
No. Neurons Layer 1	1024	512	1024	512
No. Neurons Layer 2	512	1024	256	1024
Dropout Rate	0.2	0.2	0.2	0.2

Table 8.6: Overview of hyperparameters for the FFNN in the ensemble model for Experiment 1.

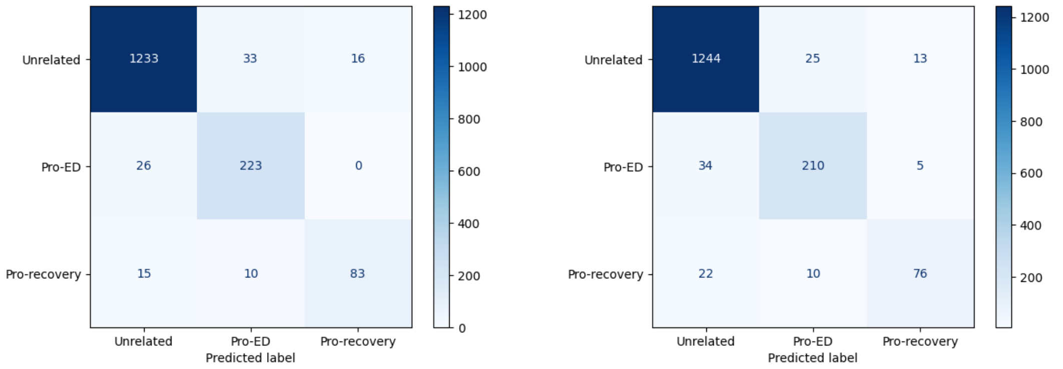
The results obtained by the ensemble learner architectures can be found in Table 8.7. The voting classifier with $n = 5$ achieved the highest macro average F$_1$-score (0.878) and weighted macro average F$_1$-score (0.939) across all models in Experiment 1. Additionally, it had the highest F$_1$-score for each individual class as well. This was also the only ensemble that performed better than the best BERTweet model. All the voting classifiers outperformed the neural networks in this experiment, as the neural networks had problems learning from the pro-recovery class especially.

The confusion matrix for the best voting classifier ($n = 5$) and the best FFNN ($n = 4$) is found in Figure 8.2. The voting classifier correctly classified more pro-ED and pro-recovery posts than the neural network in total. Additionally, the voting classifier never

Models	Unrelated			Pro-ED			Pro-recovery			Macro	Weight.
	P	R	F1	P	R	F1	P	R	F1	Avg F1	Avg F1
Soft Voting$_4$	**0.969**	0.959	0.964	0.839	**0.900**	**0.868**	0.808	**0.778**	0.792	0.875	0.938
Soft Voting$_5$	0.968	0.962	**0.965**	0.838	0.896	0.866	**0.838**	0.769	**0.802**	**0.878**	**0.939**
Soft Voting$_6$	0.965	0.959	0.962	0.824	0.884	0.853	0.816	0.741	0.777	0.864	0.934
Soft Voting$_7$	**0.969**	0.960	0.964	0.822	0.892	0.855	0.827	0.750	0.786	0.869	0.936
FFNN$_4$	0.957	0.970	0.964	**0.857**	0.842	0.850	0.809	0.704	0.752	0.855	0.932
FFNN$_5$	0.933	**0.973**	0.953	0.854	0.819	0.836	0.839	0.481	0.612	0.800	0.912
FFNN$_6$	0.928	0.970	0.949	0.850	0.771	0.808	0.699	0.472	0.564	0.774	0.902
FFNN$_7$	0.947	0.967	0.957	0.848	0.807	0.827	0.783	0.667	0.720	0.835	0.921

Table 8.7: Results from training the ensemble learner with different meta-classifiers in Experiment 1.

mistook a pro-ED post for being pro-recovery. The neural network's biggest struggle was miss-classifying pro-ED posts as unrelated.



(a) Confusion matrix voting classifier ($n = 5$).

(b) Confusion matrix FFNN ($n = 4$).

Figure 8.2: Confusion matrices for the two meta-classifiers.

Binary Classification Task

In addition to training and testing models on the multiclass task, the best performing models were also trained and tested on a binary classification task where the unrelated and the pro-recovery categories were merged. As pro-ED is the most interesting class for this study, it was desirable to create a system that was effective at detecting such content. Only the best performing models from the multiclass task were trained for this purpose.

The results from the binary classification task can be found in Table 8.8. The best performing model was DistilBERT, with a weighted macro average F$_1$-score of 0.950. This was an improvement from the highest score on the multiclass task. The best BERTweet from the multiclass task achieved the highest recall value for the pro-ED class (0.912) but was not as good overall for the binary classification task. This can be explained by the fact that the BERTweet model was better at detecting pro-recovery posts than the other models for the multiclass task and that this advantage was evened out when the pro-recovery class was merged with the unrelated class. All attention-based models outperformed the baseline SVM models on this task as well.

Models	**Non-Pro-ED**			**Pro-ED**			**Macro**	**Weight.**
	P	**R**	**F1**	**P**	**R**	**F1**	**Avg F1**	**Avg F1**
BERT$_{\text{LARGE}}$	0.970	**0.970**	**0.970**	0.831	0.831	0.831	0.901	0.949
BERTweet	**0.984**	0.949	0.966	0.762	**0.912**	0.830	0.898	0.945
DistilBERT	0.971	0.970	**0.970**	**0.833**	0.839	**0.836**	**0.903**	**0.950**
RoBERTa	0.972	0.949	0.960	0.748	0.847	0.795	0.878	0.935
ERNIE 2.0$_{\text{LARGE}}$	0.964	**0.971**	0.968	**0.833**	0.799	0.816	0.892	0.945
SVM$_1$	0.969	0.945	0.957	0.731	0.831	0.778	0.868	0.930
SVM$_2$	0.975	0.943	0.959	0.731	0.863	0.792	0.875	0.933

Table 8.8: Results from training the models on the binary classification task for Dataset T.

8.3.2 Experiment 2 - Results

The best performing models from Experiment 1 were included in Experiment 2 to make predictions on the Reddit test dataset (**Dataset R**). The results are displayed in Table 8.9. A natural drop in performance can be seen across all the models. The best performing model was ERNIE 2.0$_{\text{LARGE}}$, which achieved a weighted macro average F$_1$-score of 0.807. The ERNIE 2.0 model clearly outperforms all the other models in this experiment. BERT$_{\text{LARGE}}$ was surprisingly the weakest model of all, by achieving a weighted macro average F$_1$-score of 0.682, which was lower than the SVM$_2$ baseline. The DistilBERT trained on **Dataset T*** outperformed the RoBERTa model trained on **Dataset T** in this experiment also. However, the RoBERTa model produced the highest recall value on the pro-ED class with 0.899.

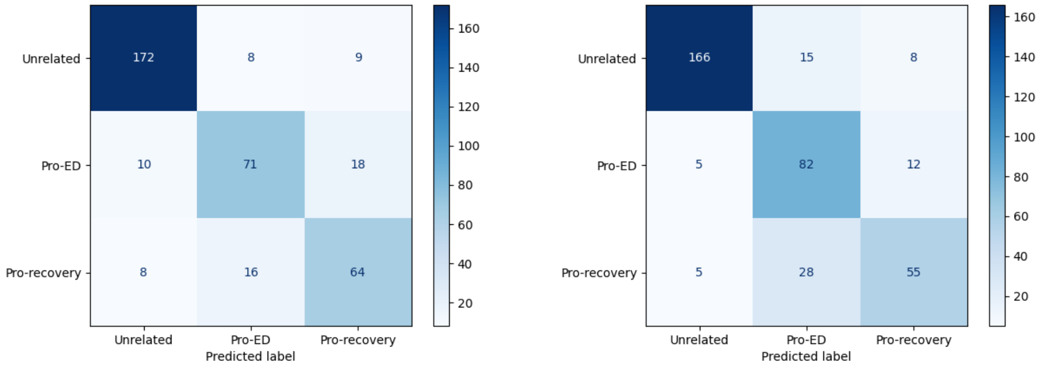
Models	**Unrelated**			**ProED**			**Pro-recovery**			**Macro**	**Weight.**
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**Avg F1**	**Avg F1**
BERT$_{\text{LARGE}}$	0.956	0.688	0.800	0.452	0.859	0.592	0.712	0.420	0.529	0.640	0.682
BERTweet	0.897	**0.878**	0.888	0.599	0.828	0.695	**0.778**	0.477	0.592	0.725	0.768
DistilBERT	0.927	0.810	0.864	0.676	0.737	0.705	0.592	0.693	0.639	0.736	0.770
DistilBERT$_{\mathbf{T*}}$	0.981	0.799	0.880	0.525	0.848	0.649	0.694	0.489	0.573	0.701	0.748
ERNIE 2.0$_{\text{LARGE}}$	0.970	0.841	**0.901**	**0.692**	0.747	**0.718**	0.648	**0.773**	**0.705**	**0.775**	**0.807**
RoBERTa	**0.987**	0.815	0.893	0.506	**0.899**	0.647	0.773	0.386	0.515	0.685	0.740
SVM$_2$	0.917	0.820	0.866	0.516	0.818	0.633	0.700	0.398	0.507	0.669	0.721

Table 8.9: Results from testing the models on Dataset R.

Two ensemble learners with a voting classifier and an FFNN as meta-classifier was created for Experiment 2 as well. The meta-classifiers used the same settings as the best performing ensemble from Experiment 1, namely $n = 5$ for the voting classifier and $n = 4$

for the FFNN. As the ranking of performing models were different in Experiment 2 than in Experiment 1, a different combination of models were used. When $n = 4$, only the ERNIE 2.0$_{\text{LARGE}}$, BERTweet, DistilBERT, and DistilBERT trained on **Dataset T*** was included in the ensemble. The RoBERTa model was added when $n = 5$.

The results from the ensemble learners can be found in Table 8.10. The highest weighted macro average F$_1$-score was 0.816, produced by the ensemble learner with $n = 4$ and an FFNN as the meta-classifier. Once again, the best ensemble learner performed better than the other models did individually. In contrast to the results from Experiment 1, the voting classifier did not outperform the FFNN but was still better than the ERNIE 2.0$_{\text{LARGE}}$ model. The confusion matrices for both the best FFNN and voting classifier can be found in Figure 8.3. The voting classifier has correctly classified a higher number of pro-ED posts, as can also be seen by the higher recall value, but the FFNN is better at classifying pro-recovery posts. The matrices also show that the models struggle to distinguish between the pro-ED and pro-recovery class.



(a) Confusion matrix FFNN ($n = 4$).

(b) Confusion matrix voting classifier ($n = 4$).

Figure 8.3: Confusion Matrices for the two meta-classifiers when tested on Dataset R.

Models	**Unrelated**			**ProED**			**Pro-recovery**			**Macro**	**Weight.**
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**Avg F1**	**Avg F1**
Soft Voting$_4$	0.943	0.878	**0.910**	0.656	0.828	**0.732**	**0.733**	0.625	0.675	0.772	0.808
Soft Voting$_5$	**0.958**	0.852	0.902	0.630	**0.859**	0.726	0.726	0.602	0.658	0.762	0.799
FFNN$_4$	0.905	**0.910**	0.908	**0.747**	0.717	**0.732**	0.703	**0.727**	**0.715**	**0.785**	**0.816**
FFNN$_5$	0.903	0.889	0.896	0.698	0.677	0.687	0.660	0.705	0.681	0.755	0.791

Table 8.10: Results from ensemble learners tested on **Dataset R**.

An investigation of how much information was lost during tokenization was conducted for Experiment 2 as well. Table 8.11 shows the average number of tokens per post in **Dataset R** and how many posts were truncated during tokenization. In contrast to **Dataset T**, some data loss was experienced as 12.3 % of the posts were truncated. This is further discussed in Section 9.1.2.

Tokenizer	Average number of tokens	Posts with number of tokens >512
WordPiece	313	12.3 %
Byte-Pair Encoding	316	12.3 %

Table 8.11: Results from tokenization of posts in Dataset R.

Models	Unrelated			ProED			Pro-recovery			Macro	Weight.
	P	R	F1	P	R	F1	P	R	F1	Avg F1	Avg F1
BERT$_{\text{BASE}}$	**0.899**	**0.948**	**0.923**	**0.697**	**0.711**	**0.704**	0.879	0.269	0.411	0.679	0.856
BERTweet	0.897	**0.948**	0.922	0.694	0.703	0.699	0.879	0.269	0.411	0.677	0.854
DistilBERT	0.898	0.947	0.922	0.696	0.707	0.701	0.882	0.278	0.423	0.682	0.855
ERNIE 2.0$_{\text{BASE}}$	**0.899**	0.947	0.922	0.692	0.703	0.697	**0.889**	**0.296**	**0.444**	**0.688**	**0.857**
RoBERTa	0.898	**0.948**	0.922	0.696	0.707	0.701	0.879	0.269	0.411	0.678	0.855
SVM$_1$	0.898	0.945	0.921	0.692	0.695	0.693	0.816	0.290	0.428	0.681	0.854

Table 8.12: Results from training on **Dataset S** and testing on **Dataset T**.

8.3.3 Experiment 3 - Results

The final experiment included training the best performing models from Experiment 1 on **Dataset S** and do testing towards the test set from **Dataset T** and **Dataset R**. Only the BASE-versions of BERT and ERNIE 2.0 were trained for this experiment due to time restrictions. The results from testing the models on **Dataset T** can be found in Table 8.12. All models seemed to perform equally good, with ERNIE 2.0$_{\text{BASE}}$ achieving the highest weighted macro average F$_1$-score of 0.857. For this experiment, the baseline SVM$_1$ performed on the same level as the attention-based models. This is interesting, as deep learning neural networks usually excel compared to basic learners when the size of the training data increases. This will be further elaborated on in Section 9.1.2. The most notable performance drop was for the pro-recovery class, where the highest F$_1$-score was 0.444. Similar for all models was a high precision value and a very low recall value for the pro-recovery class.

Models	Unrelated			ProED			Pro-recovery			Macro	Weight.	Acc
	P	R	F1	P	R	F1	P	R	F1	Avg F1	Avg F1	
BERT$_{\text{BASE}}$	0.576	**0.984**	0.727	**0.868**	0.465	0.605	0	0	0	0.444	0.525	0.617
BERTweet	0.577	0.947	0.717	0.631	0.414	0.500	**1**	0.011	0.022	0.413	0.498	0.588
DistilBERT	0.859	0.836	0.847	0.455	0.869	0.597	0.667	0.023	0.044	0.496	0.593	0.654
ERNIE 2.0$_{\text{BASE}}$	**0.932**	0.873	**0.902**	0.482	**0.939**	**0.637**	**1**	0.068	0.128	**0.555**	**0.651**	**0.702**
RoBERTa	0.816	0.894	0.854	0.500	0.838	0.626	**1**	0.034	0.066	0.515	0.609	0.678
SVM$_1$	0.753	0.921	0.829	0.533	0.727	0.615	0.800	**0.091**	**0.163**	0.536	0.617	0.676

Table 8.13: Results from training on **Dataset S** and testing on **Dataset R**.

The results from testing on **Dataset R** can be found in Table 8.13. Here, accuracy is included as a measure as there were big differences between the weighted macro average F$_1$-score and the accuracy. ERNIE 2.0$_{\text{BASE}}$ is the best performing model with a weighted macro average F$_1$-score of 0.651 and an accuracy of 0.702. This was notably better than any of the other models. The second best model was, in fact, the baseline SVM$_1$ model with a weighted macro average F$_1$-score of 0.617. In this experiment, the BERTweet model was clearly outperformed by all the other models with a weighted macro average F_1-score of 0.498. The biggest struggle for all models was classifying pro-recovery posts, which was also the case when these models were tested on **Dataset T**. A confusion matrix for the ERNIE 2.0_{BASE} model can be found in Figure 8.4, and shows that the model struggled with the difference between pro-ED and pro-recovery data. This finding is even more supported by the high recall value and low precision value for the pro-ED class from the ERNIE 2.0_{BASE} model. Looking closer at the $\text{BERT}_{\text{BASE}}$ model, it did not classify *any* of the Reddit posts as pro-recovery. This, combined with a low precision value and high recall value on the unrelated class, indicates that it classified too many posts as unrelated instead of pro-recovery.

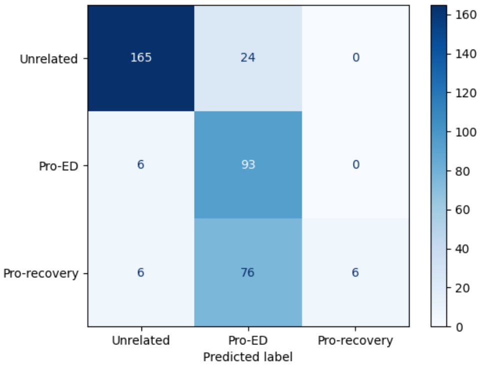


Figure 8.4: Confusion Matrix for the ERNIE 2.0 model trained on Dataset S and tested on Dataset R.

9 Discussion and Evaluation

This chapter presents an evaluation of the research done in this Thesis. In the first section, a discussion of the experimental results, the findings, and the limitations of this Master's Thesis will be provided. Secondly, an evaluation of the findings in light of the research questions formulated in Section 1.4 (Page 7) will be presented. Comparisons to relevant previous work will also be provided to underpin or contrast the discussion.

9.1 Discussion

Chapter 8 presented the three experiments conducted for the task of classifying pro-ED posts from social media. Four different datasets gathered from Twitter and Reddit were used to test the performance of several attention-based models and two ensemble learning architectures. Both the datasets and the findings from the experiments are considered the main contributions of this Thesis, which is also reflected in the research questions presented in Section 1.4 (Page 7). This section focuses on the choices made when working towards answering these research questions and will discuss the impact these choices had on the final contributions of this Thesis, as well as limitations that were found.

First, the collection, processing, and annotation of the datasets will be discussed. This includes discussion about the annotation criteria and how they suit the task of annotating pro-ED posts from Twitter and Reddit. Also, the composition of datasets is deliberated and includes both the label distribution and the amount of data. The second section of this chapter considers the models and experiments. The first part of the section considers hyperparameter optimization and discusses the choices that were made. Further, is the composition of the ensemble learners discussed regarding which model to include and the choice of meta-classifier. Lastly, a discussion about performance measures and limitations of the experiments will be presented.

9.1.1 Collection and Annotations of Datasets

In order to reach the goal of this Master's Thesis, a dataset with pro-ED posts from social media had to be collected. The collection and annotation procedures that were followed are described in Chapter 5 (Page 47) and resulted in four different datasets. As a reminder, the manually annotated Twitter dataset is denoted **Dataset T**, the balanced manually annotated Twitter dataset **Dataset T***, the semi-automatically Twitter dataset **Dataset S**, and the Reddit dataset is denoted **Dataset R**.

Section 5.5.2 (Page 61) briefly discusses the challenges encountered when collecting and annotating the datasets. For **Dataset T**, the first challenge was an unbalanced label distribution, as shown in Figure 5.4 (Page 62). The data collection procedure resulted in a lot of unrelated tweets from so-called *secondary* pro-ED users (users found through retweets in the collection procedure), which could have been avoided if the *primary* users were paid more attention to, as pointed out in Section 5.5.2. This led that a collection of posts from the unrelated tags and keywords never was conducted, as the number of unrelated posts available was substantial at this point.

Section 5.5.2 (Page 61) presents the datasets used for the experiments, and their label distributions can be found in Figure 5.4 (Page 62), 5.5 (Page 63), and 5.6 (Page 63). Unrelated is by far the most represented class in all the datasets, while pro-recovery is the least represented class. Even though it has been stated that the datasets were unbalanced after annotation, it does not mean that the collection and annotation procedures were not successful. There were two options to consider when deciding the distributions in the dataset; should the label distributions of the datasets mirror the label distribution of the social media platforms that they were collected from, or should the datasets be optimized for training classification systems. As pro-ED posts only contribute to a fraction of the total amount of posts on Twitter, a realistic dataset would consist of a huge amount of unrelated posts and only a handful pro-ED tweets. However, this would not be beneficial when the goal of this Master's Thesis is to develop a classification system that detects pro-ED posts. On the other hand, collecting a dataset where each class is equally represented is desired for most classification tasks. However, it would in this case give the models the impression that pro-ED posts occur far more often than they actually do. The label distribution in the datasets collected in this study tries to capture both options by aiming for a compromise between the two. Here, sufficient data was provided for the models to learn and recognize patterns in pro-ED posts, while at the same time letting pro-ED (and pro-recovery) being an underrepresented class. Additionally, the motivation for creating **Dataset T*** was to investigate if a higher share of pro-ED posts did in fact improve the performance of the classification systems. Unfortunately, as **Dataset T*** was the result of an undersampling procedure, the more balanced label distribution did not weigh up for the size reduction. However, investigating other distributions for pro-ED datasets could be examined in future research.

The annotation criteria used were strongly inspired by Giæver (2018) and only modified to suit the annotation of posts instead of users. Previously, when the criteria were used to classify users, it was enough that one of the criteria was identified in either the personal information about the user or one of the tweets. With the annotation of posts, the same criteria applied, but this time with a lot less data to consider in the evaluation of each data instance, as only the tweet itself was evaluated. Table 5.5 (Page 63) shows some of the tweets that were found hard to label. An unintended bias was introduced as the annotators knew that these posts were collected by using well-known pro-ED terms, and therefore it was easy to assume that these tweets had a pro-ED attitude. Although the annotators could assume that the label should be pro-ED given the context, the **tweets** themselves do not explicitly concern pro-ED or eating disorders in any way. Therefore, the tweets are labeled as *unrelated*. Additionally, tweets concerning eating disorders that did not explicitly talk about being **pro** should be, by definition, labeled as pro-recovery. It was not always the case that the annotators agreed with the criteria when encountering such tweets because it was found that the user did not state any attitude of being pro-recovery, or worse, the user was found to have a pro-ED attitude without qualifying for any of the pro-ED criteria, and was therefore labeled pro-recovery. A similar challenge was encountered with tweets mentioning nutritional diets because even though the annotators found the tweet to be pro-ED with respect to the context, the criteria for pro-ED did not cover the content of the tweet.

A phrase commonly encountered in tweets mentioning eating disorders or eating disorder habits was *not pro, just using tags for reach*, combined with several explicit pro-ED hashtags. Annotating these posts introduced a challenge; they expressed both a negative attitude towards the pro-ED community, while at the same time included pro-ED hashtags. Yom-Tov et al. (2012) explain this phenomenon as a way for pro-recovery users to spread content to pro-ED users because the content would more likely reach a pro-ED user. However, the definition of what is considered pro-ED might be highly subjective, and even though a user will not consider its own post being *pro*, it might fulfill the given pro-ED criteria and would therefore be labeled accordingly. A similar case was posts where the user self-identify as pro-ED but shows a positive attitude towards recovery on behalf of other users. Clearly, the posts should be labeled pro-ED as they satisfy criterion 1, but at the same time, the posts show a positive attitude about recovery for others, and therefore satisfy criterion 2 for pro-recovery (see Section 5.2 Page 48 for the criteria).

The pre-processing steps described in Section 5.4 (Page 53) modified the tweets such that some information could be lost. When responding to or tagging another Twitter user, the username was substituted with *MENTION*, and all URLs were substituted with *URL*. Especially the latter could provide useful information when determining the true label of a tweet. When a tweet is assumed to be, e.g., pro-ED, and the context is unknown, an URL (or its content) can either provide the context and contribute to making it qualify as being pro-ED, or the opposite, and thus reveal the true label.

Another aspect to consider when discussing the annotation criteria is to which degree they cover every aspect of the pro-ED community lingo. The annotators stumbled upon pro-ED related concepts that were not covered by the criteria for being pro-ED. Examples of such concepts are *binging* and *meanspo*. Binging could refer to either binge eating or binge watching, but even if the user means the former in a pro-ED supportive manner, it is not covered by the annotation criteria because the criteria focus on restrictive behavior. Tweets that were considered to be *meanspo*, but did not include the term *meanspo*, do also not qualify to be labeled as pro-ED. Similar inspiration content like *thinspo* are included in the criteria, and a solution could be to add *sharing of meanspiration content* as a criterion for being considered as pro-ED.

The annotation of Reddit posts unveiled dissimilarities with the annotation of tweets. As the Reddit posts are substantially longer (over 9x more words on average) than tweets, there is a lot more information in each post. As the limit for the number of characters is 40 000 characters, the users do not have to constrain themselves or be selective of what to include in order to express their thoughts. This resulted in posts where the context of the text was more clear to the annotator, which resolved many of the challenges encountered during the annotation of tweets, as discussed earlier. However, the main challenge during the annotation of Reddit posts was the presence of multiple attitudes in one single post. This challenge was briefly presented in Section 5.5.2 (Page 64) and concerns the classes *pro-ED* and *pro-recovery*. The challenging posts were often written by users who struggled or were still struggling with an eating disorder. In one part of the post, they could write about eating disorder behaviors and habits, and in the other part of the post about how the user had recovered or the desire to not fall out of recovery. It became clear early on in the annotation process that the classification criteria did not fit the nature of Reddit posts as well as they fitted tweets, which was natural as the criteria were originally developed for annotation of Twitter data. If the same procedure should be strictly followed for Reddit annotation, a post which contained one sentence that satisfied a pro-ED criterion would be labeled as pro-ED no matter what the other sentences in the post concerned. An agreement to label posts with the attitude that was the most dominant for the post was therefore made in order to obtain a dataset with labels that actually represented the content of the posts.

Although there were discovered challenges with the annotation process and criteria, many of them were edge cases and concerned only a small fraction of the dataset. Most times, the posts were easy to label, but when there was no immediate answer, a thorough consideration of each annotation criterion was needed to find the *correct* label. The inter-annotator agreement analysis did also support the fact that the criteria were suitable for annotating tweets, as the Cohen's Kappa score was categorized as *almost perfect* when considering the strength of agreement between the annotators, as defined by Landis and Koch (1977). As seen in Section 5.5.3 (Page 66), only 20 of the 1 870 tweets that were in both annotators datasets had been disagreed upon, while at the same time there had been uncertainty about the correct label (disagreement-reason number 2 and 3). The inter-annotator agreement analysis also supports the fact that annotating the Reddit posts was more difficult than annotating tweets. The Cohen's Kappa score was categorized as a *substantial* strength of agreement, which is weaker than the strength of agreement for the Twitter dataset.

Another topic worth discussing is what kind of errors and uncertainties are introduced to the process by the annotators themselves. As mentioned in Section 5.5.1 (Page 57), the annotators are two male students in their twenties with English as their second language. In addition, none of the annotators possessed any domain knowledge about the pro-ED communities nor any experience of annotating data prior to this study. Even though the inter-annotator agreement scores show a high level of agreement, this does not exclude the possibility that posts have been labeled equally, yet wrongly, by both annotators. Such errors are most likely present in the datasets due to the similarity in the demographic background and the level of prior knowledge of the annotators.

Dataset S is almost nine times larger than **Dataset T** and was created to investigate the impact of fine-tuning the models on data that were interpreted and annotated by a computer rather than by a human annotator. However, the algorithm used for the semi-automatic annotation (which was presented in Section 5.5.1 Page 57) can be considered simple. Firstly, there was a possible bias towards the pro-ED class, as a tweet was checked for pro-ED keywords before assessing if it belongs to any of the two other classes. As mentioned earlier, Yom-Tov et al. (2012) proposed that pro-recovery users apply similar words as pro-ED users in their posts. With the given annotation algorithm, there may be many examples of posts in **Dataset S** that should be pro-recovery, but end up as pro-ED instead. This problem would most likely be avoided in a manual annotation process. Secondly, only annotating on keyword occurrences ignores possible negations in the text. For instance, a posts with the text *Pro-ED content is very bad*, would be labeled as pro-ED because of the presence of the *pro-ED* keyword. Lastly, tweets from the Giæver (2018) dataset had to be added to **Dataset S** in order to collect enough tweets to obtain a similar label distribution as in **Dataset T**. Even though this study applies a very similar annotation and pre-processing procedure and uses many of the same search keywords as Giæver, there might be differences between **Dataset T** and **Dataset S** that impact the performances of the models. Despite this, the idea of using a semi-automatic annotated dataset should not be discarded, as a more thorough annotation process could lead to a dataset that is more similar to the manually annotated dataset.

9.1.2 Building the Classifier Systems

With four datasets and several models available for this study, it was possible to gain major insights into how to build a robust pro-ED classifier system. When building the different systems, a search for finding the optimal hyperparameters and features was conducted, especially for Experiment 1. However, the findings of hyperparameters and features from Experiment 1 were used throughout Experiment 2 and 3, even though there might have been other parameters that were more optimal for these specific tasks. Using the optimal parameters from Experiment 1 was done both because of time restrictions and the fact that all of the attention-based models performed quite similarly individually. For example, the best performing BERT$_{\text{BASE}}$ model trained and tested on **Dataset T** achieved a weighted macro average F$_1$-score of 0.927, while the worst achieved 0.920. As discussed earlier, there did not seem to be any optimal hyperparameter settings that performed better overall. Both batch sizes 16 and 32 and learning rates 5e-5, 2e-5, and 1e-5 were represented among the best performing models. These observations further justify why the best models and their configurations from Experiment 1 were used throughout Experiment 2 and 3.

On an individual level, the BERTweet model trained with batch size 16 and learning rate 1e-5 was the best performing model on the multiclass task when both trained and tested on **Dataset T**. Its weighted macro average F$_1$-score stands out from the rest of the performance scores in Experiment 1 but does not seem to be a *random* good result as the second best model was also a BERTweet model, with other hyperparameters. This supports the findings of Nguyen et al. (2020) where BERTweet outperformed other state-of-the-art models on NLP tasks on data from Twitter. The advantage of pre-training on Twitter text seemed crucial, as the other models that were pre-trained on more comprehensive and formal texts, like Wikipedia, were beaten by BERTweet. The results from the BERTweet model also support the findings of Guo et al. (2020), who found that BERTweet outperformed RoBERTa overall when applied to 25 social media text classification datasets. However, when the same BERTweet model was tested on Reddit data in **Dataset R**, it was beaten by both ERNIE 2.0 and DistilBERT. A plausible reason for this is that the language used on Reddit sometimes might be more similar to the language on Wikipedia than on Twitter, and thus BERTweet does not benefit from its special pre-training schema, as it did when testing it on Twitter data. In addition, BERTweet used the default sequence length of 128, as opposed to the other models, where a sequence length of 512 was used. This means that BERTweet had less data to work with during fine-tuning, as the average length of the Reddit posts was 313 tokens.

Both BERT$_{\text{LARGE}}$ and ERNIE 2.0$_{\text{LARGE}}$ outperformed their BASE-versions in Experiment 1, with BERT$_{\text{LARGE}}$ achieving the second highest macro average F$_1$-score of the models. It is natural to think that bigger models equal better performance for this task, as it has been shown in related experiments (Devlin et al., 2018; Sun et al., 2020). However, DistilBERT produced a higher weighted macro average F$_1$-score and accuracy than both the large models, even though DistilBERT has the smallest amount of parameters among the selected models. The results from the preliminary study (which were presented in Section 6.2 Page 80) support these findings when the model is applied to the task of classifying pro-ED users on Twitter. As described in Section 2.2.7 (Page 21), DistilBERT is created from the original pre-trained BERT model through knowledge distillation, and it is reasonable to assume that BERT would perform better than DistilBERT. However, these results are similar to the findings of Davidson et al. (2020), who fine-tuned BERT and DistilBERT on Reddit and Twitter data, and found that DistilBERT achieved the highest F$_1$-score. However, investigating the results of these three models from Experiment 2 introduced another interesting finding. ERNIE 2.0$_{\text{LARGE}}$ is clearly outperforming all other models with a weighted macro average F$_1$-score of 0.807, while DistilBERT and BERT$_{\text{LARGE}}$ produced a score of 0.748 and 0.682, respectively. The performance of the BERT model was also beaten by the baseline SVM, introducing doubts about whether something had gone wrong during the testing on **Dataset R**. However, a second run created the same results for the BERT model. The gap between ERNIE 2.0$_{\text{LARGE}}$ and BERT$_{\text{LARGE}}$ is substantial in this experiment and gives a strong indication that the pre-training schema for ERNIE 2.0 makes the model more robust when applied to data that is different from what it was fine-tuned on.

The findings from training the models on **Dataset T*** are up for further investigation. **Dataset T** had 14 750 tweets in the training dataset, with 15.2 % of these being pro-ED, while **Dataset T*** consisted of 8 976 tweets, with 25.0 % of these being pro-ED. The size difference between the datasets is considerable, 39.0 % of the data was removed during an undersampling process, but the number of pro-ED and pro-recovery posts are still the same. However, the models trained on **Dataset T*** did not perform better on the under-represented classes. This indicates that a larger dataset is preferable even though the amount of pro-ED data is not increased. The assumption is that by giving the models more unrelated data, they will be more sure of what posts are considered unrelated, but more importantly, what posts are considered not-unrelated.

In addition to the attention-based models, the baseline SVM was tested with several different feature groups. Some of these features were based on the findings from the data analysis in Section 5.6 (Page 68). Tweet length and sentiment score for tweets were found to differentiate between the three possible classes in the data analysis. However, when these features were added to the baseline model, the score did not increase, and these features were discarded. Further exploration of features could have been conducted, but both the findings in this Master's Thesis, Giæver (2018), and Nornes and Gran (2019) support that n-grams of tweets were the most influential features for the SVM.

The results from Experiment 3 shed new light on the performances of the models. When trained on **Dataset S** and tested on **Dataset T**, all models are performing at about the same level, including the baseline. This might imply that the biggest variance between models is found when the amount of training data is smaller, as supported by the findings of Dodge et al. (2020). Another thing worth discussing is the substantial performance drop for the pro-ED and pro-recovery classes. As mentioned earlier, the semi-automatic annotation process might have introduced differences between the pro-ED and pro-recovery classes in **Dataset T** and **Dataset S**. This is even more clear when looking at the results from training on **Dataset S** and testing on **Dataset R**, as all models struggle with the pro-recovery class. An overview of classifications done by the ERNIE 2.0 model in this experiment can be found in Table 9.1 (some of the Reddit posts are shortened for readability). Here, it is clear that the pro-ED keywords used to establish **Dataset S** have influenced the model to label the posts with ID 1 (keywords *thinspo* and *skinny*) and 2 (keywords *pro-ed* and *pro-ana*) as pro-ED instead of pro-recovery. However, post number 3 shows that the model sometimes understands that the underlying context of the post is pro-recovery, even though pro-ED keywords are present (*thinspo* and *proana*).

In general, the attention-based models outperformed the non-attention-based baseline SVM in every experiment. This is in line with several studies in recent years, as discussed in Section 4.3 (Page 41). The only exceptions were in Experiment 2, when BERT$_{\text{LARGE}}$ was outperformed, and in Experiment 3 where the SVM actually was the second best model when it was trained on **Dataset S** and tested on **Dataset R**.

The ensemble learner architectures showed promising results and improved the performances in both Experiment 1 and 2. The ensembles consisted of between four and seven models, but the composition could have been different. There are many ways to create

ID	Reddit post	ERNIE 2.0 predicted class	True class
1	i deleted my thinspo folder on my computer // i'm ten months in recovery. [...] i don't need to be skinny to feel comfortable in my body, and i wouldn't want to be skinny anyways.	Pro-ED	Pro-recovery
2	post request how can i help my girlfriend recover? get her away from the pro-ed sites? [...] she doesn't consider herself anorexic, despite constantly browsing pro-ana sites and so forth.	Pro-ED	Pro-recovery
3	[...] thinspo is thinspo regardless of your intentions. [...] its just so unnecessary and harmful and yet people still do it all the time like are you proana or are you just dumb. [...] you can spread awareness about eds without spreading thinspo.	Pro-recovery	Pro-recovery

Table 9.1: Sample of classifications done by the ERNIE 2.0 model trained on **Dataset S** and tested on **Dataset R**.

an ensemble learner, and while investigating all approaches was not possible for this study, different possibilities should be further discussed. For instance, including the SVM was an active choice to introduce diversity into the ensemble because it was trained with different features than the others (n-grams of both tweet and biography). This proved to be beneficial as it was a part of the best scoring ensemble, even though it was beaten by all the attention-based models in Experiment 1 and would therefore not be the natural choice to include before others. In addition, using models that are trained on different datasets or different parts of the same datasets could have been explored further, as this is a common ensemble architecture. The DistilBERT model trained on **Dataset T*** was included with this in mind, even though it did not improve the score. Another possible ensemble structure could have been not to use the best version of each model but simply the top performing models overall. For an ensemble with seven models ($M = 7$), this would result in an ensemble consisting of two BERTweet models, two BERT$_{\text{LARGE}}$ models, one DistilBERT model, and two ERNIE 2.0 models. Other possibilities include using a different weighting of the models. This could, for instance, be a fixed weighting where the best performing models had a bigger influence on the final prediction, but dynamical weighting approaches would also be interesting to investigate further. An example would be that the predictions from models that were especially good at classifying pro-ED posts were given a higher weighting in cases where there was uncertainty about the final class label being pro-ED or some other class. Although pre-training of language models is not as emphasized in this study as fine-tuning procedures, the inclusion of BERTweet in the ensemble makes this an interesting topic. As Experiment 2 and 3 include testing the models on other social media platforms than what they are fine-tuned for, a potential ensemble could include attention-based models pre-trained on several platforms. The idea of pre-training attention-based models on data from several social media platforms are also supported by Guo et al. (2020) as a measure to improve the performance on social media classification tasks. This could be combined with an adaptive weighting schema that changed each models' weights based on the social media platform that the ensemble would be applied to. Finally, other choices of meta-classifier or an ensemble of meta-classifiers could also have been further explored, as one is not restricted to using a voting classifier or an FFNN.

In general, all models seemed to struggle more with the under-represented classes, pro-ED and pro-recovery. In many cases, the best performing models stand out from the others when it comes to the classification of pro-recovery posts. However, this is not the class in focus, and the results could therefore be misleading. Most of the experiments are centered around the multiclass task in this study, and only Experiment 1 includes training and testing for the binary classification task. A possible different approach would be to use the binary task to find the best models for the rest of the experiments, keeping focus on the pro-ED class.

The same can be said for the performance measures. The weighted macro average F_1-score weights the scores based on the size of each class. A good score for the unrelated class might increase the weighted score more than what is desired. The regular macro average does not take the label distribution into account and could be seen as a better measure in this case. Additionally, both recall and F_1 for the pro-ED class are good candidates for measuring performance in this study. If the classifier system from this study were to be used in a real application on a social media platform, a focus on high recall score for the pro-ED class would be the go-to measurement. As pro-ED data only contributes to a small amount of all the data on social media platforms, it would be desirable for a system that filters out pro-ED posts to remove as many harmful posts as possible, at the expense of also removing some unrelated posts in the process.

An overview of how many posts that were truncated during tokenization were presented in Table 8.3 and Table 8.11 for **Dataset T** and **Dataset R** respectively. Compared with the findings from the preliminary study (see Table 6.4 Page 81), this shows that using a dataset collected on post-level instead of user-level results in no data loss for the Twitter dataset. As discussed, Reddit posts are substantially longer than tweets, but only 12.3 % of the posts are truncated during tokenization. This further supports the decision of collecting data on post-level in order to fully utilize the power of attention-based models for the task of classifying pro-ED content. This finding also shows that using the default sequence length of 512 could be decreased when fine-tuning on Twitter data. With an average length of 34 and 35 tokens per post for WordPiece and Byte-Pair Encodings, respectively, the sequence length could be set at 128 for future studies to save computational power and memory. For Reddit, keeping the sequence length of 512 is appropriate.

Not all training sessions in the experiments of this Thesis were successful. Many of the sessions ended with a model that did not converge, i.e., it did not learn and classified all posts as unrelated (the majority class). This was found to happen more often when the learning rate was 5e-5, which was the smallest learning rate tested in the hyperparameter search. There were also examples where the models seemed to learn the first epoch because the loss decreased, but when predicting on the validation dataset between epochs, they classified all posts as unrelated. In these cases, the loss did not decrease for the rest of the training. This supports the findings of Dodge et al. (2020) and Devlin et al. (2018) for small datasets. However, this problem also occurred during training on **Dataset S** which is of considerable size. The reason for this unstable training is hard to explain because it happened to nearly all models and for different configurations. One explanation could be that the models got stuck in a so-called local minimum and thought that classifying everything as unrelated was the best solution. This usually happens for smaller learning rates, which was also the case in the experiments. However, this does not imply that the lowest learning rate should not be used, as the best performing model on the binary classification task was a DistilBERT model fine-tuned with learning rate 5e-5. A common solution to avoiding getting stuck in a local minimum is to adjust the learning rates during the training session, as was done by Liu et al. (2019) with a linear learning rate warm up, followed by a linear decay to 0. Such implementations were not considered during the fine-tuning, as the learning rates used in this study were heavily inspired by the work of Devlin et al. (2018) and Nguyen et al. (2020), who used fixed learning rates for each run.

Another approach that is worth discussing is the use of different random seeds when training attention-based models. In this study, a fixed random seed is used when splitting the dataset into training and validation sets and when initializing the final classification layers used during the fine-tuning process, despite the findings of Dodge et al. (2020); Devlin et al. (2018); Nguyen et al. (2020); Risch and Krestel (2020). Additionally, cross-validation techniques during the fine-tuning procedures could have been explored. Such procedures were not carried out for the experiments in this study, as the focus was mainly on testing several hyperparameters, various models, and fine-tuning and testing on several datasets. The introduction of these techniques would further increase the complexity of the systems and is left for future work.

9.2 Evaluation

The goal of this Master's Thesis was to identify pro-ED posts from various social media platforms by applying attention-based models. Four research questions were established for this study and were considered sub-goals that helped guide the research in a structured manner towards the main goal. The following section is dedicated to evaluating the findings of this Thesis in light of each research question, and lastly, the main goal itself.

Research Question 1 *How are Twitter and Reddit used by members of pro-eating disorder communities?*

To understand the domain of the research and find possible characteristics that could be useful when building the final classifier, knowledge about eating disorders and especially online pro-eating disorder communities was obtained. During the literature review, several previous studies on online pro-ED communities were identified and were presented in Section 1.3 (Page 6) and 4.1 (Page 37). Giæver (2018) did a research on pro-ED users and identified several characteristics, where many of them showed to be influential when building a classifier. Many of the same findings were identified in **Dataset T**, which was presented in Section 5.6 (Page 68). Giæver found that pro-ED users used almost the same amount of characters and words as unrelated users, while pro-recovery users usually had longer tweets. The same patterns were found in **Dataset T**, although the numbers were scaled up because at the time the Giæver dataset was collected, Twitter was in the process of increasing the character limit from 140 to 280. When using the features for classification, the results from this study unveiled that the most influential features were TF-IDF scores of unigrams and bigrams, as found by both Nornes and Gran (2019), and Giæver (2018).

A sentiment analysis of tweets was also carried out in this Thesis. Posts labeled as pro-ED tended to have a more negative sentiment than unrelated posts, while pro-recovery tends to have a more positive sentiment. Branley and Covey (2017) studied the difference between pro-ED and pro-recovery content and reported that a lot of pro-recovery material offered support in the form of empathy, compassion, and understanding, which are concepts often associated with a positive sentiment. Borzekowski et al. (2010) and Wick and Harriger (2018) analyzed online pro-ED communities, and their findings suggest that a lot of the content are concepts related to negative sentiment. These studies, therefore, support findings from this Thesis' sentiment analysis. Although the analysis showed a negative and positive skewed distribution for pro-ED and pro-recovery, respectively, the difference from *unrelated* posts is not major. On the other hand, the even distribution of sentiment in pro-ED posts makes sense considering that one of the functions of pro-ED communities is to provide support and understanding for each other (Boniel-Nissim and Latzer, 2016).

The annotation of pro-ED content on Reddit unveiled differences among the usage of social media of the targeted communities, as briefly discussed in Section 9.1.1. The most noteworthy difference is the amount of information in each post. Reddit is a platform where users are able to express themselves in another way, compared to Twitter. The Reddit pro-ED community seems more focused on telling stories and comprehensively explain their struggles in a formal language. The posts often included several characteristics and well-known topics of pro-ED, and also tended to qualify for both being pro-ED and pro-recovery. On the other hand, posts on Twitter do not have the opportunity to include several topics and have to be more selective in the content of the post. The restrictions of the number of characters contribute to an informal language with a lot of abbreviations. This makes the nature of the texts at the two platforms very different.

Research Question 2 *What criteria should be used in the annotation of pro-eating disorder posts?*

One of the main contributions in this Thesis is the manually annotated dataset from Twitter. The quality of a dataset depends on the reliability and validity of the collection and annotation process, and a thorough process to define the annotation criteria was therefore needed. A reliable and valid set of criteria should ensure that the pro-ED criteria only captured explicit pro-ED posts, and pro-recovery criteria only captured posts where the topic was recovery from an eating disorder. The literature review conducted to answer research question 1 resulted in a set of annotation criteria, which are inspired by the work of Arseniev-Koehler et al. (2016) and Giæver (2018), and was presented in Section 5.2 (Page 48).

The discussion of the annotation process in Section 9.1.1 reports that the annotation criteria are satisfying for the task of annotating Twitter data, as the interpretation of the Cohen's Kappa score considers the agreement between the annotators to be *almost perfect* (Landis and Koch, 1977). The annotation process, and therefore the annotation criteria, can thus be considered reliable. Despite a satisfying evaluation of the annotation process, several minor challenges with the criteria were encountered, as described in Section 9.1.1. The most noteworthy challenge concerned Reddit, as the annotation criteria did not provide sufficient guidance when a post fulfilled a criterion from several classes. Common sense could be used to deal with this challenge in most cases. However, this invited subjective interpretation into the process, which was not desirable. The reliability of annotation of longer texts would therefore benefit from a reassessment of the criteria.

Research Question 3 *How can attention-based models be combined to improve the classification of pro-eating disorder posts?*

The ensemble learners from Experiment 1 and 2 were directly designed to answer Research Question 3. In this study, classification of pro-ED posts is performed on textual data from both Twitter and Reddit. The results from Experiment 1 showed that combining several models in an ensemble improved the performance score when compared to the individual models. The best ensemble learner in Experiment 1 increased several metrics compared to the best BERTweet model, including the most important ones, which were precision, recall, and F_1-score for both the pro-ED and pro-recovery classes, as well as the overall macro average F_1-score, weighted macro average F_1-score, and accuracy. This model consisted of four attention-based models and an SVM trained on TF-IDF scores of unigrams and bigrams from both tweets and biographies and used a voting classifier with a soft voting schema as the meta-classifier.

For Experiment 2, comparing the ensemble to the individual models showed that even larger performance gains were possible. The best ensemble learner outperformed the best ERNIE 2.0_{LARGE} by increasing the macro average F_1-score from 0.775 to 0.785 and the weighted macro average F_1-score from 0.807 to 0.816. Here, four attention-based models were included in the ensemble, as the results from testing on the Reddit data varied more among the individual models. The meta-classifier consisted of an FFNN instead of a voting classifier.

These two findings show that combining only a handful of fine-tuned attention-based models in a stacked ensemble architecture can improve the classification of pro-ED posts. As discussed in Section 9.1.2, several approaches to creating the ensemble could have been examined, and it is possible that other ways of building the ensembles would have further improved the performance scores.

Research Question 4 *How do attention-based models trained on data from one social media platform perform when tested on data from another platform on the task of classifying pro-eating disorder posts?*

In order to answer Research Question 4, data from both Twitter and Reddit were collected and annotated. Additionally, Experiment 2 and 3 were designed to test the performance of attention-based models for this purpose. The motivation behind investigating this research question lies in the possibility to create systems that can detect pro-ED content on social media platforms in general. As discussed earlier, the textual differences between posts on Twitter and Reddit are substantial, despite both being social media platforms. The textual differences are apparent in formality, punctuation, typos, use of abbreviations, post length, and use of slang and emojis. This contributes to making this task harder but at the same time more interesting, as it tests how well the models generalize to data from different sources and the robustness of the models.

When the models were fine-tuned on Twitter data (**Dataset T**) and tested on Reddit data (**Dataset R**) in Experiment 2, there was a decrease in performance score and an increase in variance between models. The same was found when the models were fine-tuned using the semi-automatically annotated dataset (**Dataset S**) and then tested on **Dataset R**. The decreasing performance scores were expected, but the magnitude of the changes provided insights into which models that might have the best potential to become a part of a more general pro-ED classification system. Whereas the difference between the best and the worst individual models in Experiment 1 with regards to the weighted macro average F_1-score was only 0.026 (from 0.937 to 0.911), the same differences increased to 0.125 (from 0.807 to 0.682) in Experiment 2, and 0.153 (from 0.651 to 0.498) in Experiment 3. ERNIE 2.0 was the model that impressed the most, being the best performing model in both experiments. On the other hand, BERT struggled in these two experiments and was outperformed by the baseline SVM on both occasions. This does not automatically establish ERNIE 2.0 as a good model and BERT as a bad model when creating a general pro-ED classification system, as approaches and configurations that were not investigated for this study could provide different results. However, the findings could provide a good indication of where to start when conducting future research on the same topic.

BERTweet was the only model in this study that is pre-trained on social media data, namely Twitter. Although being the best performing model in Experiment 1, it performed third best in Experiment 2, and worst in Experiment 3. This could indicate that platform-specific pre-training is not desirable when the textual format of the platform it is pre-trained for becomes too distant from the textual format it will be applied to, as may be the case for Twitter and Reddit.

Goal *Identify pro-eating disorder posts from various social media platforms by using attention-based models.*

Based on the results obtained through the experiments of this Thesis and the evaluation of the research questions, a classification system for pro-ED posts on both Twitter and Reddit is implemented successfully using attention-based models. The experiments show that combining the attention-based models in an ensemble architecture performs better than using the models individually for prediction. The performance is better for the Twitter dataset than the Reddit dataset, which is natural as Reddit data was only included for testing and was not a part of the fine-tuning process. Further research should be applied to improve the systems if they are to be used in real-life social media applications. As this study is the first to conduct classification experiments on the collected datasets, the findings should be used as a baseline comparison for future studies on the subject.

10 Conclusion and Future Work

The pro-ED community is small, but the content they post can severely impact the viewers in a negative way, as seen in Section 4.1 (Page 37). Several social media platforms have taken measures to restrict the publicity or provide advisory content when certain tags associated with pro-ED content are used as search words. To further improve this identification of pro-ED content on social media, this Thesis has investigated the usage of the NLP technique Attention and models built on this technique. Several attention-based models were combined in an ensemble architecture and achieved state-of-the-art results for the task at hand. Thus, this study has successfully managed to identify pro-ED users on Twitter and Reddit, which was the main goal. Despite satisfying results, there are several aspects to improve upon in further research before the systems can be applied at real-life social media platforms.

This chapter will provide a conclusion to the work conducted in this Thesis, the contributions to the field of classifying online pro-ED content, a statement on ethical considerations, and suggestions and ideas to which improvements can be introduced in future work. In summary, new datasets for the purpose of developing pro-ED classification systems have been made. Two of the datasets consist of manually annotated posts from Twitter and Reddit, while a third dataset is a semi-automatically annotated dataset with tweets. Additionally, classification systems for pro-ED posts on social media have been implemented. The systems are ensembles of various attention-based models and have shown to improve the performance compared to the performance of individual models. Suggestions for improvements of the work in this Thesis, along with ideas to new perspectives, are introduced in the last section of this chapter.

10.1 Contributions

This Thesis contributes to the field of classifying and analyzing pro-ED content on social media by creating several pro-ED datasets and exploiting the potential of attention-based models for the classification of textual data. A literature review focusing on previous studies on pro-ED content resulted in a set of criteria for the annotation of both pro-ED and pro-recovery posts on social media. The set consists of criteria for both manual annotation and semi-automatic annotation. By using the criteria for manual annotation, two datasets were annotated and created; a pro-ED dataset with 16 389 tweets (**Dataset T**) and a pro-ED dataset with 376 posts from Reddit (**Dataset R**). **Dataset T** is one of the main contributions of this Master's Thesis and is the first dataset with manually annotated pro-ED social media posts. Although some challenges were encountered during the annotation process, the evaluation of the inter-annotator agreement proved the dataset gathered from Twitter to be reliable and that the dataset can be used as a foundation for further research in the field. The semi-automatically annotated dataset is another contribution and is a large dataset of 136 846 tweets (**Dataset S**). As discussed in Chapter 9, the annotation algorithm was simple, and the experiments conducted on this dataset had a notable drop in performance compared to the experiments conducted on the other datasets.

The experiments resulted in several contributions, where the first is an overview of how several attention-based models perform on the task of classifying pro-ED posts from social media. The models that were tested in this Thesis are: BERT, DistilBERT, RoBERTa, BERTweet, ERNIE 2.0, and SVM. Each model was fine-tuned on three different datasets; **Dataset T**, **Dataset S**, and a version of **Dataset T** with a more balanced label distribution (**Dataset T***). These fine-tuned models were tested on both **Dataset T** and **Dataset R**, to evaluate their performances on data from different social medias. Further, two ensemble learners were built by combining the attention-based models and achieved high scores for several performance measures on the task of classifying pro-ED content. One of the ensemble learners had a voter as meta-classifier, while the other had an FFNN as meta-classifier. The ensemble architectures performed better than the individual models, a trait that is substantiated by other studies. The ensemble learners are also part of the main contributions of this Master's Thesis. Additionally, the experiments confirmed that a system that is only fine-tuned on one type of textual data not necessarily generalizes, as the models fine-tuned on the short and informal tweets struggled more when classifying the long and formal Reddit posts in **Dataset R**. There is great potential for improving the performance for this task, and possible ideas will be presented in the next section.

10.2 Ethical Considerations

Most social media content is publicly and freely available for everyone to see. In the field of natural language processing, this has led to a lot of research being conducted on the area in the last decade. However, just that the data is available does not mean that one should act carelessly when dealing with topics like mental illnesses. As this Master's Thesis considers data collection and annotation of content concerning eating disorders, which is written by private individuals who do not know that their data is being used for research purposes, some ethical considerations should be taken into account.

The datasets that were established during this study were based on a set of annotation criteria that were made to categorize social media posts as either pro-ED, pro-recovery or unrelated. Although many of the posts include a self-identification as pro-ED, some fulfill the criteria in other ways. The writers of those posts may not themselves identify with a pro-ED attitude, which raises the question of whether it is ethically correct to utilize their data for this purpose. As a result of this, it was important to carefully address the privacy of the collected users. The posts that are labeled as pro-ED in this study are done so based on the annotation criteria only, which do not make any claims to whether a user *is* pro-ED or not.

The main focus of this Thesis was to create a system that could help identity pro-ED posts in social media. Chapter 1 (Page 1) introduces the topic of moderating or restricting the spread of pro-ED content on social media platforms. This debate has two sides, where some claim that the platforms should censor content that is considered harmful, while others think this violates the freedom of speech. This study does not take a stand on the question of whether or not, or to what extent, pro-ED content should be censored. Hopefully, this study can contribute to an increased focus on the field of research, which will potentially lead to reasonable solutions for both the pro-ED community and the general public.

10.3 Future Work

During the work conducted in this Master's Thesis, several ideas for improvements were found. This section will focus on future work that could be explored to further improve the findings of the results in this study, both when considering the creation of pro-ED datasets and building a pro-ED text classification system utilizing attention-based models.

10.3.1 Dataset Extension and Annotation Procedure

The collection and annotation procedures conducted in this Master's Thesis resulted in three datasets of variable sizes collected from two different social media platforms. Previous discussions in this study suggest dataset sizes, the imbalance between classes, and annotation criteria as the three areas where there is room for improvement. In the case of training deep learning neural networks, using as much data as possible is always desirable. Therefore, it may be beneficial to collect more data and conduct further annotation in the future. In this case, collecting more pro-ED and pro-recovery data during the web scraping process should be in focus; these posts are more challenging to find due to a smaller community in contrast to the unrelated posts. In addition, the label distributions of the datasets in this study tries to capture a realistic representation of the share of pro-ED posts versus unrelated posts on the platforms while at the same time giving the models enough data to be fine-tuned on. However, it is possible to investigate other distributions and examine the impact this might have when performing classification tasks on the datasets.

As mentioned in Section 5.5.1 (Page 57), the algorithm for semi-automatic annotation of Twitter data in this study was effective, yet simple. Future studies could explore more advanced algorithms for creating a dataset that is both of substantial size and quality.

Another possibility for future studies is to collect and annotate a training dataset for Reddit. This study only includes a small dataset of 386 posts from Reddit, which is not enough to be used to fine-tune attention-based models efficiently. This would give additional insights into how well attention-based models generalize for social media data from several platforms. Additionally, examining other platforms like Tumblr, Instagram, and Facebook could be of interest.

Further investigation of both the annotation criteria and process for pro-ED posts should also be addressed. A problem encountered during this study was that the annotation criteria initially developed for Twitter data did not necessarily fit Reddit data. Creating an annotation process that can generalize to annotate data across several platforms could be beneficial, even though this requires investigating similarities and dissimilarities among platforms. However, establishing such a process could introduce challenges as it may be too universal and might not capture platform-specific edge cases.

10.3.2 Classifier Systems

The experiments in this study both explore a variety of model architectures and different parameters for these. Despite this, there are numerous other configurations and approaches left to investigate. Further investigations of features could be explored, like Nornes and Gran (2019) did by using personality as a feature. Although it might be hard to determine the personality based on a single tweet, the meta-data for a user could be used for this. Other possible features like social media network structures could also be interesting to look at.

As mentioned in Section 9.1.2, most of the experiments investigated the performance on the multiclass task. However, another approach could focus more on the binary classification task, ignoring pro-recovery as a separate class. Pro-recovery was added to the task by Giæver (2018) to explore how models differentiated between pro-ED and pro-recovery, as posts from these two groups apply similar wordings. Future studies could focus more on the specific task of detecting pro-ED posts only.

Future studies may look at techniques to decrease the variance among models by looking more specifically at the fine-tuning process of attention-based models. For instance, running the experiments several times using different random initializations for the weights and dataset splits was discarded for this study. However, this could be investigated as it has improved the performance on other text classification tasks in related research. In addition, using a dynamic learning rate schedule during fine-tuning could prevent models from not converging. Cross-validation is another technique that would help give a more nuanced overview of which models are actually performing better than others. Additionally, further pre-training of attention-based models for specific social media platforms may also be examined.

Another interesting approach would be to experiment with the composition of the ensemble learner architectures. Several proposals for future implementations were made in Section 9.1.2, and involve different weighting of models, and including models pre-trained on specific domains or platforms.

Bibliography

Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jindi, Tristan Naumann, and Matthew McDermott. Publicly available clinical BERT embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78, Minneapolis, Minnesota, USA, June 2019. Association for Computational Linguistics.

Jon Arcelus, Alex Mitchell, Jackie Wales, and Søren Nielsen. Mortality rates in patients with anorexia nervosa and other eating disorders: A meta-analysis of 36 studies. *Archives of General Psychiatry*, 68(7):724–731, 2011.

Alina Arseniev-Koehler, H. Lee, T. McCormick, and M. Moreno. #Proana: Pro-Eating Disorder Socialization on Twitter. *The Journal of adolescent health: official publication of the Society for Adolescent Medicine*, 58(6):659–64, 2016.

Anna M. Bardone-Cone and Kamila M. Cass. What does viewing a pro-anorexia website do? An experimental examination of website exposure and moderating effects. *International Journal of Eating Disorders*, 40(6):537–548, 2007. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/eat.20396`.

Natalie Boero and Cheri Pascoe. Pro-anorexia communities and online interaction: Bringing the pro-ana body online. *Body & Society*, 18:27–57, 2012.

Meyran Boniel-Nissim and Yael Latzer. *The Characteristics of Pro-Ana Community*, pages 155–167. Springer International Publishing, 2016. URL `https://doi.org/10.1007/978-3-319-32742-6_11`.

Dina Borzekowski, Summer Schenk, Jenny Wilson, and Rebecka Peebles. e-Ana and e-Mia: A content analysis of pro-eating disorder web sites. *American journal of public health*, 100(8):1526–1534, 2010.

Bernhard Boser, Isabelle Guyon, and Vladimir Vapnik. A Training Algorithm for Optimal Margin Classifiers. *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, page 144–152, 1992.

Dawn Branley and Judith Covey. Pro-ana versus Pro-recovery: A Content Analytic Comparison of Social Media Users' Communication about Eating Disorders on Twitter and Tumblr. *Frontiers in Psychology*, 8:1356, 2017. URL `https://www.frontiersin.org/article/10.3389/fpsyg.2017.01356`.

Silvia Casola and Alberto Lavelli. FBK@SMM4H2020: RoBERTa for detecting medications on Twitter. In *Proceedings of the Fifth Social Media Mining for Health*

Applications Workshop & Shared Task, pages 101–103, Barcelona, Spain (Online), December 2020. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/2020.smm4h-1.15`.

Patricia Cavazos-Rehg, Melissa Krauss, Shaina Costello, Nina Kaiser, Elizabeth Cahn, Ellen Fitzsimmons-Craft, and Denise Wilfley. "i just want to be skinny.": A content analysis of tweets expressing eating disorder symptoms. *PLoS One*, 14(1):1–11, 2019. URL `https://doi.org/10.1371/journal.pone.0207506`.

Stevie Chancellor, Tanushree Mitra, and Munmun Choudhury. Recovery amid pro-anorexia: Analysis of recovery in social media. *Proceedings of the SIGCHI conference on human factors in computing systems*, 2016:2111–2123, 2016a.

Stevie Chancellor, Jessica Annette Pater, Trustin Clear, Eric Gilbert, and Munmun De Choudhury. #thyghgapp: Instagram content moderation and lexical variation in pro-eating disorder communities. *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*, page 1201–1213, 2016b. doi: 10.1145/2818048.2819963.

Shuai Chen, Yuanhang Huang, Xiaowei Huang, Haoming Qin, Jun Yan, and Buzhou Tang. HITSZ-ICRC: A report for SMM4H shared task 2019-automatic classification and extraction of adverse effect mentions in tweets. In *Proceedings of the Fourth Social Media Mining for Health Applications (#SMM4H) Workshop & Shared Task*, pages 47–51, Florence, Italy, 2019. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/W19-3206`.

Munmun Choudhury. Anorexia on Tumblr: A Characterization Study. *Proceedings of the 5th International Conference on Digital Health 2015*, page 43–50, 2015. doi: 10.1145/2750511.2750515. URL `https://doi.org/10.1145/2750511.2750515`.

Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46, 1960.

Huong Dang, Kahyun Lee, Sam Henry, and Özlem Uzuner. Ensemble BERT for classifying medication-mentioning tweets. In *Proceedings of the Fifth Social Media Mining for Health Applications Workshop & Shared Task*, pages 37–41, Barcelona, Spain (Online), December 2020. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/2020.smm4h-1.5`.

Sam Davidson, Qiusi Sun, and Magdalena Wojcieszak. Developing a new classifier for automated identification of incivility in social media. In *Proceedings of the Fourth Workshop on Online Abuse and Harms*, pages 95–101. Association for Computational Linguistics, 2020. URL `https://www.aclweb.org/anthology/2020.alw-1.12`.

Monique Delforterie, Junilla Larsen, Anna Bardone-Cone, and Ron Scholte. Effects of Viewing a Pro-Ana Website: An Experimental Study on Body Satisfaction, Affect, and Appearance Self-Efficacy. *Eating disorders*, 22, 2014.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, 2018. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/N19-1423`.

Jesse Dodge, Gabriel Ilharco, Roy Schwartz, Ali Farhadi, Hannaneh Hajishirzi, and Noah Smith. Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping. *CoRR*, 2020.

Tilia Ellendorff, Lenz Furrer, Nicola Colic, Noëmi Aepli, and Fabio Rinaldi. Approaching SMM4H with merged models and multi-task learning. In *Proceedings of the Fourth Social Media Mining for Health Applications (#SMM4H) Workshop & Shared Task*, pages 58–61, Florence, Italy, 2019. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/W19-3208`.

Anna Feldman, Giovanni Da San Martino, Alberto Barrón-Cedeño, Chris Brew, Chris Leberknight, and Preslav Nakov, editors. *Proceedings of the Second Workshop on Natural Language Processing for Internet Freedom: Censorship, Disinformation, and Propaganda*, Hong Kong, China, 2019. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/D19-5000`.

Yousra Fettach and Lamia Benhiba. Pro-Eating Disorders and Pro-Recovery Communities on Reddit: Text and Network Comparative Analyses. In *Proceedings of the 21st International Conference on Information Integration and Web-Based Applications & Services*, pages 277–286, 2019.

Ingrid Nelson Giæver. Classification of Pro-Eating Disorder Users on Twitter. Msc, Dept. of Computer Science, Norwegian University of Science and Technology, Trondheim, Norway, June 2018.

Graciela Gonzalez-Hernandez, Davy Weissenbacher, Abeed Sarker, and Michael Paul, editors. *Proceedings of the 2018 EMNLP Workshop SMM4H: The 3rd Social Media Mining for Health Applications Workshop & Shared Task*, Brussels, Belgium, 2018. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/W18-5900`.

Santiago González-Carvajal and Eduardo Garrido-Merchán. Comparing BERT against traditional machine learning text classification. *ArXiv*, abs/2005.13012, 2020.

Yuting Guo, Xiangjue Dong, Mohammed Ali Al-Garadi, Abeed Sarker, Cecile Paris, and Diego Mollá Aliod. Benchmarking of transformer-based pre-trained models on social media text classification datasets. In *Proceedings of the The 18th Annual Workshop of the Australasian Language Technology Association*, pages 86–91, Virtual Workshop, 2020. Australasian Language Technology Association.

Kelley Harper, Steffanie Sperry, and Joel Thompson. Viewership of pro-eating disorder websites: Association with body image and eating disturbances. *The International journal of eating disorders*, 41:92–95, 2008.

Zellig Harris. Distributional structure. *Word*, 10:146–162, 1954.

Charlotte Hilton. “It’s the symptom of the problem, not the problem itself”: A qualitative exploration of the role of pro-anorexia websites in users’ disordered eating. *Issues in Mental Health Nursing*, 39(10):865–875, 2018. doi: 10.1080/01612840.2018.1493625.

C. J. Hutto, Kenny Joseph, Chris K. W., Fritz Lekschas, Pascal van Kooten, Philip, 0wlyW00d, and Max Frai. Vader-sentiment-analysis, 2014. URL `https://github.com/cjhutto/vaderSentiment`.

Instagram. Community guidelines. `https://help.instagram.com/477434105621119?fbclid=IwAR19vGH4xzRoBfmQqxNwLBsz8Qg0tQBguBWK1tEL76lijiwtIzdBlr2svxA`, 2012. Accessed: 2020-11-23.

Vebjørn Isaksen. Detecting hateful and offensive language with transfer-learned models. Msc, Dept. of Computer Science, Norwegian University of Science and Technology, Trondheim, Norway, 2019.

Dag Ingvar Jacobsen. *Hvordan gjennomføre undersøkelser?*, volume 3. Cappelen Damm akademisk, 1 edition, 2015.

Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 12 2014.

Ari Klein, Ilseyar Alimova, Ivan Flores, Arjun Magge, Zulfat Miftahutdinov, Anne-Lyse Minard, Karen O’Connor, Abeed Sarker, Elena Tutubalina, Davy Weissenbacher, and Graciela Gonzalez-Hernandez. Overview of the fifth social media mining for health applications (#SMM4H) shared tasks at COLING 2020. In *Proceedings of the Fifth Social Media Mining for Health Applications Workshop & Shared Task*, pages 27–36, Barcelona, Spain (Online), 2020. Association for Computational Linguistics.

Ritesh Kumar, Atul Kr. Ojha, Shervin Malmasi, and Marcos Zampieri. Evaluating aggression identification in social media. In *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, pages 1–5, Marseille, France, May 2020. European Language Resources Association (ELRA). URL `https://www.aclweb.org/anthology/2020.trac-1.1`.

Per Egil Kummervold, Catherine Chronaki, Berthold Lausen, Hans-Ulrich Prokosch, Janne Rasmussen, Silvina Santana, Andrzej Staniszewski, and Silje Camilla Wangberg. ehealth trends in europe 2005-2007: A population-based survey. *Journal of medical Internet research*, 2008. URL `http://www.ncbi.nlm.nih.gov/pubmed/19017584`.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. *International Conference on Learning Representations*, 2019. URL `https://openreview.net/forum?id=H1eA7AEtvS`.

J. Richard Landis and Gary G. Koch. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174, 1977. ISSN 0006341X, 15410420. URL `http://www.jstor.org/stable/2529310`.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *CoRR*, 2019.

Hector J. Levesque, Ernest Davis, and Leora Morgenstern. The winograd schema challenge. In *Proceedings of the Thirteenth International Conference on Principles of Knowledge Representation and Reasoning*, KR'12, page 552–561. AAAI Press, 2012.

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, 2019.

David Losada, Fabio Crestani, and Javier Parapar. Overview of erisk: Early risk prediction on the internet. In Patrice Bellot, Chiraz Trabelsi, Josiane Mothe, Fionn Murtagh, Jian Yun Nie, Laure Soulier, Eric SanJuan, Linda Cappellato, and Nicola Ferro, editors, *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, pages 343–361. Springer International Publishing, 2018.

David Losada, Fabio Crestani, and Javier Parapar. Overview of erisk 2019 early risk prediction on the internet. In Fabio Crestani, Martin Braschler, Jacques Savoy, Andreas Rauber, Henning Müller, David E. Losada, Gundula Heinatz Bürki, Linda Cappellato, and Nicola Ferro, editors, *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, pages 340–357. Springer International Publishing, 2019.

Norman Mapes, Anna White, Radhika Medury, and Sumeet Dua. Divisive language and propaganda detection using multi-head attention transformers with deep learning BERT-based language models for binary classification. In *Proceedings of the Second Workshop on Natural Language Processing for Internet Freedom: Censorship, Disinformation, and Propaganda*, pages 103–106, Hong Kong, China, 2019. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/D19-5014`.

Tom M. Mitchell. *Machine Learning*, chapter 1, page 2. McGraw-Hill, Singapore, 1 edition, 1997.

Elham Mohammadi, Hessam Amini, and Leila Kosseim. Quick and (maybe not so) easy detection of anorexia in social media posts. In *Conference and Labs of the Evaluation Forum*, 2019.

Marzieh Mozafari, Reza Farahbakhsh, and Noel Crespi. A bert-based transfer learning approach for hate speech detection in online social media. In *Complex Networks 2019: 8th International Conference on Complex Networks and their Applications*, Lisbon, Portugal, 2019. Springer.

Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. Bertweet: A pre-trained language model for english tweets. *CoRR*, 2020.

Andrea Hollung Nornes and Martine Alvilde Gran. Automatic Classification of Pro-Eating Disorder Twitter Accounts with Personality as a Feature. MSc, Dept. of Computer Science, Norwegian University of Science and Technology, Trondheim, Norway, June 2019.

German Ignacio Parisi, Ronald Kemker, Jose L. Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks. *Neural Networks*, 113: 54–71, 2019. ISSN 0893-6080. doi: https://doi.org/10.1016/j.neunet.2019.01.012. URL `https://www.sciencedirect.com/science/article/pii/S0893608019300231`.

Pinterest. Community guidelines. `https://policy.pinterest.com/nb/community-guidelines`, 2012. Accessed: 2020-11-23.

Janet Polivy and C. Peter Herman. Causes of eating disorders. *Annual Review of Psychology*, pages 187–213, 2002. URL `https://doi.org/10.1146/annurev.psych.53.100901.135103`.

Diana Ramírez-Cifuentes, Marc Mayans, and Ana Freire. Early risk detection of anorexia on social media. In *5th International Conference, INSCI 2018, St. Petersburg, Russia, October 24–26, 2018, Proceedings*, pages 3–14, St. Petersburg, Russia, 2018. ISBN 978-3-030-01436-0.

Julian Risch and Ralf Krestel. Bagging BERT models for robust aggression identification. In *Proceedings of the Second Workshop on Trolling, Aggression and Cyberbullying*, pages 55–61, Marseille, France, May 2020. European Language Resources Association (ELRA). URL `https://www.aclweb.org/anthology/2020.trac-1.9`.

Yoel Roth and Ashita Achuthan. Building rules in public: Our approach to synthetic & manipulated media. `https://help.instagram.com/477434105621119?fbclid=IwAR19vGH4xzRoBfmQqxNwLBsz8QgOtQBguBWK1tEL76lijiwtIzdBlr2svxA`, 2020. Accessed: 2020-05-26.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108, 2019.

Mike Schuster and Kaisuke Nakajima. Japanese and korean voice search. In *International Conference on Acoustics, Speech and Signal Processing*, pages 5149–5152, 2012.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. *CoRR*, 2015.

Magnus Själander, Magnus Jahre, Gunnar Tufte, and Nico Reissmann. EPIC: An energy-efficient, high-performance GPGPU computing research infrastructure, 2019.

F.R.E. Smink, D. van Hoeken, and H.W. Hoek. Epidemiology of eating disorders:

Incidence, prevalence and mortality rates. *Current Psychiatry Reports 14*, page 406–414, 2012.

Ian Stewart, Stevie Chancellor, Munmun Choudhury, and Jacob Eisenstein. #anorexia, #anarexia, #anarexyia: Characterizing online community practices with orthographic variation, 2017.

Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Hao Tian, Hua Wu, and Haifeng Wang. Ernie 2.0: A continual pre-training framework for language understanding. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8968–8975, 2020. URL `https://ojs.aaai.org/index.php/AAAI/article/view/6428`.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks, 2014.

Sloper Talbot. The effects of viewing pro-eating disorder websites: a systematic review. *The West Indian medical journal*, 59(6):686–97, 2010.

Tumblr. A new policy against self-harm blogs, 2012. URL `https://staff.tumblr.com/post/18132624829/self-harm-blogs`.

Ernest Tupes and Raymond Christal. *Recurrent Personality Factors Based on Trait Ratings.* Number 2 in ASD technical report. Personnel Laboratory, Aeronautical Systems Division, Air Force Systems Command, United States Air Force, 1961.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017. URL `https://arxiv.org/pdf/1706.03762.pdf`.

A. Viera and J. Garrett. Understanding interobserver agreement: the kappa statistic. *Family medicine*, 37 5:360–3, 2005.

Atro Voutilainen. An experiment on the upper bound of interjudge agreement: the case of tagging. In *Ninth Conference of the European Chapter of the Association for Computational Linguistics*, pages 204–208. Association for Computational Linguistics, June 1999.

Tao Wang, Markus Brede, Antonella Ianni, and Emmanouil Mentzakis. Detecting and characterizing eating-disorder communities on social media. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, WSDM '17, page 91–100, New York, NY, USA, 2017. Association for Computing Machinery. URL `https://doi.org/10.1145/3018661.3018706`.

Tao Wang, Markus Brede, Antonella Ianni, and Emmanouil Mentzakis. Social interactions in online eating disorder communities: A network perspective. *PLOS ONE*, 13(7):1–17, 07 2018.

Davy Weissenbacher and Graciela Gonzalez-Hernandez, editors. *Proceedings of the Fourth Social Media Mining for Health Applications (#SMM4H) Workshop & Shared*

Tas, Florence, Italy, 2019. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W19-3200.

Krista Whitehead. Hunger hurts but starving works: A case study of gendered practices in the online pro-eating-disorder community. *Canadian Journal of Sociology*, 35:595–626, 2010.

Madeline Wick and Jennifer Harriger. A content analysis of thinspiration images and text posts on Tumblr. *Body Image*, 24:13–16, 2018.

Jenny Wilson, Rebecka Peebles, Kristina Hardy, and Iris Litt. Surfing for thinness: A pilot study of pro–eating disorder web site usage in adolescents with eating disorders. *Pediatrics*, 118(6):1635–1643, 2006. URL https://pediatrics.aappublications.org/content/118/6/e1635.

Chuhan Wu, Fangzhao Wu, Junxin Liu, Sixing Wu, Yongfeng Huang, and Xing Xie. Detecting tweets mentioning drug name and adverse drug reaction with hierarchical tweet representation and multi-head self-attention. In *Proceedings of the 2018 EMNLP Workshop SMM4H: The 3rd Social Media Mining for Health Applications Workshop & Shared Task*, pages 34–37, Brussels, Belgium, 2018. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/W18-5909.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. XLNet: Generalized autoregressive pretraining for language understanding, 2019.

Daphna Yeshua-Katz and Nicole Martins. Communicating stigma: The pro-ana paradox. *Health communication*, 28(5):499–508, 2012.

Elad Yom-Tov, Luis Fernandez-Luque, Ingmar Weber, and Steven Crain. Pro-anorexia and pro-recovery photo sharing: A tale of two warring tribes. *Journal of medical Internet research*, 14:151, 2012.

Yuehua Zhao and Jin Zhang. Consumer health information seeking in social media: A literature review. *Health Information & Libraries Journal*, 2017.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *The IEEE International Conference on Computer Vision (ICCV)*, 2015.

A Pre-Processing

A.1 Social Media Abbreviations

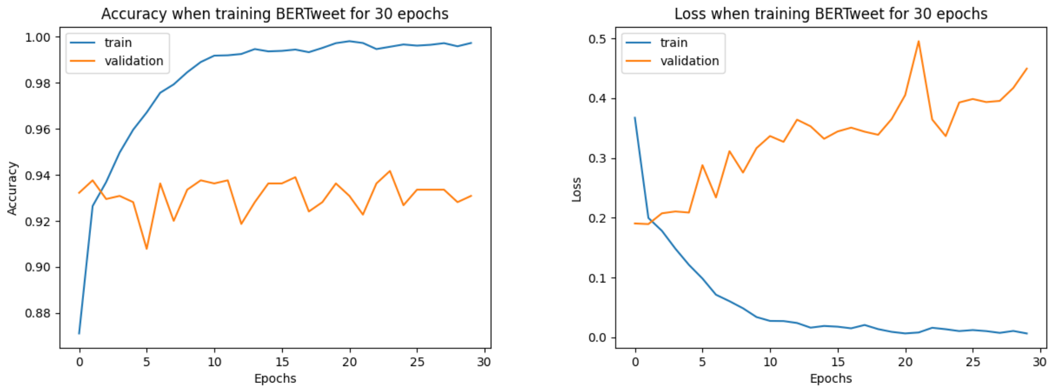
Abbreviations	Word/Phrase
ppl	people
lmk	let me know
idk	i don’t know
wtf	what the fuck
pls	please
abt	about
fr	for real
ty	thank you
rn	right now
nvm	never mind

Table A.1.1: Abbreviations that were replaced during the pre-processing of social media data.

B Experimental results

This appendix will include the tables showing the results from Experiment 1 that were not included in Section 8.3. Runs where the attention-based models did not converge are not included, which leads to some tables having different lengths and parameters. Experiment 2 and 3 are not included, as all results from these experiments are already included in Section 8.3.

B.1 Experiment 1 - Results



(a) Accuracy.

(b) Loss.

Figure B.1.1: Accuracy and loss during training of BERTweet for 30 epochs on Dataset T.

SVM features	**Unrelated**			**Pro-ED**			**Pro-recovery**			**Macro Avg F1**	**Weight. Avg F1**	**Acc**
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**			
Tweet unigram	0.946	0.949	0.947	**0.792**	0.839	0.815	0.730	0.607	0.663	0.809	0.909	0.910
Tweet bigram	0.902	0.948	0.924	0.757	0.627	0.686	0.600	0.477	0.531	0.714	0.862	0.868
Bio, unigram	0.911	0.820	0.863	0.513	0.647	0.572	0.477	**0.710**	0.549	0.661	0.798	0.786
Bio, bigram	0.911	0.829	0.868	0.537	0.647	0.587	0.442	**0.710**	0.545	0.666	0.804	0.793
Username + name, character n-grams	0.919	0.835	0.875	0.559	0.683	0.615	0.447	**0.710**	0.549	0.680	0.814	0.804
Tweet, uni+bigram	0.947	0.954	**0.951**	0.789	**0.843**	**0.816**	0.790	0.598	0.681	0.816	**0.913**	**0.914**
Tweet + bio	0.948	0.947	0.948	0.784	0.831	0.807	0.766	0.673	**0.716**	**0.824**	0.911	0.912
Tweet + bio, username + name	0.947	0.942	0.944	0.758	0.803	0.780	0.737	0.682	0.709	0.811	0.904	0.904
Tweet + bio, tweet length, sentiment score	0.938	**0.959**	0.949	0.788	0.819	0.803	**0.853**	0.542	0.663	0.805	0.908	0.911
Tweet + bio, username + name, tweet length	0.943	0.946	0.945	0.749	0.815	0.781	0.815	0.617	0.702	0.809	0.904	0.905
Tweet + bio, username + name, sentiment score	**0.950**	0.941	0.945	0.754	0.811	0.781	0.740	0.692	0.715	0.814	0.905	0.905
All feature groups	0.932	0.955	0.943	0.782	0.735	0.758	0.769	0.654	0.707	0.803	0.900	0.902

Table B.1.1: Results from Experiment 1 for the Support Vector Machine baseline model trained on Dataset T.

Models	**Unrelated**			**ProED**			**Pro-recovery**			**Macro Avg F1**	**Weight. Avg F1**	**Acc**
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**			
SVM, tweet, uni + bigrams	0.958	**0.931**	**0.945**	**0.736**	0.863	**0.795**	**0.730**	0.682	**0.705**	**0.815**	**0.906**	**0.905**
SVM, tweet + bio, uni + bigrams	**0.961**	0.917	0.938	0.713	**0.867**	0.783	0.679	**0.710**	0.694	0.805	0.899	0.896

Table B.1.2: Results from Experiment 1 for the SVM models trained on Dataset T*.

BERT Models	**Unrelated**			**Pro-ED**			**Pro-recovery**			**Macro Avg F1**	**Weight. Avg F1**	**Acc**
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**			
BASE, B = 16, LR = 1e-5	0.970	0.947	0.948	0.785	0.863	0.822	0.705	0.738	0.721	0.834	0.922	0.921
BASE, B = 32, LR = 5e-5	0.968	0.953	**0.961**	0.814	0.863	0.838	0.714	0.748	0.731	0.843	0.927	**0.926**
BASE, B = 16, LR = 2e-5	0.945	**0.972**	0.958	**0.888**	0.735	0.804	0.708	0.748	0.727	0.830	0.920	0.921
BASE, B = 16, LR = 2e-5	0.961	0.952	0.957	0.823	0.819	0.821	0.692	0.776	0.731	0.836	0.922	0.921
LARGE, B = 16, LR = 1e-5	**0.976**	0.941	0.958	0.787	0.904	**0.841**	0.726	**0.787**	0.756	0.852	0.927	0.925
LARGE, B = 32, LR = 1e-5	**0.976**	0.939	0.957	0.758	**0.932**	0.836	**0.828**	0.759	**0.792**	**0.862**	**0.928**	**0.926**

Table B.1.3: Results from Experiment 1 for the BERT models fine-tuned on Dataset T.

BERT Models	**Unrelated**			**Pro-ED**			**Pro-recovery**			**Macro Avg F1**	**Weight. Avg F1**	**Acc**
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**			
BASE, B = 16, LR = 1e-5	**0.979**	0.923	0.950	0.750	0.867	0.804	0.603	**0.794**	0.685	0.813	0.911	0.906
BASE, B = 32, LR = 1e-5	**0.979**	0.913	0.945	0.703	**0.932**	0.801	0.717	0.757	0.736	0.827	0.909	0.905
BASE, B = 32, LR = 5e-5	0.970	0.926	0.947	0.715	0.888	0.792	0.724	0.710	0.717	0.819	0.909	0.906
BASE, B = 32, LR = 2e-5	0.966	**0.952**	**0.959**	**0.800**	0.851	**0.825**	**0.727**	0.748	**0.737**	**0.840**	**0.924**	**0.923**

Table B.1.4: Results from Experiment 1 for the BERT models fine-tuned on Dataset T*.

ERNIE Models	Unrelated			ProED			Pro-recovery			Macro Avg F1	Weight. Avg F1	Acc
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**			
BASE, B = 16, LR = 1e-5	**0.972**	0.945	0.959	0.777	0.867	0.820	0.743	0.785	**0.764**	**0.847**	0.925	0.923
BASE, B = 32, LR = 1e-5	0.953	0.962	0.957	0.835	0.811	0.823	**0.775**	0.738	0.756	0.845	0.924	0.924
BASE, B = 16, LR = 2e-5	0.947	**0.965**	0.956	**0.892**	0.731	0.804	0.672	**0.804**	0.732	0.830	0.918	0.919
BASE, B = 32, LR = 2e-5	0.966	0.956	**0.961**	0.840	0.819	0.829	0.683	**0.804**	0.738	0.843	**0.927**	0.926
LARGE, B = 32, LR = 1e-5	0.970	0.941	0.956	0.778	**0.888**	0.829	0.739	0.759	0.845	0.923	0.921	
LARGE, B = 32, LR = 2e-5	0.960	0.962	**0.961**	0.858	0.803	**0.830**	0.708	0.794	0.749	0.846	**0.927**	**0.927**

Table B.1.5: Results from Experiment 1 for the ERNIE 2.0 models fine-tuned on Dataset T.

ERNIE Models	Unrelated			Pro-ED			Pro-recovery			Macro Avg F1	Weight. Avg F1	Acc
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**			
BASE, B = 16, LR = 1e-5	0.966	0.942	0.954	0.779	0.851	0.814	0.713	0.766	0.739	0.835	0.919	0.917
BASE, B = 32, LR = 1e-5	0.976	**0.937**	**0.956**	**0.788**	0.896	**0.838**	0.685	**0.794**	0.736	**0.843**	**0.924**	**0.921**
BASE, B = 16, LR = 2e-5	0.976	0.933	0.954	0.743	**0.928**	0.825	**0.772**	0.729	**0.750**	**0.843**	0.921	0.919
BASE, B = 32, LR = 2e-5	**0.977**	0.924	0.950	0.736	**0.928**	0.821	0.712	0.738	0.725	0.832	0.916	0.913

Table B.1.6: Results from Experiment 1 for the ERNIE 2.0 models fine-tuned on Dataset T*.

BERTweet	**Unrelated**			**ProED**			**Prorecovery**			**Macro**	**Weight.**	**Acc**
Model	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**Avg F1**	**Avg F1**	
B = 16, LR = 1e-5	0.968	0.959	**0.963**	0.837	**0.888**	**0.862**	0.808	**0.875**	**0.796**	**0.874**	**0.937**	**0.937**
B = 32, LR = 1e-5	**0.971**	0.950	0.961	0.806	0.867	0.836	0.733	0.794	0.762	0.853	0.929	0.927
B = 16, LR = 2e-5	0.955	**0.970**	0.962	0.841	0.851	0.846	**0.869**	0.682	0.764	0.858	0.932	0.933
B = 32, LR = 2e-5	0.956	0.964	0.960	0.848	0.759	0.801	0.697	0.794	0.742	0.834	0.922	0.922
B = 32, LR = 2e-5, epochs = 10	0.952	0.967	0.960	**0.858**	0.775	0.814	0.757	0.785	0.771	0.848	0.925	0.926
B = 32, LR = 2e-5, epochs = 30	**0.971**	0.954	0.962	0.837	0.884	0.859	0.713	0.766	0.739	0.853	0.932	0.931

Table B.1.7: Results from Experiment 1 for the BERTweet models fine-tuned on Dataset T.

BERTweet	**Unrelated**			**ProED**			**Prorecovery**			**Macro**	**Weight.**	**Acc**
Model	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**Avg F1**	**Avg F1**	
B = 16, LR = 1e-5	**0.988**	0.871	**0.926**	0.617	**0.964**	**0.752**	0.683	**0.766**	**0.722**	**0.800**	**0.886**	**0.878**
B = 32, LR = 1e-5	**0.957**	**0.928**	**0.942**	**0.712**	0.823	**0.764**	**0.710**	0.710	0.710	**0.805**	**0.900**	**0.897**
B = 64, LR = 2e-5	0.955	0.885	0.919	0.676	0.711	0.720	**0.500**	**0.776**	0.608	0.749	0.869	0.861

Table B.1.8: Results from Experiment 1 for the BERTweet models fine-tuned on Dataset T*.

Models	**Unrelated**			**ProED**			**Pro-recovery**			**Macro**	**Weight.**	**Acc**
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**Avg F1**	**Avg F1**	
DistilBERT, B = 16, LR = 5e-5	0.946	**0.968**	0.957	**0.831**	0.791	0.811	**0.798**	0.664	0.724	0.831	0.919	0.921
DistilBERT, B = 32, LR = 5e-5	**0.971**	0.955	**0.963**	0.818	**0.867**	**0.842**	0.717	0.757	**0.736**	0.847	**0.930**	**0.929**
DistilBERT, B = 16, LR = 2e-5	0.967	0.948	0.957	0.823	0.823	0.823	0.632	**0.785**	0.700	0.827	0.920	0.918
DistilBERT, B = 32, LR = 2e-5	0.968	0.955	0.962	0.810	0.855	0.832	0.721	0.748	0.734	0.843	0.927	0.926

Table B.1.9: Results from Experiment 1 for the DistilBERT models fine-tuned on Dataset T.

Models	**Unrelated**			**ProED**			**Pro-recovery**			**Macro Avg F1**	**Weight. Avg F1**	**Acc**
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**			
DistilBERT, B = 16, LR = 5e-5	0.980	**0.899**	0.938	**0.736**	0.884	0.803	**0.549**	**0.832**	0.662	0.801	0.899	0.893
DistilBERT, B = 32, LR = 5e-5	**0.979**	0.890	**0.933**	0.660	**0.920**	**0.768**	0.635	0.748	**0.687**	0.796	**0.892**	**0.885**
DistilBERT, B = 16, LR = 2e-5	0.969	**0.948**	**0.958**	**0.777**	0.884	**0.827**	**0.792**	**0.748**	**0.769**	**0.852**	**0.926**	**0.925**
DistilBERT, B = 32, LR = 2e-5	**0.984**	0.918	0.950	0.717	**0.924**	0.807	0.702	0.794	0.746	0.834	0.915	0.911

Table B.1.10: Results from Experiment 1 for the DistilBERT models fine-tuned on Dataset T*.

Models	**Unrelated**			**ProED**			**Prorecovery**			**Macro Avg F1**	**Weight. Avg F1**	**Acc**
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**			
RoBERTa, B = 16, LR = 2e-5	0.959	**0.957**	0.958	**0.809**	0.831	**0.820**	**0.767**	**0.731**	0.749	0.842	0.923	**0.923**
RoBERTa, B = 32, LR = 2e-5	**0.974**	0.931	**0.952**	0.726	**0.896**	**0.802**	0.729	0.722	**0.726**	0.826	**0.914**	**0.912**
RoBERTa, B = 16, LR = 1e-5	0.968	**0.950**	**0.959**	**0.774**	0.851	**0.811**	**0.776**	**0.769**	**0.772**	**0.847**	**0.924**	**0.923**
RoBERTa, B = 32, LR = 1e-5	**0.953**	0.955	0.954	0.799	**0.783**	0.791	0.703	0.722	0.712	0.819	0.913	0.913

Table B.1.11: Results from Experiment 1 for the RoBERTa models fine-tuned on Dataset T.

Models	**Unrelated**			**ProED**			**Pro-recovery**			**Macro Avg F1**	**Weight. Avg F1**	**Acc**
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**			
RoBERTa, B = 16, LR = 2e-5	**0.970**	0.924	**0.946**	0.697	**0.896**	**0.784**	**0.808**	0.741	**0.773**	**0.834**	**0.910**	**0.907**
RoBERTa, B = 32, LR = 2e-5	**0.971**	**0.926**	**0.948**	**0.721**	0.859	**0.784**	**0.683**	**0.759**	0.719	**0.817**	0.908	0.905

Table B.1.12: Results from Experiment 1 for the RoBERTa models fine-tuned on Dataset T*.

Frikk Hald Andersen, Eirik Dahlen

Sesame Street Pays Attention to Pro-Eating Disorder

NTNU

Kunnskap for en bedre verden