LUND UNIVERSITY

SCHOOL OF ECONOMICS AND MANAGEMENT

Master's Programme in Data Analysis and Business Economics

Harnessing AI for Suicidal Ideation Detection

Thoroughly Evaluating and Fine-Tuning Transformer Models to Identify Suicidal Ideation in Social Media Posts

by

Johannes Altnäs

Keshav Sompura

Master's Thesis - DABN01
May 2023
Supervisor: Najmeh Abiri

Abstract

This thesis explored the application of pre-trained transformer models in detecting suicidal ideation in social media posts. We leveraged social media data from platforms like Reddit and Twitter and applied a robust hyperparameter random search strategy to fine-tune and evaluate existing transformer models. Despite noise in the fine-tuning data, the models demonstrated high performance in identifying posts about suicidal ideation. However, performance as measured by F1 and average precision scores decreased when the models where applied to more realistic scenarios, indicating the need for inclusion of more general social media data during fine-tuning. Despite this, most models retained high recall scores, capturing a majority of true suicidal ideation cases. Practical implications of these findings could influence the methods used by social media organizations and mental health professionals in identifying suicidal ideation, leading to faster interventions. To advance this research, future work should aim at obtaining more robustly annotated data, expanding the hyperparameter search range, and exploring other efficient model architectures.

Acknowledgements

During this thesis process we have had the pleasure to work with several knowledgeable and helpful people.

First of we would like to thank Dr. Zachary Kaminsky (PhD), Associate Professor at Ottawa University and Chair of Suicide Prevention Research at The Royal's Institute of Mental Health Research, for his willingness to discuss and answer questions about the topic and his experience with predicting suicide risk as well as for granting us access to data to be used in our thesis.

We would also like to thank Dr. Cheri McDonald (PhD LMFT) founder and Clinical Director of A Place to Turn for answering our questions about the psychology behind suicidal behaviour and for reviewing the accuracy on some of the datasets used in this study.

Next we would like to thank Dr. Shaoxiong Ji (PhD), a doctoral researcher at the University of Helsinki within the Department of Digital Humanities currently working in the Language Technology research group, for inspiring our work and granting us access to data that was used in our thesis.

Finally, we would like to express our heartfelt gratitude to our supervisor, Dr. Najmeh Abiri (PhD) for her support, guidance, and encouragement throughout this obstacle laden thesis process. Her knowledge, patience, and availability have been crucial in shaping the direction and quality of our research. Thank you for making a significant difference during this process and being an outstanding supervisor and mentor.

Dedication

I dedicate this thesis to my parents, for their lifelong support, my partner, for unparalleled patience and encouragement, and to those struggling with mental health challenges, in hope of better recognition and support.

- *Johannes*

I dedicate this thesis to my family for their support and encouragement in all I do, and to those who are close to me who struggle with their own personal mental health battles.

- *Keshav*

Contents

1

Introduction

Suicide is one of the leading causes of death worldwide, with more than 700,000 attributed deaths each year (World Health Organization, 2021). One strong indicator of lifetime risk of suicide is suicidal ideation (henceforth referred to as SI), which generally refers to contemplation or preoccupation with suicide and/or death, although it does not have a formally accepted definition (Harmer et al., 2023). Both suicide and SI are multifaceted issues and their risk factors include both individual attributes such as internal and external psychopathology (Harmer et al., 2023) as well as socioeconomic, environmental and cultural factors (Zhang et al., 2022).

In recent years the rapid development of the Natural Language Processing (NLP) field of research has sparked a growing interest in analyzing the linguistic patterns of electronic health records (EHR), electronic messages, and social media posts in order to identify early signs of mental health concerns, such as depression, PTSD, anxiety, and SI (Greco et al., 2023). This evolving area of study has potential to assist mental health professionals in evaluation and diagnosis, as well as enabling quick and efficient targeted actions. Early detection of people at risk for suicide is identified as a central component in World Health Organization (2018)'s suicide prevention strategies, and one of the E's in their LIVE LIFE model approach. The application of NLP tools for mental health indicators could prove instrumental in identifying expressions or indications of SI that may otherwise go unnoticed or underreported.

The utilization and analysis of text posted to social media in particular has been given a lot of attention, in large part owing to the vast amounts of such data available (Zhang et al., 2022). Regarding the relevancy of such data, Park et al. (2012) found evidence that the language used on social media can help identify users with depression and might be suitable for clinical studies and Balani and De Choudhury (2015) found that users on social networks (Reddit in particular) had a high rate of self-disclosure related to mental health, where self-disclosure is described as "the process of making the self known to others" (Joinson and Paine, 2009, p. 235). Balani and De Choudhury (2015) notes that higher levels of self-disclosure helps facilitate identification of mental health issues. Regarding suicide and SI in particular, emergency room assessments by Belfor et al. (2012) found that adolescents, in addition to communicating suicidality verbally, sometimes do it by electronic means, including direct messages and social network posts. The authors also report an

increase in the number of electronic communications of suicidality over the period studied, and further notes that the majority of such communications only lead to an emergency room assessment because a peer of the adolescent makes the communication known to an adult. As such, it is likely that many adolescents do not receive the care they require or receive such care late.

At the same time as the NLP field's rapid development to enable the detection of mental health issues, AI generated text has increasingly become a viable option to provide responses to people seeking or in need of support. According to Dr. Cheri McDonald (personal communication, 17 April 2023), tailored AI generated support messages could provide more effective empathetic support to the receiver, especially compared to a standardized message that simply states where mental health resources can be found. In fact, a study evaluating responses given to users in the Reddit community "AskDocs" found that evaluators graded responses by chatbots higher in both quality and empathy dimensions as compared to answers from actual physicians (Ayers et al., 2023).

While there is a growing interest in leveraging advancements in language models, especially transformer-based models, for mental health applications, comparison and evaluation of different models has been somewhat problematic in the literature. A comparison study by Casola et al. (2022) notes that 80% of papers looked at in the pre-trained transformer model literature lack a clear and comprehensive model selection process. Furthermore, a quarter of the papers fail to mention a model selection strategy or final configuration, and while 38% of the articles reviewed do show the optimally tuned configuration, Casola et al. (2022) report that they fail to explain the search strategy or search spaces used. These gaps highlight the need for more systematic and transparent reporting of the training, fine-tuning, evaluation and application processes of transformer-based models in general, including in mental health research.

This study seeks to advance the intersection of natural language processing and mental health by using robust and reliable methods to fine-tune and evaluate several high-performing transformer models applied to the specific task of identifying suicidal ideation in social media posts. To ensure robust and reliable results, we employ random search to optimize the models' hyperparameters and utilize stable and replicable methods throughout the process.

The study will involve the collection and preprocessing of several datasets of social media posts, both with and without expressions of SI. In this study, we focus on examining data extracted from two prominent social media platforms, Reddit and Twitter. We are further focusing on English posts, as while detection in other languages is important, the current landscape of pre-trained transformer models is predominantly centered around the English language. The performances of these models will be compared against each other to identify the most effective models for detecting suicidal ideation in social media posts.

By exploring and evaluating the capabilities of pre-trained transformer models in this context, this study aims to contribute to the development of more accurate and efficient tools for identifying individuals at risk, hopefully leading to improved mental health care and support.

2

Data

2.1 Overview

Social media data is the focus of this study and the data used comes from the social media sites Reddit and Twitter.

The data used in this study originally came in several different forms, however only two main features are used to train and fine-tune the different models. Those two are the text message (e.g. the social media post) and the assigned label of suicidal ideation (SI) or non-suicidal ideation (non-SI) for each observation. No identifying features such as usernames, ages, or gender identities were used or inferred in this study.

The total data observations obtained for this study come from a combination of different data sources, some being open-source, others being obtained from previous research authors and institutions.

2.2 Reddit Data

Reddit is a free social media that is publicly available and is structured into different communities, known as subreddits, where people are free to post about different areas of interest, Reddit has over 100,000 "Active Communities" and more than 13 billion "Posts & Comments" (RedditInc, nd). There are communities that focus on mental health topics such as those relating to major mood disorders, including suicidal thoughts, anxiety, and depression; it is from a select few of these communities that the Reddit data comes from. The number of monthly users of Reddit is estimated to be around 1.66 billion (Turner, 2023b). Users of Reddit tend to be in younger age ranges, with 36% reported in the age bracket of 18-29, going down to 3% in the 65+ age bracket, as seen in figure 2.1 (Pew Research Center, 2021).

2.2.1 Suicide and Depression Detection

One of our sources comes from the website "Kaggle" which hosts the publicly available dataset titled "Suicide and Depression Detection". The dataset was published by the user "Nikhileswar Komati" and contains N=232,072 Reddit posts from the

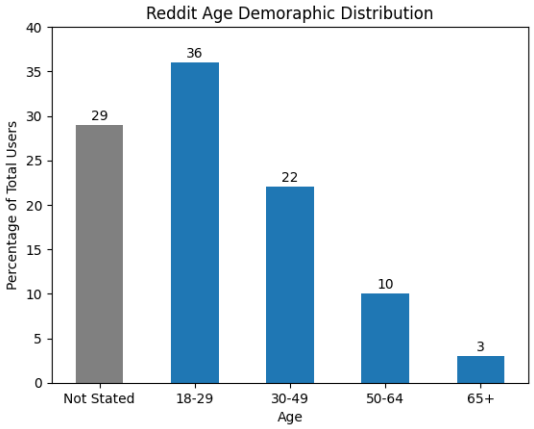


Figure 2.1: This figure depicts the age demographic distribution of the social media site "Reddit". The data comes from the Pew Research Center and is based on a survey of U.S. adults conducted from January 25 to Feb. 8, 2021. Figure source: Pew Research Center (2021)

Table 2.1: Overview information about the Reddit Suicide and Depression Detection dataset

Label	Number of Posts	Percentage of Total Posts
non-SI	116,035	50.00
SI	116,037	50.00

communities "SuicideWatch" and "depression". The posts were collected by Komati using Pushift API. The collection timeline from the community "SuicideWatch" was from December 16th, 2008, to January 2nd, 2021; the collection timeline from the community "depression" was from January 1st, 2009, to January 2nd, 2021 (Komati, 2021).

The method for labeling "SI" or "non-SI", while not explicitly stated, is assumed to be observations from the community "SuicideWatch" being labeled as "SI" and posts from the community "depression" being labeled as "non-SI". A subsection of the dataset consisting of 50% "SI" and 50% "non-SI" observations were professionally evaluated by Dr. Cheri McDonald (personal communication, 8 May 2023) as a part of this thesis and the assigned labels were determined to be 83% accurate, with a margin of error on the full dataset of roughly 10%.

Table 2.1 above contains information regarding the distribution of the two different class labels for the Reddit posts.

2.2.2 Reddit SuicideWatch and Mental Health Collection

Another utilized Reddit dataset comes from the group of researchers Ji et al. (2021) who kindly allowed us to use the Reddit dataset they collected. The dataset is named as "Reddit SuicideWatch and Mental Health Collection for Suicidal Ideation and Mental Disorder Detection" and will be referred to in the rest of this study as "SWMH". The dataset is made up of N=46,085 text posts from several different communities within Reddit relating to different areas of mental health, specif-

Table 2.2: Overview information about the Reddit SWMH dataset

Reddit Community	Number of Posts	Percentage of Total Posts
depression	18,739	34.47
SuicideWatch	10,181	18.73
Anxiety	9,549	17.56
bipolar	7,616	14.01

ically suicidality, depression, bipolar, and anxiety. The dataset was collected by the research group using the Reddit official API and a web spider created by the researchers (Ji et al., 2021). The collection timeline was not specified by the researchers. Similar to the previous dataset, when used in this thesis posts from "SuicideWatch" are assumed to be expressions of SI while posts from other Reddit communities are labeled as non-SI. Due to the agreements signed, the dataset was not shared with outside professionals, but it's label accuracy was evaluated by the authors of this thesis to be 89% with a margin of error around roughly 10%.

Table 2.2 contains the specific Reddit communities the posts came from as well as the number of posts collected from each community.

2.2.3 Reddit Self-Post Classification Task Dataset

A final Reddit dataset was collected which contains posts from communities that represent a broader spectrum of topics to simulate a more generalized social media context. The dataset, hosted on Kaggle by the user Mike Swarbrick Jones, is titled "Reddit Self-Post Classification Task" or RSPCT and contains about 1.013 million posts from 1,013 different Reddit communities, each with 1,000 examples per community (Jones, 2018). The post were collected from a two year period, beginning from June 1st, 2016 and ending June 1st, 2018 (Kwasny et al., 2023).

2.3 Twitter Data

Twitter is also a free social media platform where users post, or "tweet" into the public space. Unlike with Reddit posts, tweets are limited to a maximum of 280 characters. As of 2022, Twitter has an estimated 450 million monthly active users (Turner, 2023a). Similar to Reddit, the age demographics follow the pattern of a larger number of users falling into younger age ranges. Of a group of people surveyed by the Pew Research Center, 42% of people belonged to the 18-29 year old age bracket while 7% of people belonged to the 65+ year old age bracket, as seen in figure 2.2 (Pew Research Center, 2021).

2.3.1 Twitter Suicidal Intention Dataset

One of the Twitter datasets used is publicly available and contains a total of N=8,786 observations, each consisting of a tweet and a label indicating the absence or presence of suicidal intent. The dataset was obtained from the Github user Laxmi Kant who

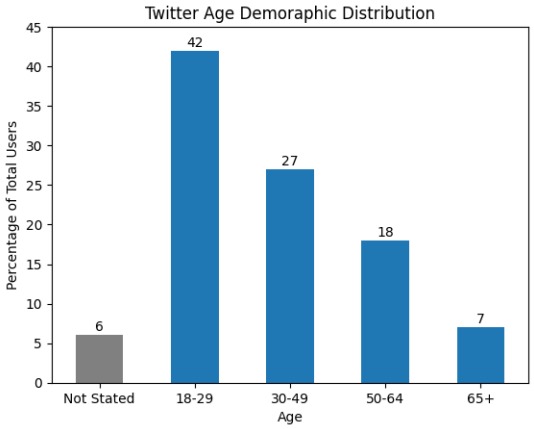


Figure 2.2: This figure depicts the age demographic distribution of the social media site "Twitter". The data comes the Pew Research Center from a survey of U.S. adults conducted January 25 to Feb. 8, 2021. Figure source: Pew Research Center (2021)

Table 2.3: Overview information about the Twitter Suicidal Intention dataset

Label	Number of Posts	Percentage of Total Posts
non-SI	4,828	54.95
SI	3,958	45.05

has the data hosted in a public github repository named "twitter-suicidal-intention-dataset" (Kant, 2020). The timeline of tweet collection is not stated and the method of classifying the tweets was not mentioned. As part of this thesis a subsection of observations, 50% labeled "SI" and 50% labeled "non-SI", were professionally evaluated by Dr. Cheri McDonald (personal communication, 8 May 2023) and the assigned labels were determined to be accurate in 85% of instances. Applied to the whole dataset this comes with a margin of error of roughly 10%.

Table 2.3 contains the number and ratio of observations in each class.

2.3.2 Twitter IMHR Dataset

The second of our Twitter datasets comes from the data collected by Dr. Zachary Kaminsky for the study conducted by Roy et al. (2020) and provided by the University of Ottawa's Institute of Mental Health Research (IMHR). The dataset consists of N=4,126,811 twitter posts, originally collected for the task of predicting future suicidal risk among individual twitter users.

The tweets were originally collected by accessing the Twitter API (Roy et al., 2020). The authors state that collection of the tweets took place over several years, beginning in September of 2016 and continued until June 2019, and was conducted through weekly keyword queries (Roy et al., 2020). During the collection period, the data was professionally evaluated and labeled. For further information about the data collection and processing, we advise to look over the methods section in the original paper.

Table 2.4 contains further information about the label distribution of the tweets.

Table 2.4: Overview information about the Twitter IMHR dataset.

Label	Number of Posts	Percentage of Total Posts
SI	1,481	0.0359
non-SI	4,125,330	99.9641

2.4 Data Security

Given the sensitive nature of this project and the data used, while publicly available, the data was handled with security measures to ensure the privacy of the users behind the messages. These security measures include storing the data in password protected environments on local memory systems, with limited access given only to the authors. It is important to note that no user identification, location, or date-time information features were collected and no attempt was made to identify the users based on the posts. During the processing of the social media data, care was also taken to ensure no extraneous processing of the data occurred beyond what was needed to prepare the data for the language models. Upon completion of this work and its relating processes, the data will be permanently deleted from the memory systems used.

Additionally, this thesis was designed to adhere to the guidelines for ethical research in the context of health research using social media data put forward by Benton et al. (2017) to the greatest possible extent. This includes, among other things, protection of sensitive data, de-identifying messages and posts for analysis if applicable, avoid sharing of sensitive or personal information and restricting data access. The research methodology and data handling processes have been carefully crafted to ensure compliance with these guidelines as well as relevant EU law, while at the same time striving to generate meaningful insights that contribute to the field.

Agreements for the handling of data where signed in the cases of the SWMH and IMHR datasets, and consultation sessions with University experts in the field of data processing were had in order to ensure legal and ethical compliance. While necessary, as a consequence of these procedures, the initiation of data processing and model training was substantially delayed within the thesis timeline, thereby influencing the scope and methodologies employed.

3

Theory

3.1 Suicidal Ideation

The term suicidal ideation (SI) is used to describe contemplations, preoccupations and thoughts about death (Harmer et al., 2023) and is an immediate and clinical precursor to suicide (Bruce et al., 2004; Nock et al., 2008). The literature generally makes difference between two types of SI, called "active" or "passive" SI. Active SI refers to ongoing and specific thoughts of suicide where the ideator has a desire to self harm and has a non-zero level of intention or expectation for death to be the outcome (Harmer et al., 2023). Passive SI on the other hand refers to a wish for death but without the acute intent or desire for self harm, although it includes indifference towards actions that might lead to accidental deaths (Harmer et al., 2023). Harmer et al. (2023) notes that although a common misconception, the risk of suicide is not greater for individuals with active SI than for those of passive SI.

The possible risk factors that lead to suicidality has been a topic for research for over 50 years, but despite this there is no proven predictor that reliably preforms better than chance when it comes to predicting who will end their life (Harmer et al., 2023). In meta-analysis only two factors out of several thousand, hopelessness and previous SI, showed some promise but are in general weak predictors (Harmer et al., 2023). Other studies researching the risk factors for SI also report socio-demographic factors such as employment status and age, as well as previous history of mental illnesses (Borges et al., 2008) as predictors for SI. Additionally, studies using the WHO-5 questionnaire for assessment of mental well being have found that individuals with suicidal ideation score significantly lower than individuals without (Awata et al., 2007). The landscape is complex, and Harmer et al. (2023) states that current theories generally suggest SI and suicidality arise from intricate interplays among many different factors.

According to Nock et al. (2008), the cross-national lifetime prevalence of suicidal ideation is 9.2%, and among these the conditional probability of a suicide attempt is 29%. The authors also found that risk of attempts was at its highest the first year of ideation. Yet, previous studies in the US have found that only about a third of those who attempt sought healthcare beforehand (Harmer et al., 2023), and a similar sentiment was reported by Belfor et al. (2012), who in regards to adolescents who

communicated suicidality via electronic means noted that a large portion might not get the help they need in time if no peer notifies an adult or professional. These findings underscore the potential benefits of implementing efficient early detection mechanisms, for example on platforms like social media, to identify and support individuals exhibiting signs of SI quickly. This, in turn, could potentially mitigate the risk of attempts, particularly within the critical first year of ideation.

3.2 Natural Language Processing and Mental Health

Natural language processing (NLP) is a subsection of computer science and AI research and an umbrella term for the various techniques that focuses on the interaction between humans and computers through the medium of language. It aims to enable machines to understand, interpret, and generate text in a manner that resembles human-like comprehension and communication. In recent years NLP techniques have gained significant traction due to advancements in machine learning algorithms, vast data availability, and increased computational power. NLP techniques can facilitate various tasks as it relates to the analysis of textual data, such as feature extraction, emotion and sentiment analysis, text classification, translation and much more. A growing area of research within psychology and mental health is concerned with the screening and detection of mental illnesses based on text data, such as social media posts, electronic health records (EHR) or interviews (Roy et al., 2020; Zhang et al., 2022).

In their review paper, Zhang et al. (2022) explores studies relating to the detection of mental illnesses using natural language processing between the years 2012 and 2021. The authors found that traditional machine learning techniques have been dominant for the majority of this period, but that deep learning based approaches have become more popular in recent years, with more papers utilizing deep learning approaches being published in 2021 than papers relying on traditional machine learning strategies.

As explained by Minaee et al. (2021), traditional machine learning approaches to text classification typically extracts features from the text, such as statistical properties of words, or otherwise present the text in an easier way for the model to process using techniques like Bag-Of-Words, N-gram, Term Frequency-Inverse Document Frequency (TF-IDF), GloVe and word2vec. The authors further state that such methods traditionally follow a two-step approach, which starts with extracting of designed features that are then fed into classification algorithms. This two-step approach however has some drawbacks, such as requiring a lot of work with feature engineering and analysis to achieve good performance, being harder to adapt to new tasks due to the importance of domain knowledge and being unable to fully capitalize on vast training data, as the features or feature templates are predetermined (Minaee et al., 2021).

Minaee et al. (2021) further provides a short history of deep learning approaches to text classification, such as feed-forward networks handling text like bag-of-words, RNN-based models such as Long Short-Term Memory (LSTM) models viewing the text like a sequence and enabling capture of long term dependencies and CNN-based models that looks for text patterns in space rather than sequential time. The authors

notes the emergence of models incorporating the concept of attention and hybrid models as well as Graph and Siamese neural networks.

Lately however, the standard approach for many NLP tasks have been the transformer models (Vaswani et al., 2017) (described in section 3.3), applicable for tasks such as translation, sentiment analysis and sequence classification (Casola et al., 2022). Transformer models addressed computational bottlenecks faced by both RNNs and CNNs, allowing for more parallelization, and enabling the training of large models on previous impossibly big datasets through the utilization of GPUs (Minaee et al., 2021). This enables the utilization of language models pre-trained on large amounts of data in order to learn contextual text representations. The most common and widely applied such models are BERT (Devlin et al., 2019) (described in section 3.3.1) or BERT derivatives such as RoBERTA (Liu et al., 2019) or DistilBERT (Sanh et al., 2020) (described in sections 3.3.2 and 3.3.3 respectively).

Reviewing the type of data used for NLP studies in the mental health sphere, Zhang et al. (2022) found that the most common type of datasets are social media texts, accounting for roughly 80% of the data used by the reviewed studies. These findings were corroborated by other empirical studies (Greco et al., 2023). Other popular data sources found were EHRs and interviews. In regards to evaluation of model performance, Zhang et al. (2022) found the most common metrics used in the context of NLP and mental health to be accuracy, recall, precision and F1-score. These metrics are the most utilized for text classification in general, including for deep learning and transformer based models (Minaee et al., 2021).

In regards to detection of suicidal ideation, previous studies have indicated that BERT models outperform more conventional Deep Learning approaches like BiLSTM (Bidirectional Long Short- Term Memory) (Haque et al., 2020). These studies mainly utilize the base models and not models pre-trained for the purpose, recently however models adapted to the field of mental health have begun to emerge. For example, Ji et al. (2021) further developed transformer models in the field of mental health and created models pre-trained on mental health data. These models were shown to perform well in a wide variety of downstream mental health tasks, including classification of several mental illnesses, suicidal ideation among them, with improved performance compared to many existing models.

3.3 Transformers

The genesis of contemporary transformer models can be traced back to a seminal paper published in 2017. The paper is titled "Attention is All You Need" and was created by Vaswani et al. (2017) who primarily consist of Google researchers, where they unveiled the transformer neural network architecture. This method facilitated training on extensive and varied corpora, yielding high-performance on NLP tasks, even with a limited amount of task-specific fine-tuning data.

Transformer models are based on the concept of self-attention, which is used to relate the various positions of a single sequence, such as the words in a sentence or paragraph, to generate a representation of the sequential space (Vaswani et al., 2017). In short, attention enables the model to understand the relationships between words in a sentence by making them "pay attention" to other words, considering

Table 3.1: This table, originally from Vaswani et al. (2017), compares various layer types in terms of maximum path lengths, per-layer complexity, and the minimum number of sequential operations required. The variables 'n', 'd', 'k', and 'r' represent sequence length, representation dimension, kernel size of convolutions, and the size of the neighborhood in restricted self-attention, respectively. Table source: Vaswani et al. (2017)

Layer Type	Complexity per Layer	Sequential Operations	Maximum Path Length
Self-Attention	$O(n^2 * d)$	$O(1)$	$O(1)$
Recurrent	$O(n * d^2)$	O(n)	$O(n)$
Convolutional	$O(k * n * d^2)$	$O(1)$	$O(log_k(n))$
Self-Attention (restricted)	$O(r * n * d)$	$O(1)$	$O(n/r)$

their importance when generating a meaningful representation. This helps the model capture context and inter-word relationships better, improving the models ability to understand and generate human-like text.

This transformer architecture addresses the previous limitation that constrained traditional RNN, CNN, and LSTM models that were used for similar NLP tasks such as text classification. The previous models could not compute tasks in parallel and as a result ran into memory constraints when working with longer sequential tasks whereas these transformer models can handle larger amounts of data (Vaswani et al., 2017). This is done through the utilization of self-attention to compute attention scores for every word in parallel, modeling the influence of each word against all other words. This feature allows for greater parallelization compared to CNNs and RNNs, enabling efficient training of large models on substantial data using GPUs (Minaee et al., 2021). As a result, transformer models are more computationally efficient, reducing computational time compared to RNNs and CNNs through a reduction in maximum path length, as shown in table 3.1 (Vaswani et al., 2017). This allows for transformers to create stronger context connections across longer distances, increasing its predictive capabilities.

The architecture of transformers is depicted in figure 3.1 (Vaswani et al., 2017) and works as follows. The structure is divided into an encoder stack and a decoder stack. The encoder stack first takes in an input sequence which is tokenized into an embedding representation space, where words with similar meanings have similar vector representations, and then a positional embedding is added to the tokenized input embeddings (Vaswani et al., 2017). From there the tokenized sequence is then passed first into the multi-head attention layer where the attention mechanism takes place across multiple attention heads where different attention heads learn different relations within the same sequence, in our case a social media message. An example of this is can be seen in figure 3.2 (Vaswani et al., 2017). The output of these multi-head attention layers are then concatenated together and normalized where they are then fed into the second sublayer of the encoder, a fully connected feed forward neural network. The outputs from the neural network are then added together from each encoder stack layer and normalized to produce an output which can then be fed into the decoder stack. Throughout the encoder process, residual connections are made around the multi-head attention layer and the feed forward layer (Vaswani

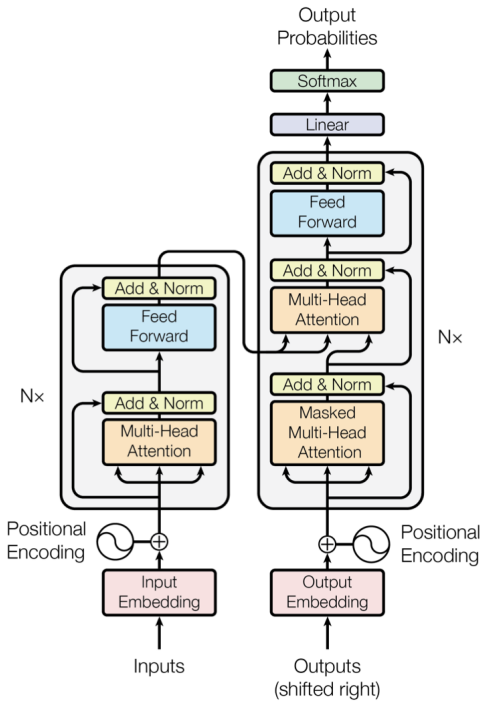


Figure 3.1: This figure depicts the transformer model architecture. The encoder stack makes up the left portion and the decoder stack makes up the right portion. Figure source: Vaswani et al. (2017)

et al., 2017). Residual connections were first seen in He et al. (2015)'s paper and works to alleviate the vanishing gradient problem by allowing the neural network to bypass layers during backpropagation. The reason for normalizing after each sublayer is not stated, however based on the work done by Ba et al. (2016), layer normalization increases training stability and can considerably decrease training time.

The overall purpose of the decoder stack is to generate a response based on what was originally input into the encoder stack. The decoder generates one output token at a time, using softmax to select the most probable token, to create a sequence; after each token generation the decoder auto-regressively utilizes the previously generated output tokens as an input into the decoder stack at each step (Vaswani et al., 2017). The decoder stack follows a similar structure, changing outputs into an embedding representation with a positional encoding feature. The decoder, however contains three sublayers, a masked multi-head attention layer, a standard multi-head attention layer, and a feed forward neural network layer, with concatenation and layer normalization after each sublayer in the decoder, as well as residual connections around each sublayer; this process is depicted in figure 3.1 (Vaswani et al., 2017). The masked multi-head attention layer is different from the standard multi-head attention layer in that it is not able to attend to tokens in subsequent positions, which is done to ensure predictions on only "known outputs" at each step (Vaswani et al., 2017).

These transformer models utilize the self-attention mechanism through multi-head attention layers as depicted in figure 3.3 (Vaswani et al., 2017). These multi-headed attention layers are primarily comprised of a form of attention Vaswani et al. (2017)

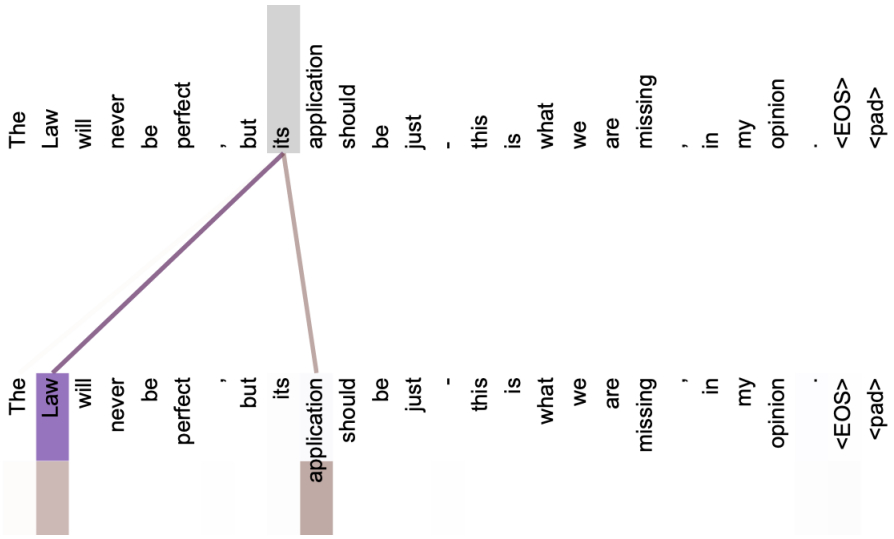


Figure 3.2: This figure depicts the isolated attention for just the word "its" from two different attention heads. The word "its" has a learned relation to the word "Law" in one attention head, and to the word "application" in another attention head. Figure source: Vaswani et al. (2017).

describe as "Scaled Dot-Product Attention". The attention operation essentially maps a query and a key-value pair to an output, all of which are structured into vector space for each token. The formula for how attention is calculated is shown in equation 3.1 below.

$$Attention(Q, K, V) = Softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{3.1}$$

The Q, K, and V matrices are the query, key and value matrices and d_k is the dimension of the queries and keys. They are derived by applying three linear transformations (by weight matrices W_{nxk}) to the embedding matrix of the input sequence (X_{mxn}). The dot-product $Q * K^T$ yields the attention score matrix, where a larger value means more attention since a higher dot product indicates more similar vectors. This is then scaled by d_k in order to avoid vanishing gradients before being compressed into the [0,1] space by the softmax function. Finally it is multiplied by the value matrix V which yields new adjusted embedding for the tokens. By forcing the tokens to consider each other we adjust the value matrix to better represent the whole sequence. By having multiple attention heads the model can apply multiple transformations to the embeddings, each with its own parameters and semantic focus.

The user "dontloo" on the *Cross Validated* forum gives an intuitive explanation to this attention mechanism.

> The key/value/query concept is analogous to retrieval systems. For example, when you search for videos on Youtube, the search engine will map your query (text in the search bar) against a set of keys (video title, description, etc.) associated with candidate videos in their database, then present you the best matched videos (values). (dontloo., 2019)

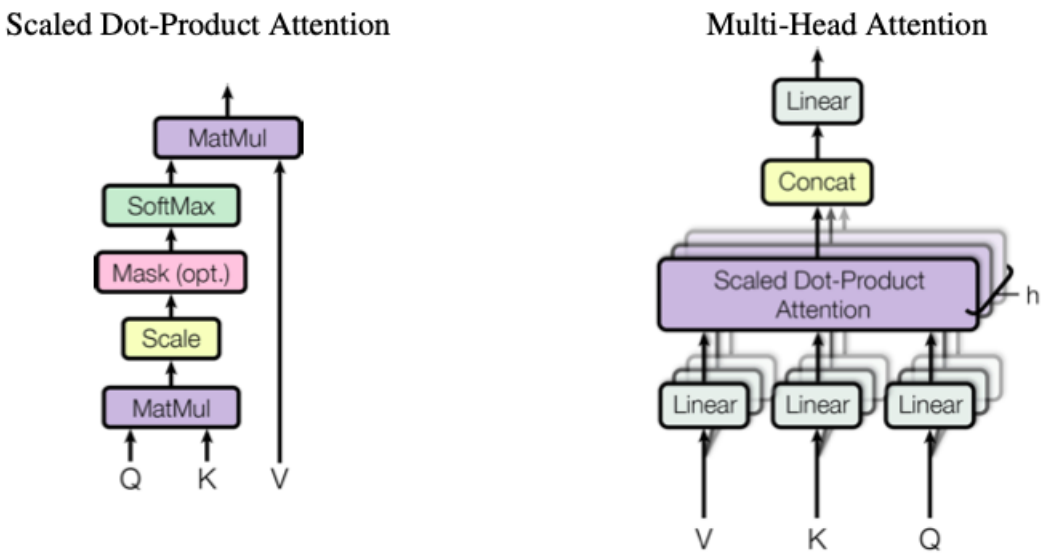


Figure 3.3: Composition of the Scaled Dot-Product Attention process (left) and the Multi-Head Attention process (right). Figure source: Vaswani et al. (2017).

Additionally, transformers utilize positional encoding to register the location of each token in the original input and output sequence. This is fundamental to the transformer architecture because they do not contain recurrent properties such as those that exist in recurrent neural networks (RNNs) and long short-term memory neural networks (LSTMs). The recurrent process for RNNs, and by relation LSTMs, is depicted in figure 3.4 (Goodfellow et al., 2016). These positional encodings, as described by Vaswani et al. (2017), insert information regarding the relative and absolute location of each token in the sequence.

It is this transformer architecture that forms the foundation of today's large language models such as OpenAI's ChatGPT and Google's Bard, as well as the pre-trained BERT models that make up the base of the models used in our study.

3.3.1 BERT

Bidirectional Encoder Representations from Transformer, known by the acronym BERT, was developed by Devlin et al. (2019) at Google AI Language and utilizes the transformer architecture and attention mechanism that was created by Vaswani et al. (2017). The key mechanism that makes BERT a versatile and computationally powerful model is its bidirectional ability to learn the context of text documents using transformers in combination with word embeddings. BERT is able to learn from left-to-right word contexts as well as right-to-left word contexts, which before ELMo (Peters et al., 2018) and its preceding paper by Peters et al. (2017), was a novel concept, with traditional methods only learning from left-to-right pre-trained embeddings. BERT advanced the NLP field because previous methods such as ELMo used bidirectional LSTMs which were "not deeply bidirectional" (Devlin et al., 2019) whereas the transformer framework allows for long distance word dependencies and contextual relations. BERT out performed previous state-of-the-art NLP methods and models on 11 standard NLP benchmark tasks, becoming the go to state-of-the-art model for a wide variety of NLP tasks.

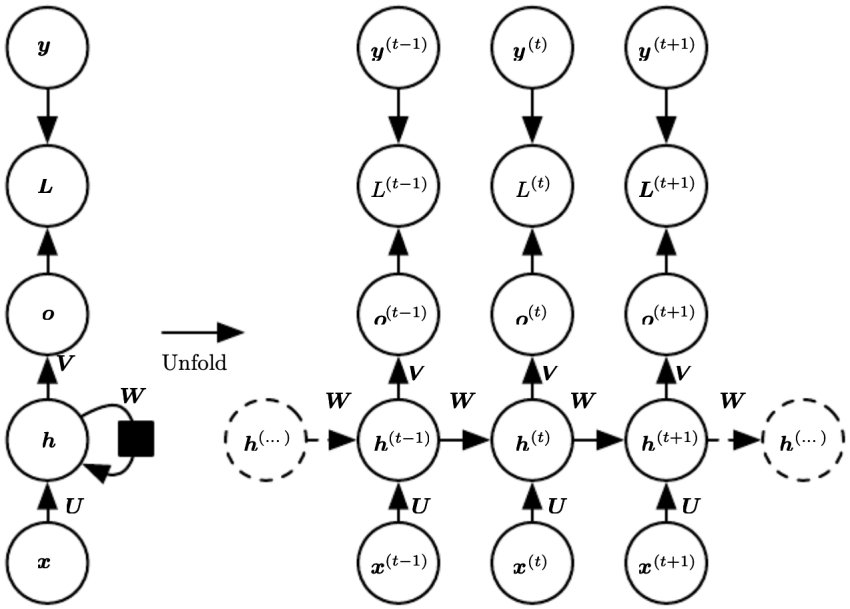


Figure 3.4: This figure depicts the "computational graph" of the recurrent process that make up the foundation of recurrent neural networks. Here $\boldsymbol{x}$ represents the input sequence values, $\boldsymbol{W}$ represents the weight matrix, $\boldsymbol{h}$ represents the hidden units, $\boldsymbol{o}$ represents the output values, $\boldsymbol{L}$ represents the Loss of each output, and $\boldsymbol{y}$ represents the target value. Figure source: Goodfellow et al. (2016).

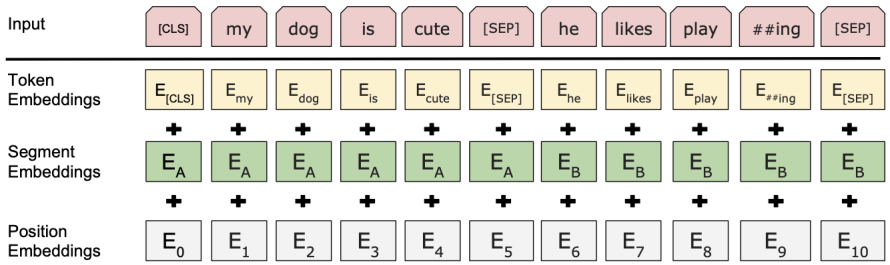


Figure 3.5: This figure depicts the processes that transforms the input sequence into the embedding representation that BERT then acts upon. Figure source: Devlin et al. (2019).

In addition to the position encoding of input tokens, BERT also includes the special pre-defined tokens "[CLS]" and "[SEP]" (Devlin et al., 2019). The [CLS] token is placed at the beginning of every sequence and is used as a way to aggregate knowledge for classification tasks (Devlin et al., 2019). The [SEP] token on the other hand is used to separate sentence pairs which are packed into a single sequence (Devlin et al., 2019). This process of adding and transforming the input sequence before the transformer stacks of BERT is illustrated in figure 3.5 (Devlin et al., 2019).

BERT is able to be applied to a variety of tasks because it has been pre-trained on a large dataset and because of its ability to be easily fine-tuned. BERT was pre-trained using English Wikipedia which contains 2,500 million words (Devlin et al., 2019) and BookCorpus which contains 800 million words that come from a large corpus of books collected by Zhu et al. (2015) (see Devlin et al., 2019). BERT was pre-trained to perform well on two tasks, the first being predicting masked words

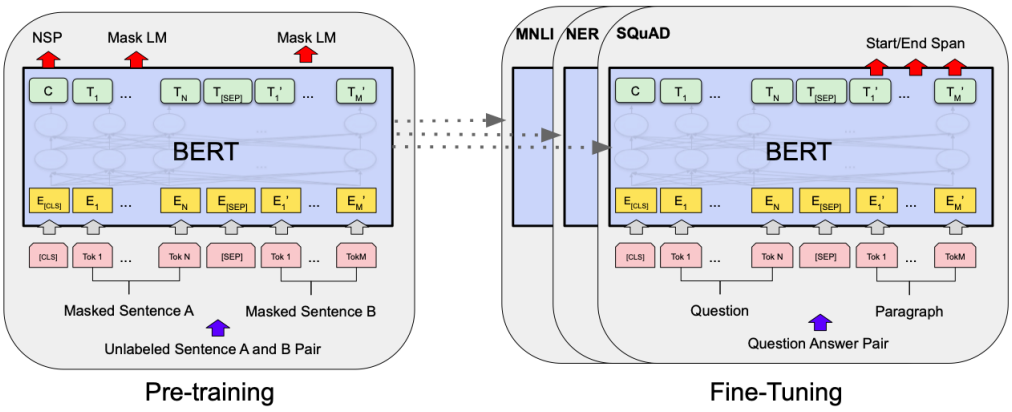


Figure 3.6: A high level perspective of the process of pre-training and fine-tuning for BERT. Excluding the output layers, the same architecture based on the transformer architecture by Vaswani et al. (2017) is used in both processes of BERT. Figure source: Devlin et al. (2019).

and the other being next sentence prediction (Devlin et al., 2019). However, it is important to note that during the pre-training process, unlabeled sentences were used so that BERT can be used as a generalized pre-trained model (Devlin et al., 2019). The pre-trained BERT model can be further fine-tuned for specific tasks by plugging "the task-specific inputs into BERT and fine-tune all the parameters end-to-end" (Devlin et al., 2019, p.5) with a relatively inexpensive amount of compute using a TPU or a GPU because of the transformer's ability to run tasks in parallel. An illustration of the pre-training and fine-tuning processes of BERT is depicted in figure 3.6 (Devlin et al., 2019).

In total, the base BERT model put forward by Devlin et al. (2019) is made up of 110 million parameters. Those parameters come from 12 transformer blocks, a hidden size of 768, and 12 self attention heads (Devlin et al., 2019). Based on this architecture, the base BERT model outperformed previous state-of-the-art NLP methods and models on benchmark tasks including the General Language Understanding Evaluation (GLUE), the Stanford Question Answering Dataset (SQuAD) v1.1 & v2.0, and the Situations with Adversarial Generations (SWAG) dataset (Devlin et al., 2019).

3.3.2 RoBERTa

Submitted in July 2019, Liu et al. (2019) at Facebook AI created an improved version of BERT under the name RoBERTa, standing for Robustly optimized BERT approach. The Facebook team sought to optimize Google AI's large BERT model which is made up of 24 transformer layers, a hidden size of 1024, and 16 attention heads for an approximate 355M parameters. They concluded that performance could be improved by

> ...training the model longer, with bigger batches over more data; removing the next sentence prediction objective; training on longer sequences; and dynamically changing the masking pattern applied to the training data (Liu et al., 2019, p.10).

Facebook AI experiments with larger batch sizes of 8K rather than the original BERT's 256, finding success in performance improvements while keeping total training quantity the same (Liu et al., 2019). Along with this, RoBERTa experimented with step sizes of 100K, 300K, and 500K compared BERT's original step size of 1M (Liu et al., 2019). The authors found performance improvements in the increasing of step size from 100K to 500K, while using a batch size of 8K for each step.

Specifically, RoBERTa trains using 160GB of uncompressed text data compared to the original 16GB of uncompressed text data. In addition to BookCorpus and English Wikipedia, the datasets CC-News (76GB) , OpenWebText (38GB), and Stories (31GB) are used for a larger training dataset, which respectively adds English news, general web, and story-like content (Liu et al., 2019).

As described by Liu et al. (2019), RoBERTa was trained using mixed precision floating point arithmetic. The reason is not stated, however mixed precision training can speed up computation while maintaining performance (Micikevicius et al., 2018).

In Devlin et al. (2019)'s pre-training of BERT, the masking of words was done once in a static manner, where each epoch of training saw same words masked repeatedly. Liu et al. (2019) found that masking words in a new pattern every time a sequence was introduced into the model resulted in a slight improvement and so was used to pre-train RoBERTa.

3.3.3 DistilBERT

A smaller, lightweight variant of BERT named DistilBERT was created by Sanh et al. (2020) at Hugging Face in October 2019. In short, Sanh et al. (2020) were able to distill BERT down in size by 40% (down to a total of 66 million parameters) and increase computational efficiency by 60% all "while retaining 97% of its language understanding capabilities". This was done through knowledge distillation, reducing architecture size, teacher-student initialization, and by applying best practices for pre-training BERT (Sanh et al., 2020).

DistilBERT was pre-trained using a technique known as knowledge distillation (Bucilu et al., 2006; Hinton et al., 2015) as discussed in Sanh et al. (2020) where a smaller student model (e.g. DistilBERT) learns from a larger teacher model (e.g. BERT). This is achieved through the introduction of a triple loss training combination. The first loss, L_{ce}, is created by distilling the "loss over the soft target probabilities of the teacher":

$$L_{ce} = \sum t_i * \log(s_i) \tag{3.2}$$

where t_i refers to the teacher and s_i refers to the student (Sanh et al., 2020). During training, *softmax-temperature* is used:

$$p_i = \frac{exp(z_i/T)}{\sum_j exp(z_j/T)} \tag{3.3}$$

which uses T to adjust the smoothness of the output distribution and the model score (z_i) for class i; where T is set to 1 when the model is used for testing and inference (Sanh et al., 2020). Along with L_{ce}, the loss of predicting masking words

(*masked language modeling*) L_{mlm} as used by BERT in Devlin et al. (2019), and a *cosine embedding loss* L_{cos} which causes the student and teacher hidden state vectors to become more equivalent, were used together to create the triple loss training combination for DistilBERT (Sanh et al., 2020).

The architecture of DistilBERT is shrunk by reducing the number of layers by a factor of 2 as compared to the general architecture of BERT. Along with this, token-type embeddings and the pooler were also removed to increase computational efficiency (Sanh et al., 2020). Additionally, BERT is also leveraged as a teacher network by adapting every other layer as an initialization for DistilBERT (Sanh et al., 2020).

Training of DistilBERT is also done using the best practices discovered during the creation of RoBERTa: using large batch sizes, dynamic masking, and excluding next sentence prediction loss (Liu et al., 2019; Sanh et al., 2020).

3.4 Transfer Learning

Transfer learning is the process of applying (transferring) knowledge gained by solving one task when solving other, similar, tasks. Transfer learning has become more and more prevalent within NLP research, where large language models like BERT or GPT are pre-trained by preforming various tasks on vast amounts of unlabeled data and then fine-tuned for more specific downstream tasks. When fine-tuning, the training process is continued with the parameters learned during pre-training but with a smaller, more task specific, and generally labeled data sets. When it comes to transformer based models like BERT, this fine-tuning process can be relatively fast due to the good internal representation of language the model gained during pre-training. Leveraging pre-trained models has been shown to yield superior performance across a variety of Natural Language Processing tasks (Houlsby et al., 2019).

3.5 Hyperparameter Optimization

Hyperparameter optimization is the process of tuning the hyperparameters of machine learning models to increase performance, especially when it comes to the wide range of options that exist in neural network based models (Feurer and Hutter, 2019). Hyperparameter optimization has a number of practical implications, including reducing the burden of manually researcher hyperparameter tuning, improving performance of models, and "improving the reproducibility and fairness of scientific studies" (Feurer and Hutter, 2019).

3.5.1 Random Search

Random search is a specific type of hyperparameter optimization method which randomly explores different combinations of hyperpameters within a predefined search space defined by the researcher. An illustration of random search compared against grid search is depicted in figure 3.7 (Bergstra and Bengio, 2012). The benefit of random search is that it is a simple, easily implemented process that serves as a

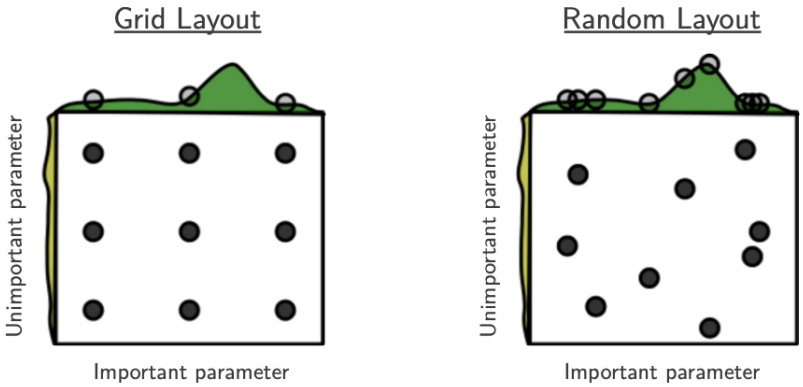


Figure 3.7: A depiction of grid search versus random search for hyperparameter optimization. Nine trials are performed where the grid layout searches equally across both hyperparameters where as the random layout randomly divides the trials across both hyperparameters. The results demonstrate the failure of grid search in higher dimensions, as compared to random search which is able to capture the important region of the hyperparameters because of its ability to explore. Figure source: Bergstra and Bengio (2012).

baseline hyperparameter optimization algorithm because it does not consider any assumptions about the model it is optimizing and will eventually reach a point of performance very close to the overall optimum (Feurer and Hutter, 2019). This is due to the fact that it is able to explore across higher dimensions due to its ability act in a random manner, additionally giving it the benefit of being computationally more efficient than a traditional grid search (Bergstra and Bengio, 2012).

4

Methods

4.1 Models

Previous studies have shown that further pre-training of transformer models that is domain or task relevant improves its performance on downstream tasks (Sun et al., 2020). For this reason, it's of interest to look at models pre-trained in the mental health domain to compare their performances at the specific task of classifying suicidal ideation. In this paper we fine-tune and evaluate a total of four pre-trained language models (PLMs), developed by Ji et al. (2022), Naseem et al. (2022) and Vajre et al. (2021). Additionally we evaluate DistilBERT, a smaller version of BERT which has not received domain or task relevant pre-training.

In their paper titled "MentalBERT: Publicly Available pre-trained Language Models for Mental Healthcare", Ji et al. (2022) develop and release two pre-trained language models (PLMs), MentalBERT and MentalRoBERTa, with the goal of benefiting machine learning in the mental health research space. The authors state that the models were additionally pre-trained using mental related Reddit data. The authors goes on to show that the released models preform better in mental health detection and classification tasks when compared to other pre-trained language models. We here explore the uncased versions of MentalBERT and MentalRoBERTa for fine-tuning, hyperparameter optimization and testing. Both of the models evaluated here are based on the same base BERT architecture which is discussed in the theory sections 3.3.1 and 3.3.2. Due to limitations in time and computational resources, all models developed by Ji et al. (2022) cannot be evaluated. Specifically, neither the cased nor the larger version of the MentalBERT variants will be fine-tuned and evaluated.

Naseem et al. (2022) developed and made available the PLM PHS-BERT, a pre-trained version of $BERT_{large}$ and the largest model used in this study. The model was developed for use in social media related public health surveillance (PHS) and benchmarked against existing PLMs. The authors state that PHS-BERT was additionally pre-trained using health related tweets using keywords related to disease, symptom, vaccine, and mental health. Some of the downstream tasks the model was fine-tuned and evaluated on included suicidal risk detection, stress detection and identification of depression in users. In all tasks the PHS-BERT showed improvement in F1 scored compared to the PLMs it was compared to. The models

performance on the detection of latent suicide risk was benchmarked against MentalBERT by Ji et al. (2022) using the R-SSD dataset by Cao et al. (2019) and showed better performance. Our data set, context, and specific problem is however different from the benchmarked task as the task here is to classify expressions of suicidal ideation and not latent suicide risk.

Finally we also evaluate the performance of the PLM PsychBERT, introduced by Vajre et al. (2021) as a BERT model pre-trained on both formal academic texts as well as informal text from social media regarding mental health. The model showed top performance in classification tasks related to mental health sequences but was not benchmarked against any of the other models in this study. PsychBERT was developed to improve interpretability and explainability compared to other classifiers in the field.

In addition to the PLMs outlined above, the DistilBERT model introduced by Sanh et al. (2020) is also included in this evaluation. The performance of DistilBERT in relation to the rest is of extra interest both because it's not pre-trained in the mental health domain and because it's much smaller in terms of memory occupied and is faster to train than the models described above. As such, should the model perform comparably well in detecting suicidal ideation it might be preferred for deployment in situations where computational resources are scarce, or perhaps serve as a basis for future, smaller, PLMs.

4.2 Metrics

The main metrics used to evaluate and test the models are accuracy, F1-score, which consists of recall and precision, and the average precision (AP) metric. The metrics are based on the models performance in correctly predicting true positive and true negatives, with an example of a true positive (SI) case being "I feel so alone, I just want it to stop. I don't plan on being around I just want to end it all" and an example of true negative (non-SI) being "This pizza is to die for!". Note that these examples are created by the authors for the sake of illustration and are not actual social media posts.

Precision is calculated by taking the number of true positive predictions and dividing it by all positive predictions, its formula presented in equation 4.1 below.

$$Precision = \frac{TruePositives}{TruePositives + FalsePositives} \tag{4.1}$$

A high precision value therefore means that the model is not producing a large ratio of false positives, which in our case means labeling non-SI posts as SI.

Recall is a complementary measure to precision, and is calculated by taking the number of positive predictions and dividing it with the number of observations which should have been predicted as positive. As such, recall measures the models ability to correctly identify positive cases, and a high recall value means that a high number of the positive (SI) observations are correctly classified. The formula for recall is presented in equation 4.2 below.

$$Recall = \frac{TruePositives}{TruePositives + FalseNegatives} \tag{4.2}$$

As neither precision or recall is enough alone to determine performance of a model, the F1 metric is commonly utilized. The F1 metric is the harmonic mean of both recall and precision, combining both aspects into one metric. The formula for the F1 metric is shown in equation 4.3 below.

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{4.3}$$

An alternative way to evaluate the models performance with one number is the average precision (AP) score. The AP score can be particularly useful in cases when the positive class (in this case, sequences expressing SI) may be rare compared to the negative class. Because it takes into account the ranking of the predicted probabilities that each sequence is an expression of SI, it provides a more informative measure of performance than just looking at overall accuracy. The equation for the AP score is listed below (4.4), which is the formula used by scikit learn, the module we employ for the metrics.

$$AP = \sum_{n} (R_n - R_{n-1}) * P_n \tag{4.4}$$

The notations R_n and P_n are the recall and precision at the nth threshold respectively (Scikit-Learn, nd).

4.3 Robust Random Search

In order to avoid issues regarding reproducibility and to avoid using unstable methods that are otherwise prevalent in the PLM literature (Casola et al., 2022), a robust approach to hyperparameter optimization was taken for the models included in this study. Specifically we preform a random search strategy for hyperparameter optimization, wherein each randomized hyperparameter combination is fine-tuned three times with different (random) seeds. As such, the performance of the model across the chosen metrics can be more reliably ascertained, and the results are reported with standard deviations.

A random search strategy is utilized as Bergstra and Bengio (2012) found the approach to be more computationally efficient than pure grid search, while still finding better models in a majority of the tested circumstances. The authors found the random approach especially efficient when search space is high-dimensional or where the impact of certain hyperparameters is much larger than others. Because each separate trial (evaluation of a combination of hyperparameters) in a randomized search are independent and identically distributed, meaning every individual trial is considered separate and doesn't depend on the outcome of the others, it's easy to add new trials, to stop and continue later or to utilize several units to parallelize the process (Bergstra and Bengio, 2012). These properties are valuable when working with limited resources.

Random search works by randomly drawing hyperparameter values from defined search space(s) or distribution(s) for a set number of iterations (trials). While there are pre-built, efficient modules available to perform these types of searches for transformer models, due to complications between versions of python, CUDA and these modules, as well as the time constraint of the project, a custom random search solution was instead implemented.

The hyperparameters explored are learning rate, weight decay, batch size, dropout rate, and precision type. They were selected based on their acknowledged impact on the performance of transformer models. There exist several additional hyperparameters such as optimizers, learning rate schedulers or warmup-ratios, but due to time and computational constraints we limit the parameters explored.

4.3.1 Search Space

A search space, which refers to the range of possible values that can be assigned to each hyperparameter, was defined for each of the selected hyperparameters. These search spaces were kept small due to limitations in both time and computational resources, and additional focus was instead put on making the results robust and replicable.

Learning Rate

The available options for the learning rate is set to the discrete finite space [5e-6, 1e-5, 5e-5]. Previous studies regarding BERT and its variants have generally kept to the 1e-5 to 5e-5 range when fine-tuning for downstream tasks (see Devlin et al., 2019; Liu et al., 2019 & Lan et al., 2020) and the papers the above models come from have stayed this course (see Ji et al., 2022; Naseem et al., 2022 & Vajre et al., 2021). Initial testing revealed that higher learning rates generally led to poor performance, but the case was not as clear for lower learning rates. As such, 5e-6 is also included in the search range.

Weight Decay

This hyperparameter specifies the decoupled weight decay (Loshchilov and Hutter, 2019) to apply to every layer in AdamW, excluding bias and layernorm (HuggingFace., nd). Appropriate weights vary per task and dataset; Devlin et al. (2019) employs a rate of 0.01 when fine-tuning BERT for downstream tasks while Liu et al. (2019) uses 0.1 or 0.01 depending on the context. We define the discrete search space to be [0.001, 0.01, 0.1].

Batch Size

Batch sizes are known to influence fine-tuning results and is one of the most common hyperparameters to tune. Previous studies have generally utilized batch sizes between 16-64 for fine-tuning BERT or BERT derived models for downstream tasks (Devlin et al., 2019; Liu et al., 2019; Sun et al., 2020), with some going so high as 128 (Lan et al., 2020). Here we employ a search space defined by [16, 32, 128], in order to try the most common choices as well as a larger than typical size. Larger batch sizes may also affect training time of the model.

It is important to note, however, that the search space defines *effective* batch size. Due to limitations in GPU memory, batch sizes over 8 can't be loaded simultaneously on the GPU. Larger effective batch sizes are achieved through gradient accumulation where, instead of updating the model weights after every batch, the gradients are accumulated over multiple smaller batches before performing an update. This allows for larger batch sizes than would otherwise be allowed by the GPU memory limit (HuggingFace, 2022). As such, the batch size on the GPU was always kept at 8 in order to maximize GPU utilization, except for PHS-BERT where the batch size on the device was set to one with similar effective GPU utilization, and larger effective batch sizes were achieved through gradient accumulation.

Dropout Rate

Dropout is a regularization technique that works to counter overfitting of models and to improve generalized performance by randomly turning off a certain fraction of neurons during each iteration (Srivastava et al., 2014). For BERT-related models, a dropout rate of 0.1 is very common (Devlin et al., 2019; Liu et al., 2019) but other rates are also prevalent and the rate is largely context dependent (Srivastava et al., 2014). We utilize a search space of [0.1, 0.3, 0.5] which covers a relatively wide range compared to existing approaches.

Precision type

Precision type refers to the format in which numbers are stored and processed during training and fine-tuning the models. Floating point 32-bit, or FP32, allocates as the name suggests 32-bits for storing each number which results in a large range of possible numbers and a high degree of precision and is generally the standard precision used. Floating point 16-bit (FP16) allocates half as many bits for storing numbers, leading to a lower range of available numbers and lower precision in exchange for a smaller memory footprint. Put simply, if you do not require the range of numbers and precision offered by FP32, FP16 is a viable option which offers faster transfer operations through memory bandwidth and faster mathematical calculations due to the reduced precision (particularly on GPUs) (NVIDIA., 2023). The concept of mixed precision training, first put forward by Micikevicius et al. (2018), applies this idea by storing as many variables as prudent in half-point (FP16) format for the calculations and converting them back to FP32 for the optimization step (HuggingFace, nd). Mixed-precision training yields significant computational speed-ups, especially if using newer versions of NVIDIA GPUs (NVIDIA., 2023).

While rarely explicitly mentioned, FP32 is seemingly the default precision used also in the space of language models, however mixed precision training is explicitly utilized for RoBERTa (Liu et al., 2019). Initial testing for this thesis saw little to no difference in accuracy between FP32 and mixed precision training, and follow up tests performed on the fine-tuned models showed similar results (see Appendix A.1). As such, the precision options available during the hyperparameter search were limited to mixed precision training, using either FP16 or BF16. B-Float 16 or BF16 also utilizes 16-bits but is not as precise as FP16, instead using those bits to provide a much larger dynamic range and can therefore be valuable for avoiding overflow problems which can plague FP16.

Issues between hardware and software versions caused Tensor Float 32-bit (TF32), also known as "the magical data type" (HuggingFace, nd) due to its tendency to speed up operations, to be unavailable for the computation. TF32 has been shown to increase throughput in several contexts compared to FP32 and can be used in conjunction with FP16 or BF16 for the computation of the optimization step.

Other Hyperparameters

We train using the default HuggingFace optimizer AdamW, a version of the adaptive gradient method Adam (Kingma and Ba, 2017) where the weight decay has been decoupled from the gradient updates (Loshchilov and Hutter, 2019). In short, L2 regularization adapts the sums of the loss function's gradient and the regularizer's gradient, while decoupled weight decay only adapts the loss function's gradients, keeping the weight decay step separate. Loshchilov and Hutter (2019) showed that implementing weight decay with Adam instead of L2 regularization significantly improved its generalisation performance.

Additionally, a linear learning rate scheduler with a warm-up period was used to gradually reduce the learning rate during training, helping the optimization process to converge more effectively, a common practice (Goodfellow et al., 2016). The linear scheduler is a common choice (Vaswani et al., 2017; Devlin et al., 2019; Liu et al., 2019; Goodfellow et al., 2016). The warm-up ratio was set to 5%, meaning for the first 5% of estimated steps the learning rate increases from zero to the maximum learning rate (chosen by the random hyperparameter choice) to then decrease linearly. The warm-up ratio is a hyperparameter and appropriate values vary depending on the task and dataset, although rules of thumb exist (Ma and Yarats, 2021). Rates in previous studies have varied; Vaswani et al. (2017) used a warmup ratio of 4% while Liu et al. (2019) had a warm-up ratio of approximately 6%.

Given additional computational resources, these hyperparameters could have been fine-tuned as well instead of being fixed for the random search procedure.

4.3.2 Early Stopping

When fine-tuning the models during the robust hyperparameter random search, we also employed early stopping, meaning the fine-tuning process was stopped if the model did not show improvement in a certain metric during a limited training period. Early stopping has been shown to be equivalent to L2 regularization, but as described by Loshchilov and Hutter (2019), decoupled weight decay is not the same as L2 regularization and combinations of weight decay and early stopping are not uncommon (see, for example, Liu et al., 2019).

The metric chosen to base the early stopping procedure was the loss from the evaluation dataset. The loss function used is Cross Entropy-loss, which measures the difference between the predicted probability and the true label. As such, if the model predicts a high probability (higher confidence) for the correct label the loss will be smaller, while lower probability in correct predictions will yield larger loss values. The reverse is true when the model produces incorrect predictions. Put differently, a lower cross-entropy loss value implies that the model's predicted probabilities are

closely aligned with the true class labels, indicating greater confidence in the models predictions.

The patience of the early stopping procedure, meaning how many iterations can pass with no improvement in evaluation loss before the training is stopped, was set with evaluation steps to be equal to 1.5 epochs. Thus, if the model did not improve for 1.5 epochs the training was interrupted, the best model achieved during the training in terms of evaluation loss was retrieved and recorded. The patience was chosen to provide the model with enough leeway to bounce back based on the relatively low number of epochs expected, while also not increasing computational time unnecessarily.

4.3.3 Sequence Length

The sequence limit for the standard BERT, DistilBERT and RoBERTa models is 512 tokens. This does not, however, mean that the maximum length of a submitted sequence is 512. Due to [CLS] and [SEP] tokens, as well as special tokens for certain word forms, the actual word limit is lower and dependent on the input in question. Utilizing 512 as the upper limit however, we can get a feeling of the number of our observations which will be truncated, meaning cut-off to fit the 512 maximum token sequence.

In figure 4.1 below we can see that while a majority of the inputs contain less than 512 words, a few (roughly 5%) contain more than 512 words and are guaranteed to be truncated.

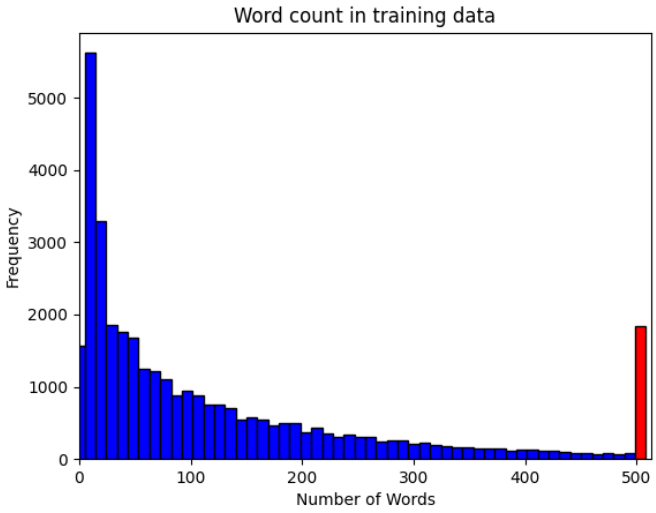


Figure 4.1: This histogram displays the word count distribution of the training data set. The x-axis represents the number of words, and the y-axis shows frequency. The last red bar groups records with over 512 words and thus the number of sequences that are guaranteed to be truncated.

Additionally, due to PHS-BERT being based on large BERT it takes up more memory on the GPU and the maximum token length was reduced to 256 in order to enable any training to be possible. The impact of this in terms of inputs truncated is visualized in figure 4.2 below, where we can see that roughly 83.5% of the obser-

vations contain 256 or fewer words, while roughly 95.0% of the observations contain less than 512 words.

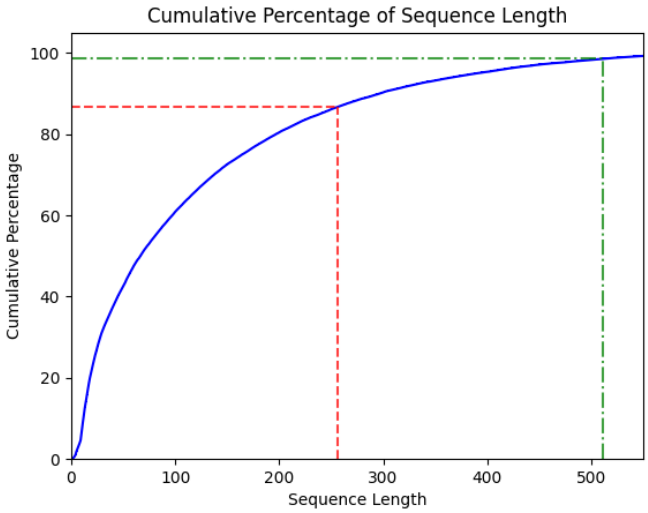


Figure 4.2: The cumulative plot illustrates the percentage of inputs with a word count below or equal to a value on the x-axis. Horizontal and vertical lines intersect the plot at $x = 256$ and $x = 512$ to illustrate the percentage of inputs unaffected by a maximum input length of 256 or 512 respectively.

4.4 Datasets

Below the datasets used for training and testing the best performing models are described.

4.4.1 Training Datasets

For the fine-tuning of the models during the random search, a combination of several of the datasets presented in section 2 were used. This process begins by undersampling the Twitter Suicidal Intention dataset and SWMH dataset to address the issue of class imbalance, as the non-SI class is overrepresented in both datasets. Such an imbalance could lead to biased performance, as the model may overfit to the majority class and struggle to generalize well for the minority class. Undersampling can help ensure that the models assign equal importance to both classes during training and learns meaningful representations of both classes, thereby improving generalization performance (Dal Pozzolo et al., 2015).

Undersampling works by randomly selecting a subset of majority class samples, making the number of observations of this class more comparable to the number of minority class observations (Dal Pozzolo et al., 2015). It is important to note that undersampling is not without it's problems, as it may lead to the loss of valuable information from the majority class, as some samples are removed from the dataset. Furthermore it means that the distribution of training and test/production datasets might not be the same, called a "distribution shift", which could result in discrepancies between the performance in training and production (Massachusetts Institue of Technology, nd).

By undersampling we were left with a balanced SWMH dataset containing 10k SI and non-SI labels respectively, and a Twitter Suicidal Intention dataset consisting of a total of 7,996 tweets, half of which (3,998) are labeled as SI. A subset of the already balanced Reddit Suicide and Depression Detection dataset was then selected, 8k of each label. We then combined the three balanced datasets into one balanced data set consisting of a total of 43,996 observations. Ideally the IMHR dataset would also have been utilized here in order to get a better balance between Reddit and Twitter observations, however access to this dataset was not yet functional at the commencement of the random search. The reasoning behind the greater number of observations selected from the SWMH dataset than from the Reddit Suicide and Depression Detection dataset is that SWMH was the product of a published research paper and was assumed to be of slightly higher quality. At this point the evaluation of the datasets by Dr. Cheri McDonald and the authors had not yet been completed, however, after evaluations were made the assumptions were shown to very likely be true.

After creation of this combined dataset, 20% of its (8,799) observations were set aside as testing data to be used once the random hyperparameter search was completed and the best combinations for each model selected. The remaining 35,197 observations were used for the random hyperparameter search, split 70% for training and 15% each for validation and testing.

4.4.2 Testing Datasets

Apart from the testing dataset set aside using the same distribution as the training dataset, we utilize two other datasets to test the best performing models in a more generalized social media setting.

The first of the generalized datasets is comprised of 182 positive SI cases from the SWMH dataset that were not part of the original training or testing datasets, and is combined with a random subset of 10,000 observations from 50 randomized Reddit communities sourced from the RSPCT Dataset (Jones, 2018). This creates a more imbalanced dataset with a wider variety of sequences, simulating a more realistic scenario for implementation on Reddit although the true ratio of SI to non-SI is unknown and would depend on deployment details. Going forward, this dataset is referred to as the Normal Reddit dataset.

Additionally, a subset of the IMHR Twitter dataset from Roy et al. (2020) consisting of 1,481 SI observations and 100,000 non-SI observations is used to test the models' performance on a more realistic Twitter scenario. Going forward this dataset will be referred to as the IMHR Twitter dataset.

4.5 Hardware and Time Consumption

The random search, fine-tuning and testing of the model were primarily carried out on an NVIDIA 3060TI graphics card with 8GB of memory. Additionally, the unit used for the computations had 16GB of RAM. Total time required for the random search was approximately six days.

5

Empirical Analysis

This section will present the results of the robust hyperparameter random search during the fine-tune training of the pre-trained transformer models. Each random combination for each model was evaluated at 3 random seeds, across a total of 40 models (8 times total for each BERT model variant). In addition to this, the best hyperparameter combination found for each model during the random search will be presented, as well as their performance during the random search against a validation dataset and against the testing datasets.

5.1 Results

After performing the robust hyperparameter random search, the best performing hyperparameter combination for each model is presented in table 5.1. In addition to this, the table also includes the number of epochs it takes for the given model to reach convergence, i.e. the lowest evaluation loss value, which is evaluated every 0.5 epochs and called back if there are 3 evaluation steps of non-decreasing loss. For a full overview of all combinations tested during the random search, the seeds used, and their results, please see the figures and tables in Appendix B.

The most apparent result from the hyperparameter search, according to table 5.1, is that a dropout rate of 0.1 performs the best for all models. Furthermore, it seems most models with an appropriate tune converge after 2-4 epochs, with 4 out of 5 of the models converging after 2 epochs of fine-tuning. Effective batch size does not seem to play an important role in determining model performance in this specific context, as all three configurations [16, 32, 128] make an appearance. A learning rate of 1e-05 and a weight decay of 0.1 make up the top 3 out of 5 hyperparameter choices found, with a learning rate of 5e-05 and a weight decay of 0.001 making up the other 2 of the 5 hyperparameters chosen. Most models, 4 out of 5, chose to use the half/mixed-precision approach offered by FP16.

The best performances for each model during the robust hyperparameter random search are described in table 5.2. The best performing model, MentalRoBERTa, has a validation accuracy of 91.4% and F1-score of 91.6% while the worst performing model, DistilBERT, has a validation accuracy of 89.1% and F1-score of 89.3%, a difference of only 2.3 percentage points for both metrics. The average training run-

Table 5.1: Best performing hyperparameter combination for each model from the robust random search trials.

Model	Learning Rate	Weight Decay	E. Batch size	Dropout Rate	bf16/ fp16	Epochs at Convergence
MentalRoBERTa	1e-05	0.1	128	0.1	bf16	4.0
MentalBERT	1e-05	0.1	16	0.1	fp16	2.0
PHS-BERT	1e-05	0.1	32	0.1	fp16	2.0
PsychBERT	5e-05	0.001	128	0.1	fp16	2.0
DistilBERT	5e-05	0.001	32	0.1	fp16	2.0

Table 5.2: Performance metrics using the best performing hyperparameter combinations for each model found during the robust random search trials against the validation dataset.

Model	Avg. Accuracy	Std. Accuracy	Avg. F1	Std. F1	Avg. Recall	Std. Recall	Avg. Precision	Std. Precision	Avg. Training Runtime (s)
MentalRoBERTa	0.9144	0.0003	0.9161	0.0007	0.9254	0.0118	0.9073	0.0098	3443.6
MentalBERT	0.9045	0.0027	0.9039	0.0029	0.8986	0.0099	0.9093	0.008	2225.7
PHS-BERT	0.8999	0.0041	0.8994	0.0034	0.8951	0.0036	0.9038	0.0104	5413.7
PsychBERT	0.8958	0.0006	0.8975	0.0008	0.9134	0.0025	0.8821	0.0009	2133.7
DistilBERT	0.8913	0.0032	0.8927	0.0046	0.8959	0.0211	0.8902	0.0131	1171.2

time, including the 1.5 epochs where early callback ended the training was, however, much smaller for DistilBERT, taking an average of 1171.2 seconds versus MentalRoBERTa's average of 3443.6 seconds, nearly 3 times as long as DistilBERT. Overall, it seems that Ji et al. (2022)'s pre-training approach to its "Mental" models is more applicable to this social media based task, performing best and second best in terms of accuracy and F1-score, with the larger, optimized variant MentalRoBERTa outperforming the base size MentalBERT. Naseem et al. (2022)'s PHS-BERT does, however, perform better than either Vajre et al. (2021)'s PsychBERT or Sanh et al. (2020)'s DistilBERT, which might be due to it being developed for use in social media compared to PsychBERT's split pre-training on academic and social media text. DistilBERT was the only model that had not been pre-trained on social media text, which combined with the fact it is the most lightweight model offers compelling arguments as to why it performs the worst out of all 5 of the models tested.

The performance metrics and results from testing the trained models with the best performing hyperparameters against the testing data with the same distribution as the training data is described table 5.3. The table shows no change in the order of the model performances in terms of accuracy and F1-score, with the best to worst models being MentalRoBERTa, MentalBERT, PHS-BERT, PsychBERT, and DistilBERT. The metrics are slightly lower as compared to the validation dataset, with MentalRoBERTa having the highest accuracy of 90.7% and F1-score of 90.9% while the DistilBERT having the lowest accuracy of 87.9% and F1-score of 88.2%, a difference in performance of only 2.8 and 2.7 percentage points respectively. F1 and precision scores slightly decreased across all models, while recall remains fairly high.

Table 5.3: Performance metrics for each model against the test dataset consisting of the same distribution as the training data after fine-tuning each model using the best found hyperparameters during the random search.

Model	Accuracy	F1	Recall	Precision	AP
MentalRoBERTa	0.9072	0.9085	0.9214	0.8959	0.9717
MentalBERT	0.8958	0.8975	0.9125	0.883	0.9593
PHS-BERT	0.8934	0.8962	0.92	0.8735	0.9623
PsychBERT	0.8927	0.8943	0.9073	0.8816	0.9586
DistilBERT	0.8792	0.8822	0.9048	0.8608	0.9508

This indicates that the models still correctly identify most (90%+) of the SI cases, but with slightly more false positives. Looking at the balance between precision and recall offered by the AP-score metric, we can see that these values are fairly high, indicating that the models are still fairly good at distinguishing between posts with and without expressions of SI.

The performance of the fine-tuned models against the test dataset (with the same distribution as the training data) can be compared to their performance without any fine-tuning, as shown in table 5.4 below. Without fine-tuning, we can see that several models seem to stick to one prediction for almost all inputs (e.g. MentalRoBERTa predicting all inputs as SI), and that most models are performing no better than a coinflip, having an accuracy close to 0.5. Results for the non-fine-tuned models on the other testing datasets are similarly worse and can be seen in Appendix C.

Table 5.4: Performance metrics for each model against the test dataset consisting of the same distribution as the training data ***before*** *fine-tuning.*

Model	Accuracy	F1	Recall	Precision
PHS-BERT	0.5665	0.4966	0.4277	0.5920
PsychBERT	0.5157	0.6715	0.9902	0.5080
MentalBERT	0.5129	0.6420	0.8734	0.5075
MentalRoBERTa	0.5	0.6667	1	0.5
DistilBERT	0.4109	0.4513	0.4845	0.4223

Table 5.5: Performance metrics for each model against the Normal Reddit testing dataset which has rates of suicidal ideation closer to the actual population of Reddit posts. Performances are after fine-tuning each model using the best found hyperparameters and primarily on text about or closely related to suicidal ideation.

Model	Accuracy	F1	Recall	Precision	AP
MentalBERT	0.9913	0.7861	0.8681	0.7182	0.8482
PHS-BERT	0.9824	0.6575	0.9176	0.5123	0.7079
PsychBERT	0.9564	0.4152	0.8407	0.2757	0.4481
DistilBERT	0.9524	0.399	0.8571	0.26	0.4344
MentalRoBERTa	0.946	0.3673	0.8516	0.2341	0.42

When the fine-tuned models were tested against the Normal Reddit dataset (discussed in 4.4.2) without additional training, all models' F1 and AP scores were reduced, sometimes drastically as can be seen in table 5.5. The new dataset simulates a rate of suicidal ideation, while still fairly high, closer in line with the proportion expressed in reality. The best performing model, based on accuracy, F1 and AP score becomes MentalBERT while the worst performing model becomes MentalRoBERTa. This shift could indicate that MentalRoBERTa was over fit, even being outperformed by DistilBERT. Recall and precision values also show interesting shifts. PHS-BERT outperforms all other models in terms of recall with almost 92%, practically maintaining its previous level, meaning it captures most of the true SI cases. While it's F1 score is weighed down by a higher number of false positives, roughly 5 out of every 10 SI predictions, depending on the context this conservative approach of prioritizing detection of true SI cases over increased numbers of false positives might very well be preferred. This reasoning can also be applied to the other models, as recall remained relatively high while precision dropped indicating that they tend to be overinclusive and assign an SI label to many non-SI posts. AP-scores remain relatively high for MentalBERT and PHS-BERT, indicating that these models still strike an acceptable balance between recall and precision, while dropping significantly for the remaining models as precision plummets and false positives increase.

Based on these results, inclusion of more general text data when fine-tuning the models might lead to improved model performance in these types of generalized settings. For further insight into the performance on the models, ROC and PRC curves for each model and for all three test datasets can be found in Appendix D.

Table 5.6: Performance metrics for each model against IMHR Twitter testing dataset. Performances are after fine-tuning each model using the best found hyperparameters and primarily on text about or closely related to suicidal ideation.

Model	Accuracy	F1	Recall	Precision	AP
MentalBERT	0.9911	0.7335	0.8231	0.6614	0.7928
PHS-BERT	0.9817	0.5828	0.8589	0.4411	0.6467
MentalRoBERTa	0.9821	0.5755	0.8170	0.4442	0.6564
DistilBERT	0.9798	0.5253	0.7522	0.4036	0.4883
PsychBERT	0.9244	0.2460	0.8298	0.1444	0.1425

The fine-tuned models were also tested against the IMHR Twitter dataset (discussed in 4.4.2) with similar outcome, as seen in table 5.6. While MentalBERT, PHS-BERT and PsychBERT showed a decrease in performance according to F1 and AP metrics, MentalRoBERTa and DistilBERT performed better on this Twitter dataset than the Normal Reddit dataset. MentalBERT and PHS-BERT remained the two best performing models however, MentalBERT maintains a fairly good balance between recall and precision resulting in an AP score of 0.79 and PHS-BERT showing the best recall metric at 85.9% of true SI cases correctly identified. Similar to the performance on the Normal Reddit dataset, recall scores remain relatively high, although the scores were lower than for the Normal Reddit and original test dataset.

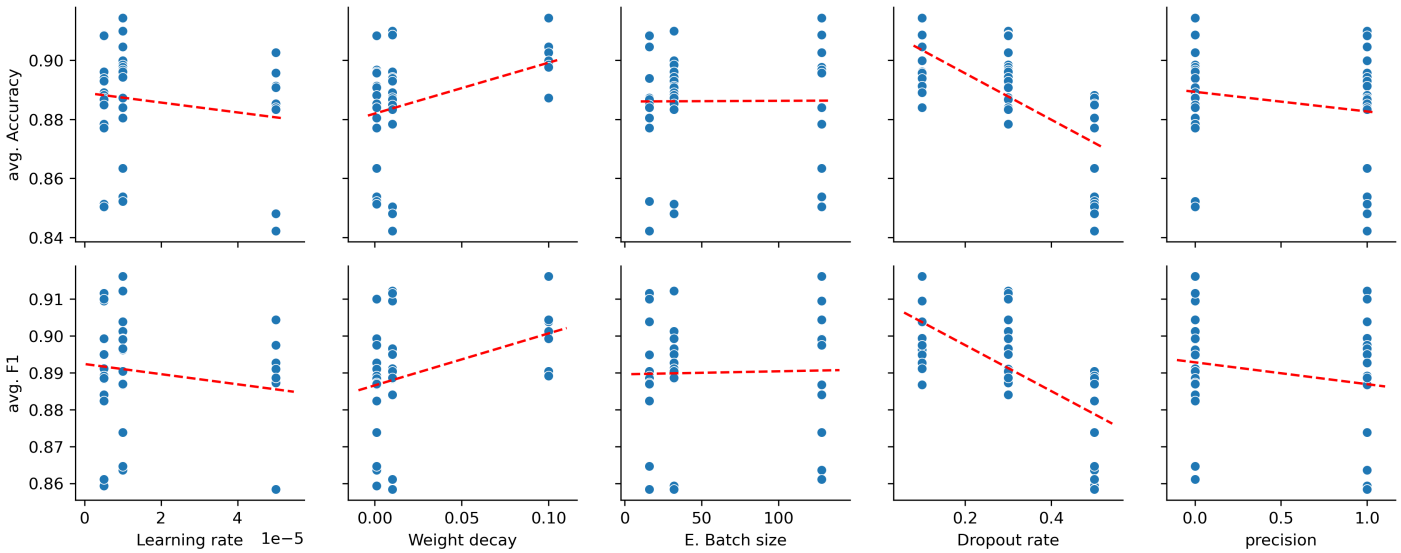


Figure 5.1: Paired plot overview analyzing the impact of hyperparameters on accuracy and the F1-scores of the models. The above plot combines the trials from all models during the robust hyperparameter random search. The red dashed line is a linear trendline.

The lower precision and the tendency to over label cases as SI again suggests the need to include more general social media posts in the training data, if the use case is deployment in a generalized social media context. Another potential factor here is the fact that twitter-posts make up less than half of the training data observations, and a more balanced training dataset might produce better results.

It's also important to note that during our study we did not account for colloquial expressions sometimes used in social media posts. Phrases such as "I want to die" or "kill me" can frequently occur in social media discourse, emanating not from SI, but rather from feelings such as embarrassment, exasperation, or as hyperbolic self-expression. As such expressions were not considered during training, these colloquialisms may impact the performance of our models in accurately identifying genuine instances of SI when tested on a more generalized dataset.

Examining all the models run during the hyperparameter random search, we can examine the general impacts of hyperparameter ranges on accuracy and F1 score. Looking at figure 5.1, we can see the effects of the different hyperparameters on the evaluation metrics. From the figure, we can see that there is a positive relationship between weight decay and the evaluation metrics of accuracy and F1 score. This demonstrates that the model could potentially benefit from increasing weight decay, however it does seem during the random search a weight decay of 0.1 was selected less frequently than either 0.001 or 0.01 so more trials are suggested. We can also see that there is a negative relationship between dropout rate and the evaluation metrics. As such, it might be of interest to explore dropout rates below 0.1. Learning rate, effective batch size, and precision (where 1 represents FP16 and 0 represents BF16) show weak or no relations to the evaluation metrics and more trials with wider search spaces are suggested to further study their effect in this particular context. For additional figures exploring the relationship between hyperparameter selections and metric results, see the parallel coordinate plots in Appendix E.

As computation time is also important in terms of hardware costs and in terms of

usability at scale, training computation times were recorded as a measure of model computation speed. Figure 5.2 depicts the average and standard deviation for the total time taken during the fine-tuning of each model. We can see that DistilBERT is by far the fastest model and has the smallest range in deviation from the norm. This is due to DistilBERT having the smallest architecture in terms of the number of parameters. MentalBERT, PsychBERT, and MentalRoBERTa all perform roughly the same in terms of fine-tuning time taken, as well as their standard deviations. PHS-BERT has the largest standard deviation and takes the longest to train, which could be due to being built on the large BERT model, which was not optimized like RoBERTa was.

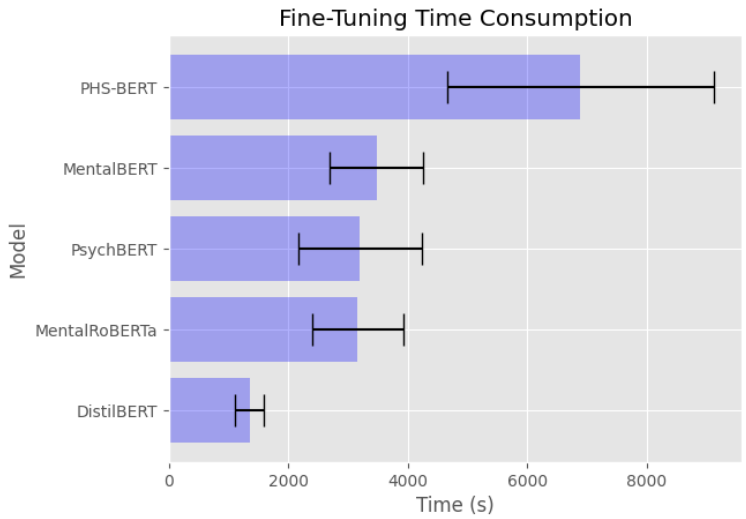


Figure 5.2: Average training runtime during model training/fine-tuning for each model. The errorbars represent one standard deviation from the average. Note that this is total training time and is thus affected by the number of epochs required until convergence.

Due to the time constraints of our study, we did not fine-tune the models further with the additional Normal Reddit and IMHR Twitter datasets, but leave this aspect open for further research.

6

Conclusion

6.1 Summary of Findings

The aim of this study was to develop the existing research into a robust transformer model that is capable of detecting suicidal ideation in text based social media posts, with the hope of being able to create an early detection system. Through our research, we were able to robustly fine-tune and evaluate pre-trained transformer models that currently exist in the mental health space. The models were trained on social media posts from Reddit and Twitter and went through a robust hyper-parameter random search where each combination was tried against three random seeds, where then the best performing combination for each model went to a final series of tests.

Even with noise in labels of the fine-tune training data, the models were able to achieve a high level of performance in the classification of suicidal ideation. Against data that had the same balanced distribution used in the fine-tune training, the best performing model had a test accuracy and F1-score of approximately 0.91 while the lowest performing, yet most lightweight, had a test accuracy and F1-score of approximately 0.88.

Placing the models into an more realistic scenarios, the models had a substantial decrease in their F1-score ranging from the best model at approximately 0.79 to the worst model at approximately 0.25, suggesting a need to include general social media posts in the fine-tune training data if these models are to be implemented in a generalized setting. With that said, however, many models still retained recall scores in the 0.8-0.9 range, meaning relatively few true SI cases were missed. While both the recall and especially the precision could be improved, a more conservative approach with a higher rate of false positives might be preferable in early detection systems as it could be indicative and helpful in detecting borderline SI cases.

6.2 Practical Implications

The results of this study have a number of practical implications that could potentially influence the way social media organizations and mental health professionals approach the task of identifying suicidal ideation in social media posts.

The study demonstrates that transformer models are effective at detecting suicidal ideation from social media posts. Following from this, these type of models can possibly be introduced into current social media platforms or be built into mental health monitoring tools to look for expressions of suicidal ideation, increasing the ability to provide faster intervention which could lead to more lives saved. It is important to note for policy makers and organizations that care must be taken to ensure implementation methods are ethical and respect users' privacy.

This study also demonstrates the trade-offs that exist between computation time and performance, which needs to be thought about when being put into applications. For example, in real-time monitoring cases such as social media platforms, DistilBERT may be more useful because of its computation speed, but in clinical settings a more accurate model variant such as BERT or RoBERTa may be more applicable.

6.3 Research Limitations & Future Research

While this study has made advancements in the use of pre-trained transformers to identify suicidal ideation in social media, it is not without its limitations. The text data used was sourced from specific social media platforms, which might not fully capture the complexity and variability of language used in different contexts and expressions of SI. Furthermore, access to quality and professionally annotated data is scarce, and while subsets of some of the datasets used were professionally evaluated, utilization of larger and higher quality datasets could lead to improved model performance by covering a wider range of language and semantics surrounding suicidal ideation and reducing incorrectly labeled observations. Collecting and utilizing more diverse, representative, and robustly labeled datasets is therefore something we see as a key area of future research.

The study was further limited by the constraint of our computational resources, resulting in relatively small search spaces for the hyperparameters and limiting the number of models that were included. As it's likely that there are hyperparameter combinations of interest that exist outside of the space created, future research could expand the search space for both hyperparameters and models, allowing more combinations to be robustly explored.

Finally, future research could investigate other model architectures or variations of transformers. Specifically, future studies could focus on the trade-off between computational efficiency and performance discussed above. Reducing the computational resources required for both fine-tuning and pre-training would make deployment more broadly feasible. One such area to explore, for example, is utilizing smaller transformer models like DistilBERT. While it performed the worst out of the models fine-tuned here it was not far behind, and in some testing scenarios performed better than larger models; and if it was further pre-trained, DistilBERT might be able to perform similarly to the others with a higher computational efficiency.

Bibliography

Awata, S., Bech, P., Koizumi, Y., Seki, T., Kuriyama, S., Hozawa, A., Ohmori, K., Nakaya, N., Matsuoka, H., Tsuji, I., and et al. (2007). Validity and utility of the japanese version of the who-five well-being index in the context of detecting suicidal ideation in elderly community residents. *International Psychogeriatrics*, 19(1):77–88. Available Online: https://doi.org/10.1017/S1041610206004212 [Accessed: 2023-05-19].

Ayers, J. W., Poliak, A., Dredze, M., Leas, E. C., Zhu, Z., Kelley, J. B., Faix, D. J., Goodman, A. M., Longhurst, C. A., Hogarth, M., and Smith, D. M. (2023). Comparing Physician and Artificial Intelligence Chatbot Responses to Patient Questions Posted to a Public Social Media Forum. *JAMA Internal Medicine*. Available Online: https://doi.org/10.1001/jamainternmed.2023.1838 [Accessed: 2023-05-19].

Ba, J. L., Kiros, J. R., and Hinton, G. E. (2016). Layer normalization. Available Online: https://doi.org/10.48550/arXiv.1607.06450 [Accessed: 2023-05-19].

Balani, S. and De Choudhury, M. (2015). Detecting and characterizing mental health related self-disclosure in social media. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '15, page 1373–1378, New York, NY, USA. Association for Computing Machinery. Available Online: https://doi.org/10.1145/2702613.2732733 [Accessed: 2023-05-19].

Belfor, E. L., Mezzacappa, E., and Ginnis, K. (2012). Similarities and differences among adolescents who communicate suicidality to others via electronic versus other means: a pilot study. *Adolescent Psychiatry*, (2):258–262. Available Online: https://doi.org/10.2174/2210676611202030258 [Accessed: 2023-05-19].

Benton, A., Coppersmith, G., and Dredze, M. (2017). Ethical research protocols for social media health research. In *Proceedings of the First ACL Workshop on Ethics in Natural Language Processing*, pages 94–102, Valencia, Spain. Association for Computational Linguistics. Available Online: https://doi.org/10.18653/v1/W17-1612 [Accessed: 2023-05-19].

Bergstra, J. and Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(10):281–305. Available Online: http://jmlr.org/papers/v13/bergstra12a.html [Accessed: 2023-05-19].

Borges, G., Angst, J., Nock, M., Ruscio, A., and Kessler, R. (2008). Risk factors for the incidence and persistence of suicide-related outcomes: A 10-year

follow-up study using the national comorbidity surveys. *Journal of affective disorders*, 105:25–33. Available Online: https://doi.org/10.1016/j.jad.2007.01.036 [Accessed: 2023-05-19].

Bruce, M. L., Ten Have, T. R., Reynolds, C. F., 3rd, Katz, I. I., Schulberg, H. C., Mulsant, B. H., Brown, G. K., McAvay, G. J., Pearson, J. L., and Alexopoulos, G. S. (2004). Reducing suicidal ideation and depressive symptoms in depressed older primary care patients: a randomized controlled trial. *Journal of the American Medical Association (JAMA)*, (291):1081–1091. Available Online: https://doi.org/10.1001/jama.291.9.1081 [Accessed: 2023-05-19].

Bucilu, C., Caruana, R., and Niculescu-Mizil, A. (2006). Model compression. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '06, page 535–541, New York, NY, USA. Association for Computing Machinery. Available Online: https://doi.org/10.1145/1150402.1150464 [Accessed: 2023-05-19].

Cao, L., Zhang, H., Feng, L., Wei, Z., Wang, X., Li, N., and He, X. (2019). Latent suicide risk detection on microblog via suicide-oriented word embeddings and layered attention. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1718–1728, Hong Kong, China. Association for Computational Linguistics. Available Online: https://doi.org/10.18653/v1/D19-1181 [Accessed: 2023-05-19].

Casola, S., Lauriola, I., and Lavelli, A. (2022). Pre-trained transformers: an empirical comparison. *Machine Learning with Applications*, 9:100334. Available Online: https://doi.org/10.1016/j.mlwa.2022.100334 [Accessed: 2023-05-19].

Dal Pozzolo, A., Caelen, O., and Bontempi, G. (2015). When is undersampling effective in unbalanced classification tasks? In Appice, A., Rodrigues, P. P., Santos Costa, V., Soares, C., Gama, J., and Jorge, A., editors, *Machine Learning and Knowledge Discovery in Databases*, pages 200–215, Cham. Springer International Publishing. Available online: https://doi.org/10.1007/978-3-319-23528-8_13.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). 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. Association for Computational Linguistics. Available Online: https://doi.org/10.18653/v1/N19-1423 [Accessed: 2023-05-19].

dontloo. (2019). What exactly are keys, queries, and values in attention mechanisms? Available Online: https://stats.stackexchange.com/questions/421935/what-exactly-are-keys-queries-and-values-in-attention-mechanisms [Accessed: 2023-05-19].

Feurer, M. and Hutter, F. (2019). *Hyperparameter Optimization*, pages 3–33. Springer International Publishing, Cham. Available Online: https://doi.org/10.1007/978-3-030-05318-5_1 [Accessed: 2023-05-19].

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. Available Online: http://www.deeplearningbook.org [Accessed: 2023-05-19].

Greco, C. M., Simeri, A., Tagarelli, A., and Zumpano, E. (2023). Transformer-based language models for mental health issues: A survey. *Pattern Recognition Letters*, 167:204–211. Available Online: https://doi.org/10.1016/j.patrec.2023.02.016 [Accessed: 2023-05-19].

Haque, F., Nur, R. U., Jahan, S. A., Mahmud, Z., and Shah, F. M. (2020). A transformer based approach to detect suicidal ideation using pre-trained language models. In *2020 23rd International Conference on Computer and Information Technology (ICCIT)*, pages 1–5. Available Online: https://doi.org/10.1109/ICCIT51783.2020.9392692 [Accessed: 2023-05-19].

Harmer, B., Lee, S., Duong, T. v. H., and Saadabadi, A. (2023). *Suicidal Ideation*. StatPearls Publishing. Available Online: https://www.ncbi.nlm.nih.gov/books/NBK565877/ [Accessed: 2023-05-19].

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Deep residual learning for image recognition. Available Online: https://doi.org/10.48550/arXiv.1512.03385 [Accessed: 2023-05-19].

Hinton, G., Vinyals, O., and Dean, J. (2015). Distilling the knowledge in a neural network. Available Online: https://doi.org/10.48550/arXiv.1503.02531 [Accessed: 2023-05-19].

Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., de Laroussilhe, Q., Gesmundo, A., Attariyan, M., and Gelly, S. (2019). Parameter-efficient transfer learning for nlp. Available Online: https://doi.org/10.48550/arXiv.1902.00751 [Accessed: 2023-05-19].

HuggingFace (2022). Performance and scalability: How to fit a bigger model and train it faster. Available Online: https://huggingface.co/docs/transformers/v4.18.0/en/performance#gradient-accumulation [Accessed: 2023-05-19].

HuggingFace (n.d.). Efficient training on a single gpu. Available Online: https://huggingface.co/docs/transformers/perf_train_gpu_one#floating-data-types [Accessed: 2023-05-19].

HuggingFace. (n.d.). Trainer. Available Online: https://huggingface.co/docs/transformers/main_classes/trainer [Accessed: 2023-05-19].

Ji, S., Li, X., Huang, Z., and Cambria, E. (2021). Suicidal ideation and mental disorder detection with attentive relation networks. *Neural Computing and Applications*. Available Online: https://doi.org/10.1007/s00521-021-06208-y [Accessed: 2023-05-19].

Ji, S., Zhang, T., Ansari, L., Fu, J., Tiwari, P., and Cambria, E. (2022). MentalBERT: Publicly available pretrained language models for mental healthcare. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*,

pages 7184–7190, Marseille, France. European Language Resources Association. Available Online: https://aclanthology.org/2022.lrec-1.778 [Accessed: 2023-05-19].

Joinson, A. and Paine, C. B. (2009). Self-disclosure, Privacy and the Internet. In *Oxford Handbook of Internet Psychology*. Oxford University Press. Available online: https://doi.org/10.1093/oxfordhb/9780199561803.013.0016 [Accessed 2023-05-20].

Jones, M. (2018). The reddit self-post classification task. Available Online: https://www.kaggle.com/datasets/mswarbrickjones/reddit-selfposts [Accessed: 2023-05-18].

Kant, L. (2020). Github repository: Laxmimerit/twitter-suicidal-intention-dataset. Available Online: https://github.com/laxmimerit/twitter-suicidal-intention-dataset [Accessed: 2023-05-19].

Kingma, D. P. and Ba, J. (2017). Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015)*, San Diego, USA. Available Online: https://doi.org/10.48550/arXiv.1412.6980 [Accessed: 2023-05-19].

Komati, N. (2021). Suicide and depression detection. Technical Report Version 14, Kaggle. Available Online: https://www.kaggle.com/datasets/nikhileswarkomati/suicide-watch [Accessed: 2023-05-19].

Kwasny, R., Friar, D., and Papallo, G. (2023). An imagenet-like text classification task based on reddit posts. Available Online: https://www.evolution.ai/post/an-imagenet-like-text-classification-task-based-on-reddit-posts [Accessed: 2023-05-18].

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., and Soricut, R. (2020). Albert: A lite bert for self-supervised learning of language representations. Available Online: https://doi.org/10.48550/arXiv.1909.11942 [Accessed: 2023-05-19].

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. Available Online: https://doi.org/10.48550/arXiv.1907.11692 [Accessed: 2023-05-19].

Loshchilov, I. and Hutter, F. (2019). Decoupled weight decay regularization. In *International Conference on Learning Representations*. Available Online: https://doi.org/10.48550/arXiv.1711.05101 [Accessed: 2023-05-19].

Ma, J. and Yarats, D. (2021). On the adequacy of untuned warmup for adaptive optimization. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(10):8828–8836. Available Online: https://doi.org/10.1609/aaai.v35i10.17069 [Accessed: 2023-05-19].

Massachusetts Institue of Technology (n.d). Class imbalance, outliers, and distribution shift. Available online: https://dcai.csail.mit.edu/lectures/imbalance-outliers-shift/#distribution-shift [Accessed 2023-05-19].

Micikevicius, P., Narang, S., Alben, J., Diamos, G., Elsen, E., Garcia, D., Ginsburg, B., Houston, M., Kuchaiev, O., Venkatesh, G., and Wu, H. (2018). Mixed precision training. In *International Conference on Learning Representations*. Available Online: https://doi.org/10.48550/arXiv.1710.03740 [Accessed: 2023-05-19].

Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., and Gao, J. (2021). Deep learning–based text classification: A comprehensive review. *ACM Comput. Surv.*, 54(3). Available Online: https://doi.org/10.1145/3439726 [Accessed: 2023-05-19].

Naseem, U., Lee, B. C., Khushi, M., Kim, J., and Dunn, A. G. (2022). Benchmarking for public health surveillance tasks on social media with a domain-specific pretrained language model. Available Online: https://doi.org/10.48550/arXiv.2204.04521 [Accessed: 2023-05-19].

Nock, M. K., Borges, G., Bromet, E. J., Alonso, J., Angermeyer, M., Beautrais, A., Bruffaerts, R., Chiu, W. T., de Girolamo, G., Gluzman, S., and et al. (2008). Cross-national prevalence and risk factors for suicidal ideation, plans and attempts. *The British Journal of Psychiatry*, 192(2):98–105. Available Online: https://doi.org/10.1192/bjp.bp.107.040113 [Accessed: 2023-05-19].

NVIDIA. (2023). Train with mixed precision. Available online: https://docs.nvidia.com/deeplearning/performance/mixed-precision-training/index.html [Accessed: 2023-05-18].

Park, M., Cha, C., and Cha, M. (2012). Depressive moods of users portrayed in twitter. In *Proceedings of the 18th ACM International Conference on Knowledge Discovery and Data Mining, SIGKDD 2012*, pages 1–8. Available Online: https://nyuscholars.nyu.edu/en/publications/depressive-moods-of-users-portrayed-in-twitter [Accessed: 2023-05-19].

Peters, M. E., Ammar, W., Bhagavatula, C., and Power, R. (2017). Semi-supervised sequence tagging with bidirectional language models. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1756–1765, Vancouver, Canada. Association for Computational Linguistics. Available Online: https://doi.org/10.18653/v1/P17-1161 [Accessed: 2023-05-19].

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics. Available Online: https://doi.org/10.18653/v1/N18-1202 [Accessed: 2023-05-19].

Pew Research Center (2021). Use of online platforms, apps varies – sometimes widely – by demographic group. Available Online: https://www.pewresearch.org/internet/2021/04/07/social-media-use-in-2021/pi_2021-04-07_social-media_0-03/ [Accessed: 2023-05-18].

RedditInc (n.d.). Homepage. Available Online: https://www.redditinc.com [Accessed: 2023-05-19].

Roy, A., Nikolitch, K., McGinn, R., Jinah, S., Klement, W., and Kaminsky, Z. A. (2020). A machine learning approach predicts future risk to suicidal ideation from social media data. *npj Digital Medicine*, 3(1). Available Online: https://doi.org/10.1038/s41746-020-0287-6 [Accessed: 2023-05-19].

Sanh, V., Debut, L., Chaumond, J., and Wolf, T. (2020). Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. Available Online: https://doi.org/10.48550/arXiv.1910.01108 [Accessed: 2023-05-19].

Scikit-Learn (n.d.). sklearn.metrics.average_precision_score. Available online: https://scikit-learn.org/stable/modules/generated/sklearn.metrics.average_precision_score [Accessed 2023-05-19].

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958. Available online: http://jmlr.org/papers/v15/srivastava14a.html [Accessed 2023-05-19].

Sun, C., Qiu, X., Xu, Y., and Huang, X. (2020). How to fine-tune bert for text classification? Available Online: https://doi.org/10.48550/arXiv.1905.05583 [Accessed: 2023-05-19].

Turner, A. (2023a). How many users does twitter have? Available Online: https://www.bankmycell.com/blog/how-many-users-does-twitter-have [Accessed: 2023-05-19].

Turner, A. (2023b). Reddit user statistics: How many people use reddit? Available Online: https://www.bankmycell.com/blog/number-of-reddit-users/ [Accessed: 2023-05-19].

Vajre, V., Naylor, M., Kamath, U., and Shehu, A. (2021). Psychbert: A mental health language model for social media mental health behavioral analysis. In *2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 1077–1082. Available Online: https://doi.org/10.1109/BIBM52615.2021.9669469 [Accessed: 2023-05-19].

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. (2017). Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc. Available Online: https://doi.org/10.48550/arXiv.1706.03762 [Accessed: 2023-05-19].

World Health Organization (2018). National suicide prevention strategies: Progress, examples and indicators. Available Online: https://apps.who.int/iris/handle/10665/279765 [Accessed: 2023-05-19].

World Health Organization (2021). Suicide. Available online: https://www.who.int/news-room/fact-sheets/detail/suicide [Accessed 2023-05-19].

Zhang, T., Schoene, A. M., Ji, S., and Ananiadou, S. (2022). Natural language processing applied to mental illness detection: a narrative review. Available Online: https://doi.org/10.1038/s41746-022-00589-7 [Accessed: 2023-05-19].

Zhu, Y., Kiros, R., Zemel, R., Salakhutdinov, R., Urtasun, R., Torralba, A., and Fidler, S. (2015). Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 19–27. Available Online: https://doi.org/10.1109/ICCV.2015.11 [Accessed: 2023-05-19].

Appendix A

Precision comparison

Table A.1: Difference in performance using FP32 precision vs FP16 precision for the final models (excluding PHS-BERT). Values are from FP32 performance minus FP16 performance

Model	Δ Accuracy	Δ F1	Δ Recall	Δ Precision
MentalRoBERTa	0.0031	0.0030	0.0027	0.0032
MentalBERT	-0.0016	-0.0009	0.0052	-0.0065
PsychBERT	0.0009	0.0029	0.0211	-0.0136
DistilBERT	0.0027	-0.0003	-0.0234	0.0216

Appendix B

Hyperparameter search results

Table B.1: Results from the robust random search of hyper parameters with accompanying metrics and model training time. The robust random search was conducted with 3 runs with random seeds for each configuration to establish mean and standard deviation values. NaN values in "Epochs at Convergence" represent the model not reaching convergence, using a maximum of 7 epochs. Sorted by accuracy.

Model	Learning Rate	Weight Decay	E. Batch Size	Dropout Rate	bf16/ fp16	Avg. Accuracy	Std. Accuracy	Avg. F1	Std. F1	Avg. Recall	Std. Recall	Avg. Precision	Std. Precision	Avg. Training Runtime	Epochs at Convergence
mental/mental-roberta-base	1e-05	0.1	128	0.1	bf16	0.9144	0.0003	0.9161	0.0007	0.9254	0.0118	0.9073	0.0098	3443.6	4.0
mental/mental-roberta-base	1e-05	0.01	32	0.3	fp16	0.91	0.0027	0.9122	0.0018	0.9362	0.0077	0.8896	0.0104	3096.3	3.5
mental/mental-roberta-base	5e-06	0.01	128	0.1	bf16	0.9086	0.0012	0.9095	0.0011	0.9188	0.0022	0.9003	0.0022	4079.6	5.0
mental/mental-roberta-base	5e-06	0.01	16	0.3	bf16	0.9086	0.002	0.9115	0.002	0.9322	0.004	0.8918	0.0027	3656.3	4.0
mental/mental-roberta-base	5e-06	0.001	16	0.3	fp16	0.9083	0.0008	0.91	0.0005	0.9279	0.0093	0.8929	0.0082	3437.6	4.0
mental/mental-bert-base-uncased	1e-05	0.1	16	0.1	fp16	0.9045	0.0027	0.9039	0.0029	0.8986	0.0099	0.9093	0.008	2225.7	2.0
mental/mental-bert-base-uncased	5e-05	0.1	128	0.3	bf16	0.9027	0.002	0.9044	0.0013	0.9216	0.0049	0.8878	0.007	2316.8	2.0
publichealthsurveillance/PHS-BERT	1e-05	0.1	32	0.1	fp16	0.8999	0.0041	0.8994	0.0034	0.8951	0.0036	0.9038	0.0104	5413.7	2.0
mental/mental-bert-base-uncased	1e-05	0.1	32	0.3	bf16	0.8984	0.0015	0.9013	0.0007	0.9175	0.0064	0.8857	0.0071	3248.9	4.0
mental/mental-bert-base-uncased	1e-05	0.1	128	0.3	bf16	0.8977	0.006	0.8993	0.0052	0.9137	0.002	0.8854	0.0113	3845.9	NaN
mental/mental-bert-base-uncased	1e-05	0.001	128	0.3	bf16	0.8968	0.002	0.8991	0.0019	0.92	0.0011	0.8791	0.0029	3980.6	NaN
mental/mental-bert-base-uncased	5e-06	0.001	32	0.3	bf16	0.8961	0.0016	0.8992	0.0014	0.9169	0.0011	0.8822	0.0032	3916.5	5.0
publichealthsurveillance/PHS-BERT	1e-05	0.01	32	0.1	bf16	0.8961	0.003	0.8963	0.0034	0.899	0.0144	0.8939	0.0111	4877.3	1.5
mnaylor/psychbert-cased	5e-05	0.001	128	0.1	fp16	0.8958	0.0006	0.8975	0.0008	0.9134	0.0025	0.8821	0.0009	2133.7	2.0
mnaylor/psychbert-cased	1e-05	0.01	32	0.3	fp16	0.8943	0.0016	0.8965	0.0009	0.9168	0.0077	0.8773	0.0078	3397.0	4.0
publichealthsurveillance/PHS-BERT	5e-06	0.01	16	0.1	bf16	0.8939	0.0007	0.8948	0.0009	0.9037	0.0125	0.8865	0.0103	5233.5	1.5
publichealthsurveillance/PHS-BERT	5e-06	0.01	32	0.3	fp16	0.893	0.0016	0.895	0.0011	0.9027	0.0045	0.8875	0.0059	6567.3	3.0
distilbert-base-uncased	5e-05	0.001	32	0.1	fp16	0.8913	0.0032	0.8927	0.0046	0.8959	0.0211	0.8902	0.0131	1171.2	2.0
distilbert-base-uncased	5e-05	0.001	32	0.3	bf16	0.8908	0.0025	0.891	0.0033	0.8945	0.0152	0.888	0.0109	1048.7	2.0
mnaylor/psychbert-cased	5e-06	0.01	32	0.1	bf16	0.8891	0.0019	0.8911	0.0011	0.8976	0.009	0.8849	0.0093	2345.1	2.0
distilbert-base-uncased	5e-06	0.001	32	0.5	fp16	0.8881	0.0016	0.8891	0.0025	0.8982	0.0098	0.8803	0.0048	1748.0	4.0
mnaylor/psychbert-cased	1e-05	0.01	16	0.3	bf16	0.8874	0.0052	0.8904	0.0048	0.9158	0.0043	0.8665	0.0067	2184.3	3.0
distilbert-base-uncased	1e-05	0.1	32	0.5	bf16	0.8872	0.0019	0.8904	0.001	0.9167	0.0083	0.8657	0.0086	1336.1	2.5
distilbert-base-uncased	5e-06	0.1	16	0.5	fp16	0.8872	0.0019	0.8891	0.0016	0.9052	0.0108	0.8738	0.0096	1383.6	3.0
distilbert-base-uncased	5e-06	0.01	16	0.3	fp16	0.8867	0.0026	0.8883	0.0028	0.9023	0.0185	0.8754	0.0144	1432.9	3.0
distilbert-base-uncased	5e-05	0.001	32	0.5	fp16	0.8854	0.0036	0.8884	0.0016	0.9032	0.0216	0.8751	0.0204	1048.7	2.0
publichealthsurveillance/PHS-BERT	5e-06	0.01	16	0.5	bf16	0.885	0.0044	0.8885	0.004	0.918	0.0042	0.861	0.0063	9633.4	5.0
mnaylor/psychbert-cased	5e-05	0.001	16	0.3	bf16	0.884	0.004	0.8873	0.0024	0.9134	0.0122	0.8629	0.0145	2228.0	2.0
distilbert-base-uncased	1e-05	0.001	128	0.1	fp16	0.884	0.0018	0.8868	0.0015	0.8993	0.0023	0.8746	0.004	1623.1	4.0
publichealthsurveillance/PHS-BERT	5e-05	0.01	32	0.3	fp16	0.8834	0.0079	0.8886	0.0068	0.9202	0.0038	0.8592	0.0117	4283.9	1.5
mental/mental-bert-base-uncased	1e-05	0.001	16	0.5	bf16	0.8805	0.0077	0.887	0.0058	0.9278	0.0073	0.8499	0.0152	3577.3	5.0
mnaylor/psychbert-cased	5e-06	0.01	128	0.3	bf16	0.8785	0.0018	0.884	0.0009	0.9164	0.0073	0.8539	0.0073	4650.2	NaN
publichealthsurveillance/PHS-BERT	5e-06	0.001	16	0.5	bf16	0.8771	0.0058	0.8824	0.0035	0.9221	0.0141	0.8464	0.0181	8417.2	4.0
mental/mental-bert-base-uncased	1e-05	0.001	128	0.5	fp16	0.8634	0.0066	0.8738	0.0045	0.9459	0.009	0.8121	0.0136	4659.0	NaN
mnaylor/psychbert-cased	1e-05	0.001	128	0.5	fp16	0.8537	0.0085	0.8636	0.0043	0.9157	0.0215	0.8183	0.0252	4325.6	NaN
mental/mental-roberta-base	1e-05	0.001	16	0.5	bf16	0.8522	0.0032	0.8646	0.0055	0.9347	0.0238	0.8048	0.0084	2442.5	3.0
mnaylor/psychbert-cased	5e-06	0.001	32	0.5	fp16	0.8513	0.0091	0.8593	0.0043	0.9082	0.0257	0.817	0.0288	4307.2	NaN
publichealthsurveillance/PHS-BERT	5e-06	0.01	128	0.5	bf16	0.8504	0.0066	0.8611	0.0037	0.917	0.0122	0.812	0.0158	10681.4	NaN
mental/mental-roberta-base	5e-05	0.01	32	0.5	fp16	0.848	0.0028	0.8584	0.0024	0.9224	0.0095	0.8029	0.0069	1516.1	1.0
mental/mental-roberta-base	5e-05	0.01	16	0.5	fp16	0.8422	0.0152	0.8583	0.0105	0.9558	0.015	0.7796	0.0237	3562.6	5.0

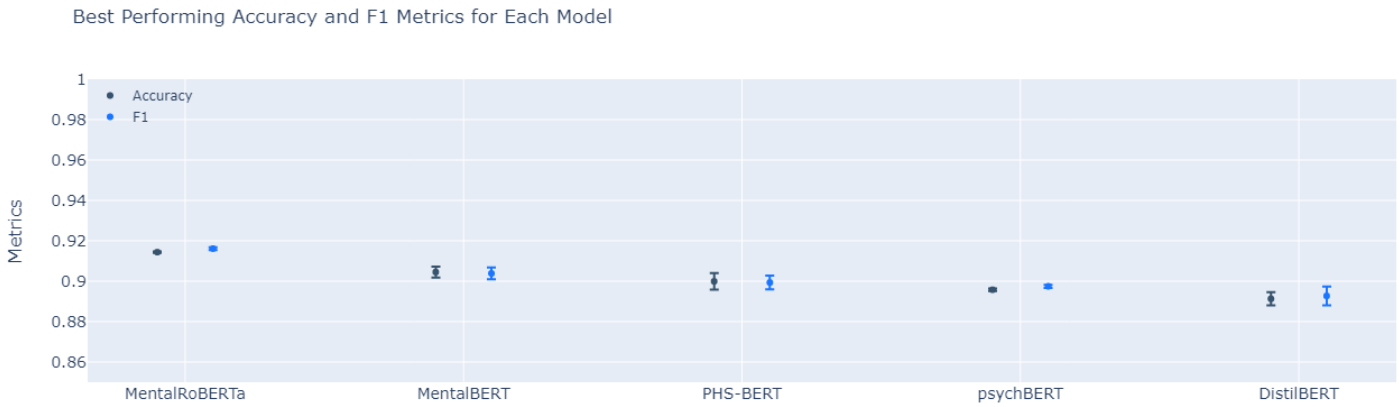


Figure B.1: Performance of models in terms of Accuracy and F1 with error bars included

Table B.2: Seeds utilized during the random search for each trial. Ordering is the same as in Tabel B.1 above.

Model	Seeds
mental/mental-roberta-base	[1106763746, 299029916, 1417749834]
mental/mental-roberta-base	[1255185856, 2970252694, 3921262637]
mental/mental-roberta-base	[2622675586, 519181682, 2752315950]
mental/mental-roberta-base	[1737358439, 3210977169, 2605503547]
mental/mental-roberta-base	[3588745318, 376064971, 1891258533]
mental/mental-bert-base-uncased	[3053375588, 1317930322, 3214932045]
mental/mental-bert-base-uncased	[3233306623, 3351088375, 1527662849]
publichealthsurveillance/PHS-BERT	[2324431265, 2876649110, 1555847211]
mental/mental-bert-base-uncased	[2739813511, 3890827836, 461376504]
mental/mental-bert-base-uncased	[1888996467, 195290664, 1146866943]
mental/mental-bert-base-uncased	[2840093105, 2299852178, 36673843]
mental/mental-bert-base-uncased	[2022802130, 3040685039, 2323165098]
publichealthsurveillance/PHS-BERT	[3874017394, 2910039224, 3994667487]
mnaylor/psychbert-cased	[1019745909, 3211200192, 1623538460]
mnaylor/psychbert-cased	[407076172, 4146655470, 604330228]
publichealthsurveillance/PHS-BERT	[2981685284, 2201290722, 1093677434]
publichealthsurveillance/PHS-BERT	[4103314878, 480048397, 1277830954]
distilbert-base-uncased	[4108280851, 3040279568, 1253580082]
distilbert-base-uncased	[2319314102, 147835935, 3858126191]
mnaylor/psychbert-cased	[4133111773, 1899621510, 1768381052]
distilbert-base-uncased	[1482956036, 2375570518, 3829412630]
mnaylor/psychbert-cased	[1600068697, 3163146025, 1613653034]
distilbert-base-uncased	[3133029574, 708903114, 3295217873]
distilbert-base-uncased	[4246465560, 2836992208, 1386759166]
distilbert-base-uncased	[3847934361, 381967677, 283826318]
distilbert-base-uncased	[908014777, 2394830440, 1084101292]
publichealthsurveillance/PHS-BERT	[2804882604, 1636867452, 898615959]
mnaylor/psychbert-cased	[865138490, 2489312644, 4132410272]
distilbert-base-uncased	[4028267788, 3793292022, 2785677833]
publichealthsurveillance/PHS-BERT	[4087653450, 813281531, 4145280712]
mental/mental-bert-base-uncased	[1465149761, 1402130911, 404401887]
mnaylor/psychbert-cased	[493895715, 2073071199, 2692848488]
publichealthsurveillance/PHS-BERT	[3939105037, 53792266, 2727208313]
mental/mental-bert-base-uncased	[1880377600, 2664175979, 2319429930]
mnaylor/psychbert-cased	[760699015, 928405771, 3088084685]
mental/mental-roberta-base	[4170216869, 2499548483, 123682896]
mnaylor/psychbert-cased	[2774934368, 701698718, 626148011]
publichealthsurveillance/PHS-BERT	[3390767782, 3500503143, 3647528456]
mental/mental-roberta-base	[950600914, 3051318926, 2011703506]
mental/mental-roberta-base	[2192230731, 2704465100, 552926065]

Appendix C

Untrained Test Results

Table C.1: Performance metrics for each model on the Normal Reddit Test dataset ***before*** *fine-tuning each model.*

Model	Accuracy	F1	Recall	Precision
MentalBERT	0.9816	0.0000	0.0000	0.0000
PHS-BERT	0.9816	0.0000	0.0000	0.0000
PsychBERT	0.2936	0.0485	0.9780	0.0249
DistilBERT	0.0777	0.0355	0.9231	0.0181
MentalRoBERTa	0.0184	0.0362	1	0.0184

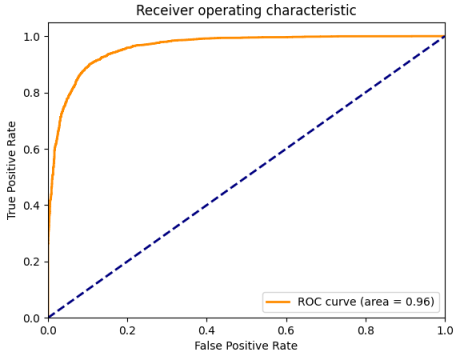
Table C.2: Performance metrics for each model on the IMHR Twitter Test dataset ***before*** *fine-tuning each model.*

Model	Accuracy	F1	Recall	Precision
MentalRoBERTa	0.830140	0.020375	0.118839	0.011143
DistilBERT	0.662619	0.016847	0.194463	0.008805
MentalBERT	0.355799	0.029691	0.663065	0.015185
PsychBERT	0.066904	0.030755	0.995949	0.015619
PHS-BERT	0.014864	0.029293	1.000000	0.014864

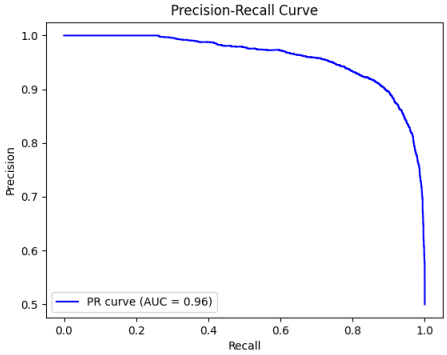
Appendix D

ROC and PRC curves

D.1 Test Dataset

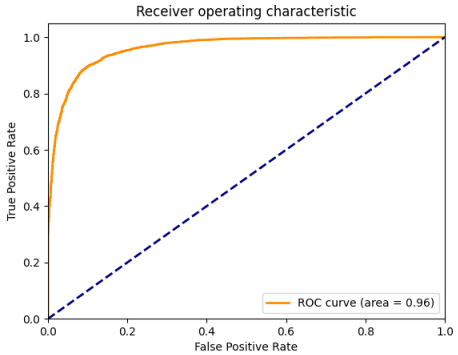


(a) Optimal threshold: 0.0045

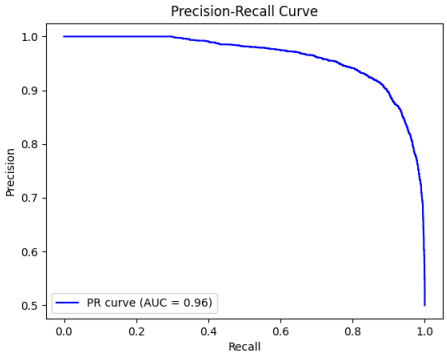


(b) Optimal threshold: 0.5221

Figure D.1: ROC and PRC curves for MentalBERT applied to the standard test dataset

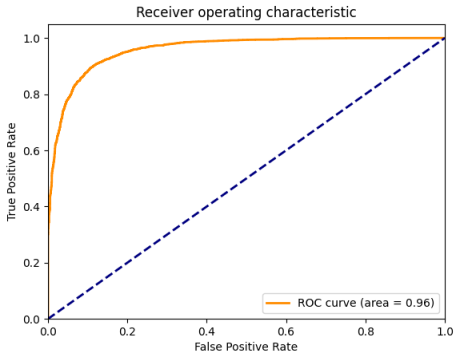


(a) Optimal threshold: 0.0018

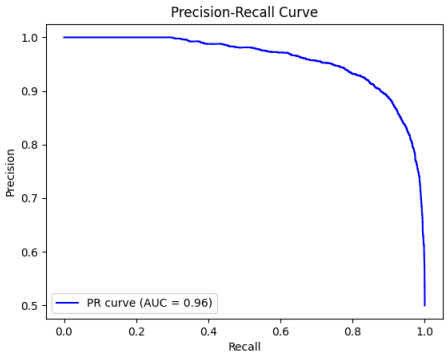


(b) Optimal threshold: 0.6733

Figure D.2: ROC and PRC curves for PHS-BERT applied to the standard test dataset

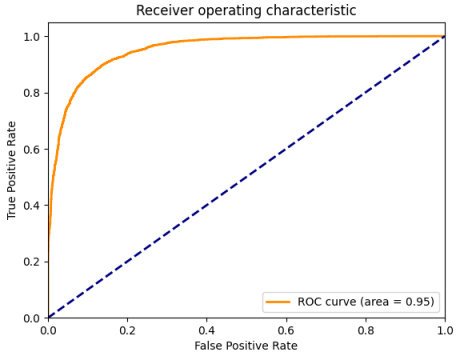


(a) Optimal threshold: 0.0070

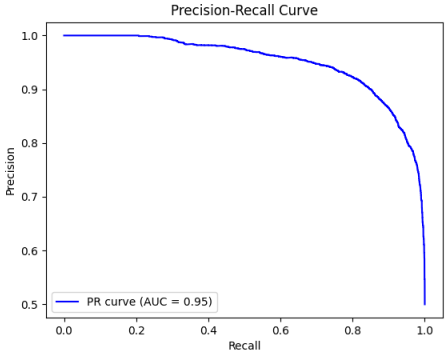


(b) Optimal threshold: 0.5070

Figure D.3: ROC and PRC curves for PsychBERT applied to the standard test dataset

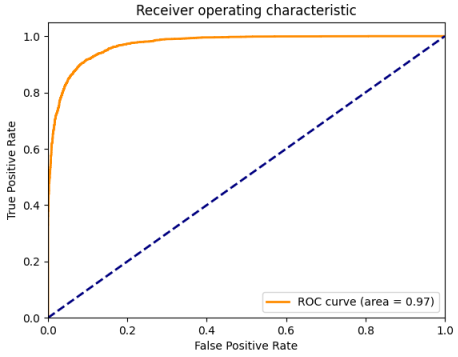


(a) Optimal threshold: 0.0276

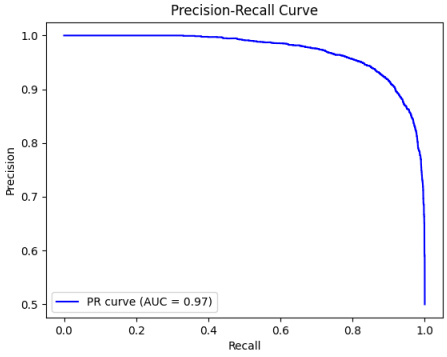


(b) Optimal threshold: 0.5129

Figure D.4: ROC and PRC curves for DistilBERT applied to the standard test dataset



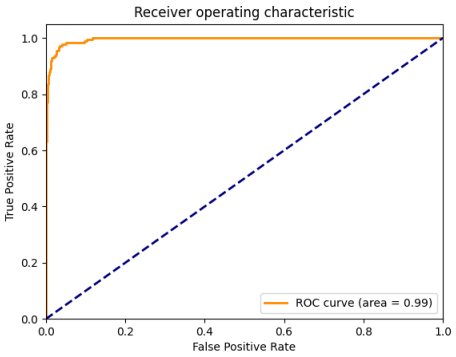
(a) Optimal threshold: 0.0029



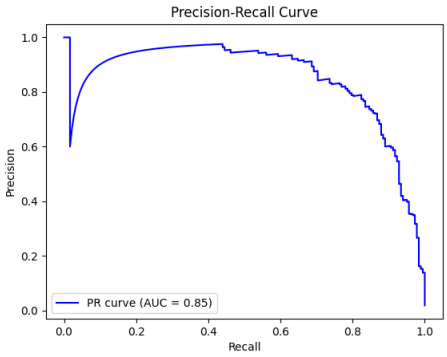
(b) Optimal threshold: 0.5521

Figure D.5: ROC and PRC curves for MentalRoBERTa applied to the standard test dataset

D.2 Normal Reddit Dataset

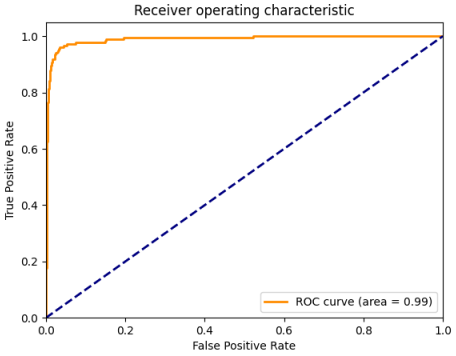


(a) Optimal threshold: 0.0027

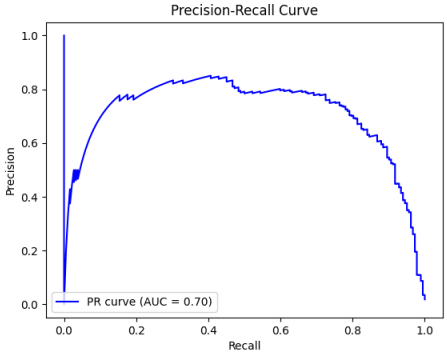


(b) Optimal threshold: 0.6412

Figure D.6: ROC and PRC curves for MentalBERT applied to the normal reddit dataset

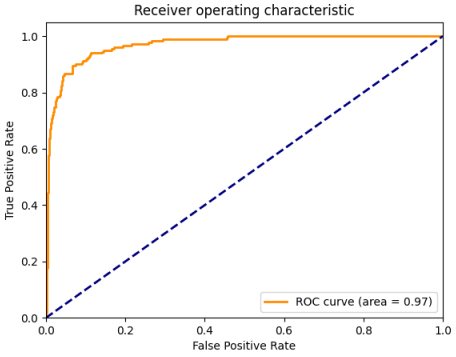


(a) Optimal threshold: 0.0007

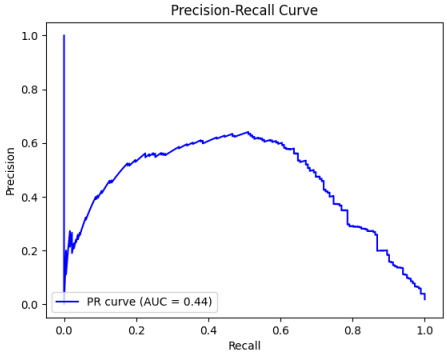


(b) Optimal threshold: 0.9865

Figure D.7: ROC and PRC curves for PHS-BERT applied to the normal reddit dataset

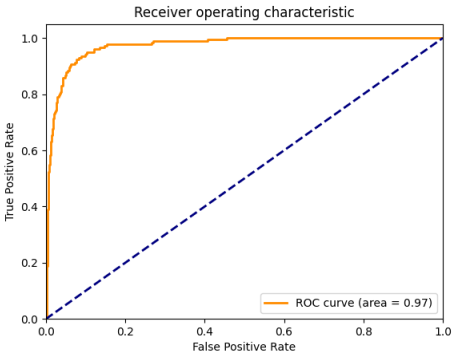


(a) Optimal threshold: 0.0034

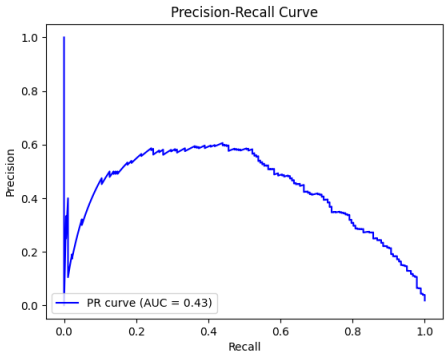


(b) Optimal threshold: 0.9963

Figure D.8: ROC and PRC curves for PsychBERT applied to the normal reddit dataset

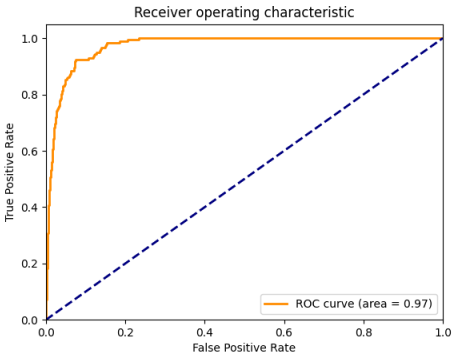


(a) Optimal threshold: 0.0150

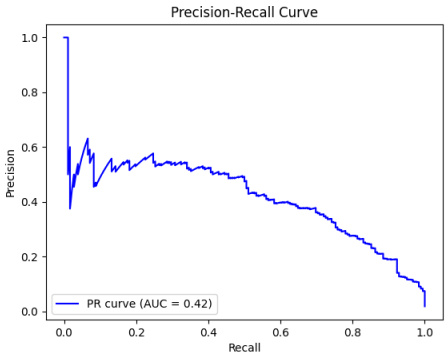


(b) Optimal threshold: 0.9888

Figure D.9: ROC and PRC curves for DistilBERT applied to the normal reddit dataset



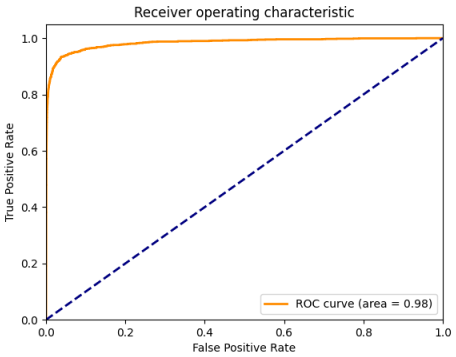
(a) Optimal threshold: 0.0019



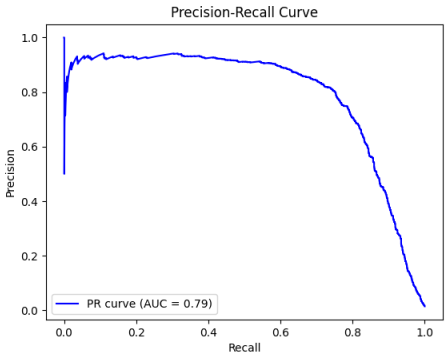
(b) Optimal threshold: 0.9314

Figure D.10: ROC and PRC curves for MentalRoBERTa applied to the normal reddit dataset

D.3 IMHR Dataset

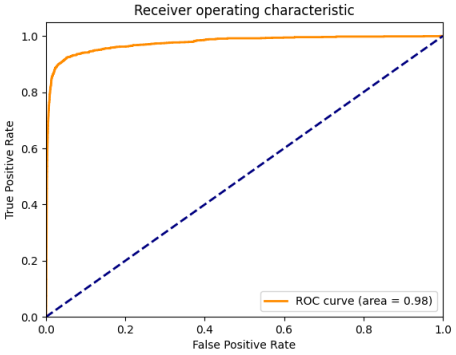


(a) Optimal threshold: 0.0027

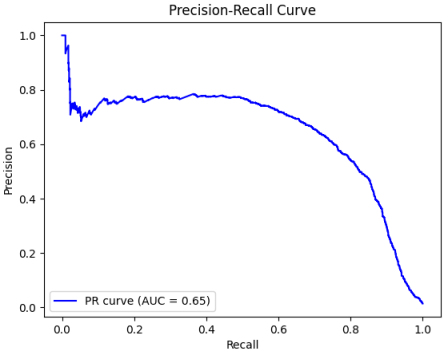


(b) Optimal threshold: 0.7188

Figure D.11: ROC and PRC curves for MentalBERT applied to the IMHR Twitter dataset

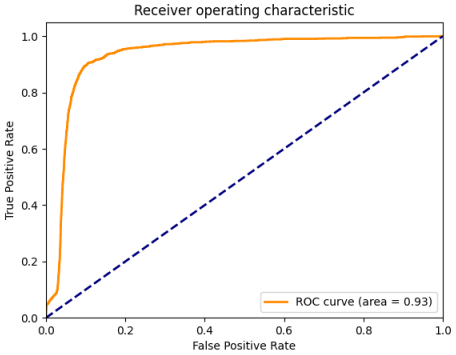


(a) Optimal threshold: 0.0007

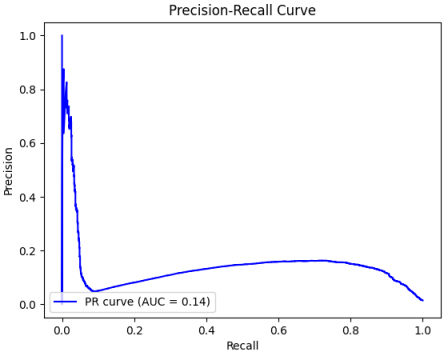


(b) Optimal threshold: 0.8204

Figure D.12: ROC and PRC curves for PHS-BERT applied to the IMHR Twitter dataset

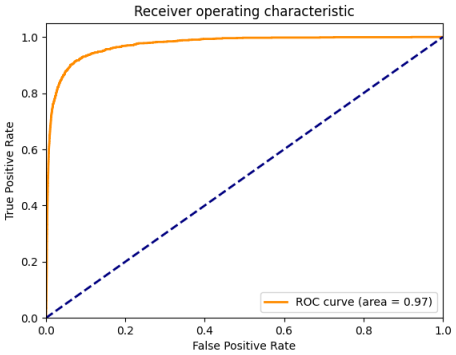


(a) Optimal threshold: 0.0018

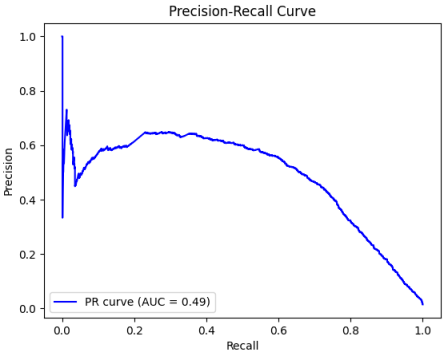


(b) Optimal threshold: 0.9992

Figure D.13: ROC and PRC curves for PsychBERT applied to the IMHR Twitter dataset

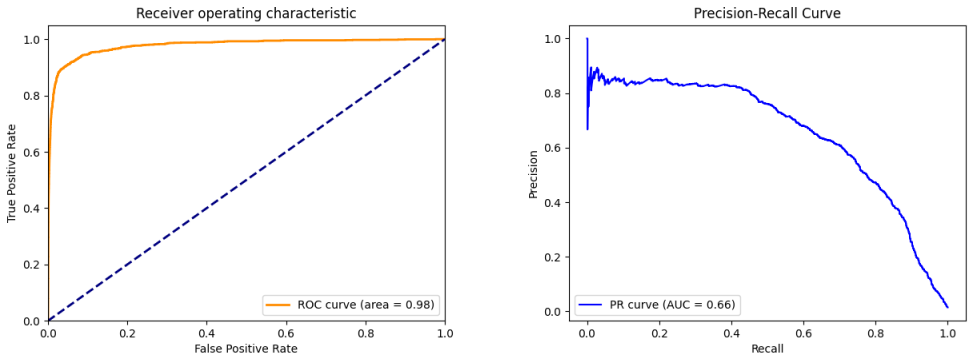


(a) Optimal threshold: 0.0079



(b) Optimal threshold: 0.6619

Figure D.14: ROC and PRC curves for DistilBERT applied to the nIMHR Twitter dataset



(a) Optimal threshold: 0.0018 *(b) Optimal threshold: 0.7178*

Figure D.15: ROC and PRC curves for MentalRoBERTa applied to the IMHR Twitter dataset

Appendix E

Parallel Coordinates Plots

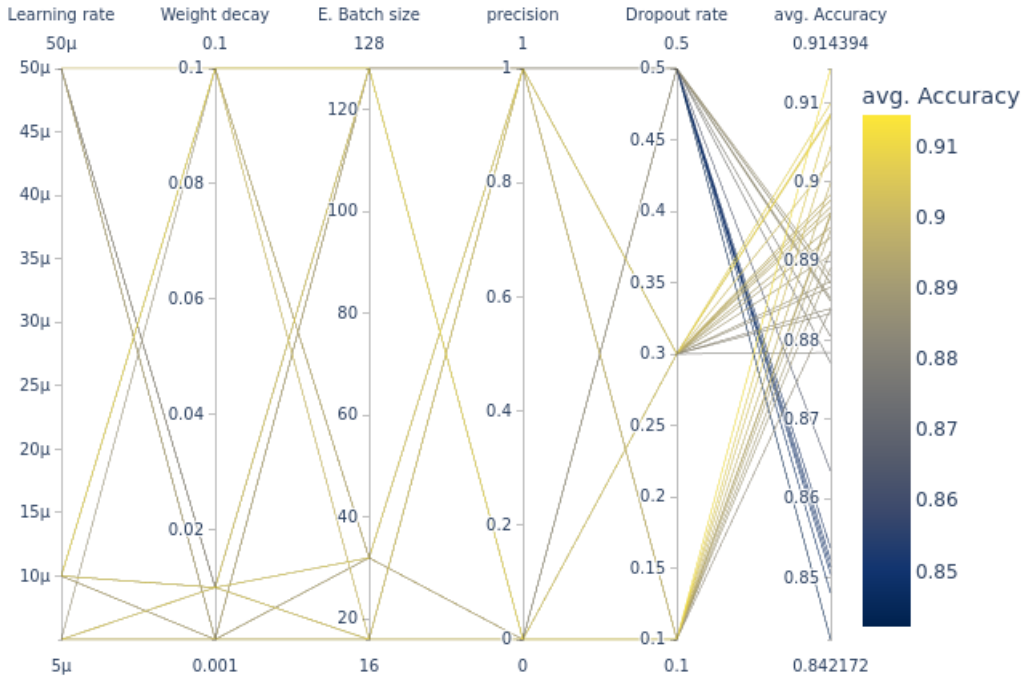


Figure E.1: Parallell coordinates plot to visualize the relationship between hyperparameter values and the average accuracy of the model. Note that the figure above contains the runs for all models.

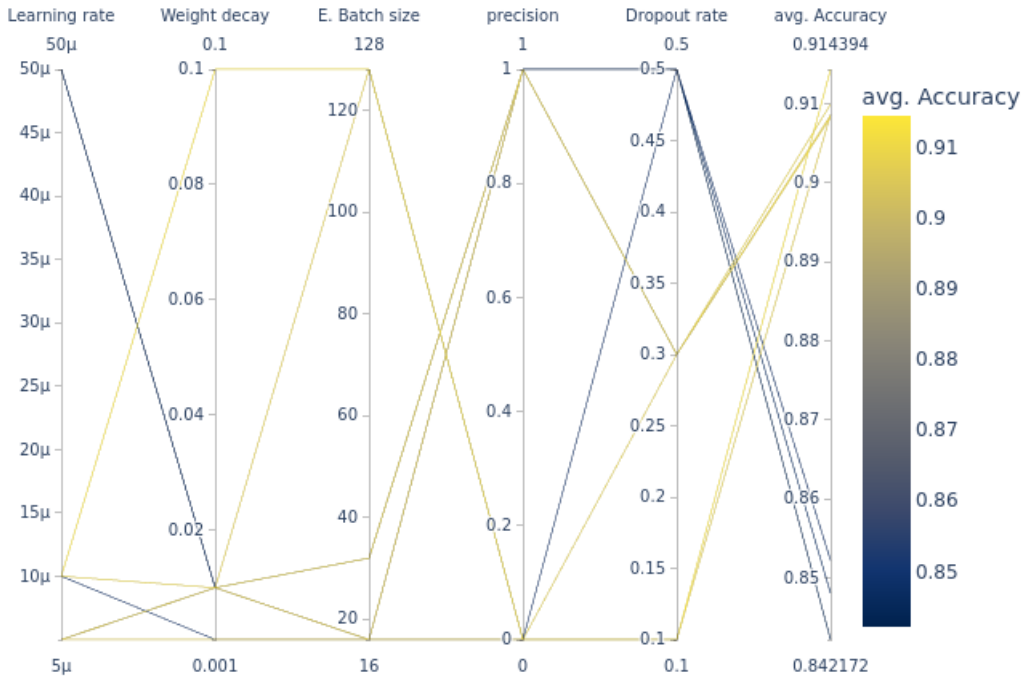


Figure E.2: Overview of hyperparameter selections and their impact on the accuracy metric for MentalRoBERTa model.

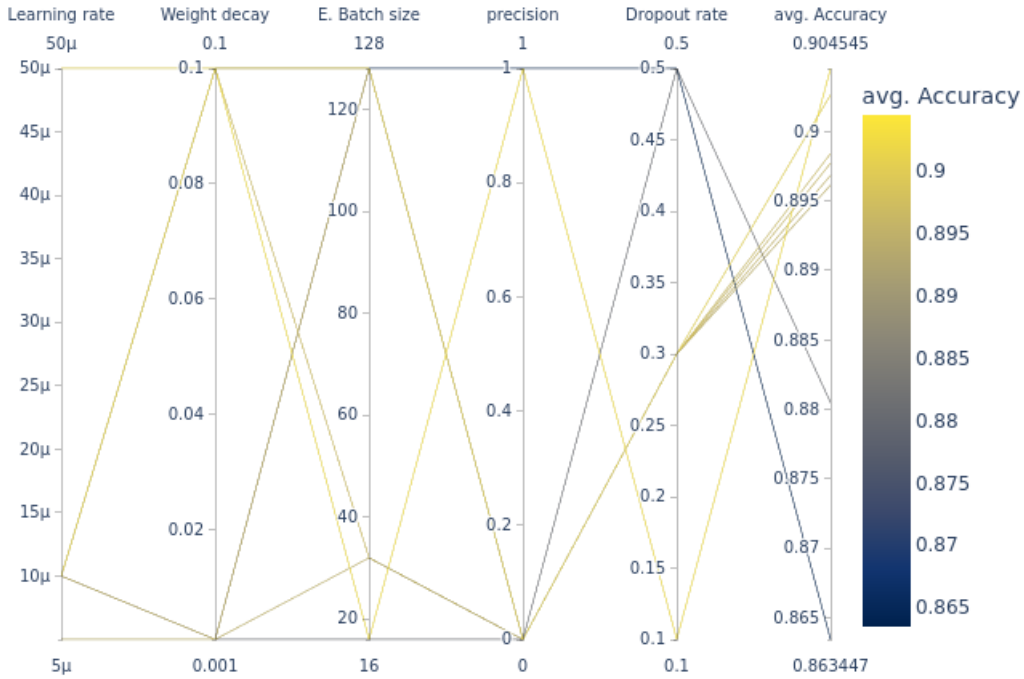


Figure E.3: Overview of hyperparameter selections and their impact on the accuracy metric for MentalBERT model.

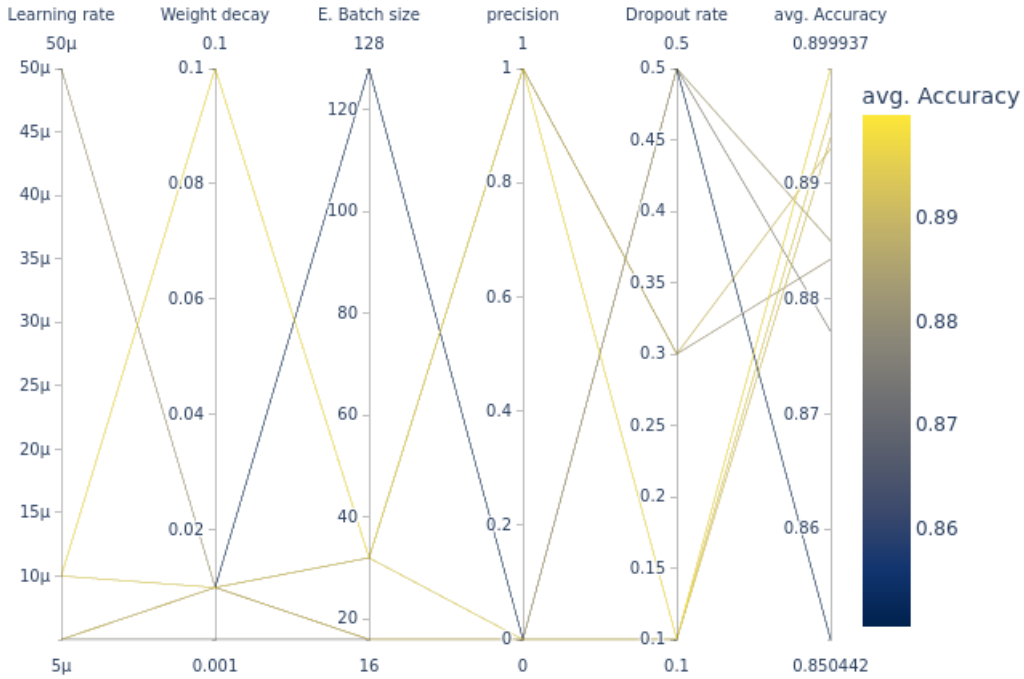


Figure E.4: Overview of hyperparameter selections and their impact on the accuracy metric for PHS-BERT model.

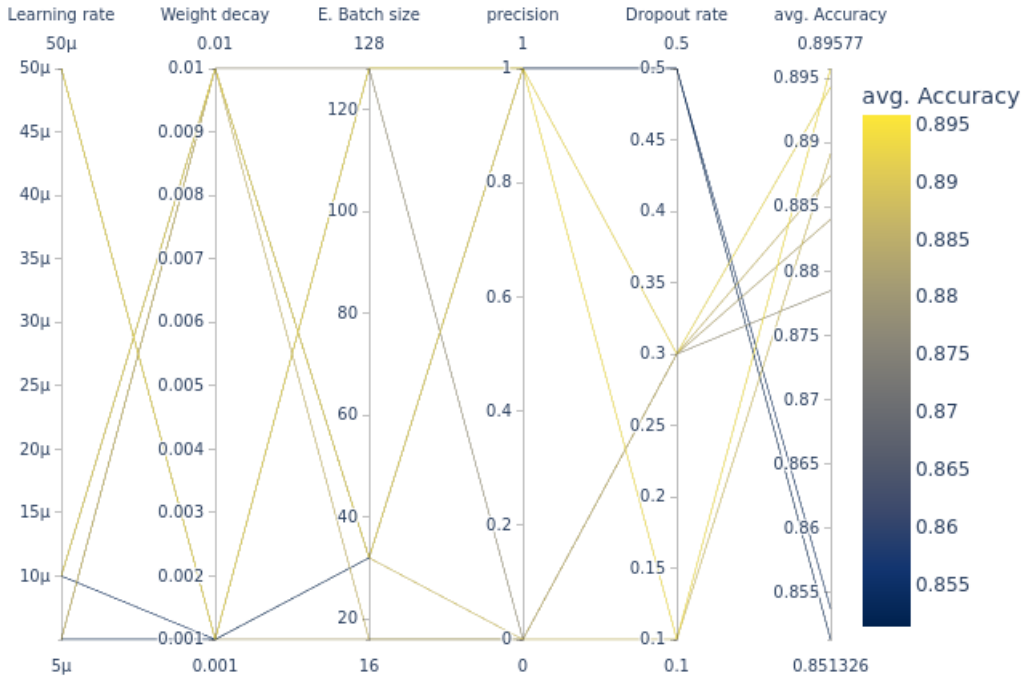


Figure E.5: Overview of hyperparameter selections and their impact on the accuracy metric for PsychBERT model.

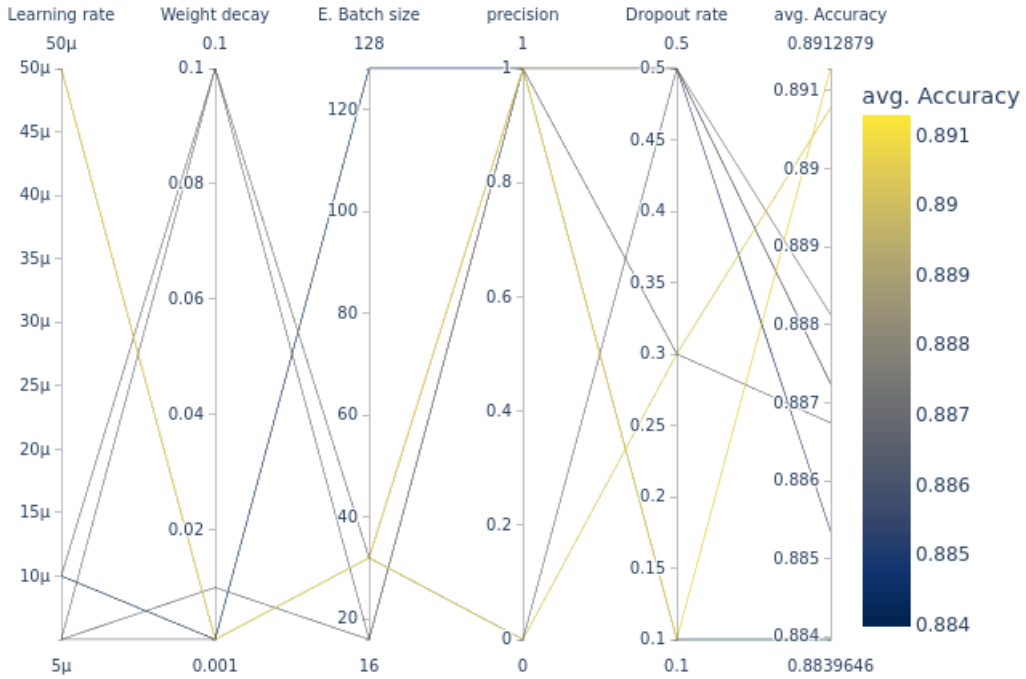


Figure E.6: Overview of hyperparameter selections and their impact on the accuracy metric for DistilBERT model.