

# **Deep Learning for Early Detection of Depression in Reddit**

**A.T.L.H.Y. Samarasinghe  
2021**



# **Deep Learning for Early Detection of Depression in Reddit**

**A dissertation submitted for the Degree of Master of  
Computer Science**

**A.T.L.H.Y. Samarasinghe  
University of Colombo School of Computing  
2021**





## DECLARATION

I hereby declare that the thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis. This thesis has also not been submitted for any degree in any university previously.

Student Name: A.T. L. H. Y. Samarasinghe

Registration Number: 2018/MCS/077

Index Number: 18440776



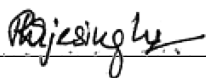
14/09/2021

Signature of the Student & Date

This is to certify that this thesis is based on the work of ~~Mr.~~ /Ms. A.T.L.H.Y. Samarasinghe under my supervision. The thesis has been prepared according to the format stipulated and is of acceptable standard.

Certified by,

Supervisor Name: M.W.A.C.R Wijesinghe



29/11/2021

Signature of the Supervisor & Date

I would like to dedicate this thesis to  
My family for their continuous support and encouragement  
My institution mentors for guiding me through the process with patience  
My friends for motivating me to keep going

## ACKNOWLEDGEMENTS

This work was made possible by the constant support from many individuals.

I would like to express my sincere gratitude to my supervisor Mrs. M.W.A.C.R Wijesinghe for her immense support and encouragement. I'm highly indebted to her for the patience and supervision shown to me throughout the entire research process.

I'm extremely thankful to my co-supervisor Dr. A.R. Weerasinghe for his wealth of knowledge. His suggestions and guidance in narrowing down the research topic was really helpful.

Special thanks to Prof. Losada Carril David Enrique from University of Santiago de Compostela (USC) for providing me access to "eRisk2018" dataset and for his contribution to the research community in the field of mental health and risk assessment.

Further, I express my deepest gratitude to Mr. Randil Pushpananda, Research Project Module Coordinator, Ms. AI Perera, Assistant Coordinator for being so understanding and cooperative during the period of this course.

Lastly, I would like to thank my family for the unconditional support and continuous encouragement and, my colleagues and friends for always motivating me to make progress. This wouldn't have been possible without their helping hands.

## ABSTRACT

Mental health problems represent a major public health challenge worldwide, and depression in particular is a serious and widespread form of mental illness. The major issue with depression is that if it is not diagnosed and treated as early as possible, the negative impact it can have on the individual's life is huge. In many cases, professionals do not have access to patients during early stages of depression unless patients themselves or their relatives report the symptoms to a clinician. However, with the continuous increase in popularity of social media platforms among people all over the world, it was discovered that social media posts could be favorable in detecting mental health illnesses. Therefore, in this study we developed a state-of-art deep learning model to predict the likelihood of a social media user to be diagnosed with depression at an early stage. Then we evaluated the model using an Early Risk Detection Error (ERDE) metric which rewards early detections and penalize late detections. The results show that our deep learning model performs reasonably well on Reddit platform, but there's more room for improvement, particularly with imbalanced datasets.

# TABLE OF CONTENTS

<a href="#">ACKNOWLEDGMENTS</a> .....	iii
<a href="#">ABSTRACT</a> .....	iv
<a href="#">TABLE OF CONTENTS</a> .....	v
<a href="#">LIST OF FIGURES</a> .....	vi
<a href="#">LIST OF TABLES</a> .....	vii
<a href="#">APPENDICES</a> .....	viii
<a href="#">CHAPTER 1: INTRODUCTION</a> .....	1
<a href="#">1.1 Motivation</a> .....	1
<a href="#">1.2 Statement of the problem</a> .....	1
<a href="#">1.3 Research Aim and Objectives</a> .....	3
<a href="#">1.3.1 Aim</a> .....	3
<a href="#">1.3.2 Objectives</a> .....	3
<a href="#">1.4 Scope</a> .....	3
<a href="#">1.5 Structure of the thesis</a> .....	4
<a href="#">CHAPTER 2: LITERATURE REVIEW</a> .....	5
<a href="#">2.1 Introduction</a> .....	5
<a href="#">2.2 A Literature Review</a> .....	5
<a href="#">2.2.1 Detection of Depression</a> .....	5
<a href="#">2.2.2 Early Detection of Depression</a> .....	7
<a href="#">2.3 Summary</a> .....	10
<a href="#">CHAPTER 3: METHODOLOGY</a> .....	13
<a href="#">3.1 Machine Learning Techniques</a> .....	13
<a href="#">3.1.1 Deep Learning</a> .....	13
<a href="#">3.1.2 BERT</a> .....	14
<a href="#">3.1.3 K-Fold Cross Validation</a> .....	15
<a href="#">3.2 Dataset</a> .....	16



<u>3.2.1 Introduction</u> .....	16
<u>3.2.2 Data</u> .....	17
<u>3.2.3 Structure</u> .....	18
<u>3.2.4 Summary</u> .....	19
<u>3.2.5 Process</u> .....	20
<u>3.3 Methodology</u> .....	20
<u>3.3.1 Data Preparation</u> .....	21
<u>3.3.2 Language Model Fine-Tuning</u> .....	21
<u>3.3.3 Fine-Tuning BERT for Text Classification</u> .....	23
<u>3.3.4 Prediction</u> .....	24
<u>CHAPTER 4: EVALUATION AND RESULTS</u> .....	26
<u>4.1 Evaluation</u> .....	26
<u>4.1.1 Perplexity</u> .....	26
<u>4.1.2 Performance Metrics</u> .....	26
<u>4.1.3 ERDE Metric</u> .....	27
<u>4.2 Results</u> .....	28
<u>4.2.1 Perplexity</u> .....	28
<u>4.2.2 Performance Metrics</u> .....	29
<u>4.2.3 ERDE Metric</u> .....	29
<u>CHAPTER 5: CONCLUSION AND FUTURE WORK</u> .....	30
<u>APPENDICES</u> .....	I
<u>REFERENCES</u> .....	IX

## LIST OF FIGURES

Figure 3.1: BERT captures the context of both left and right .....	14
Figure 3.2: Example of Next Sentence Prediction .....	15
Figure 3.3: File Structure of eRisk2018 Dataset .....	18
Figure 3.4: XML File Format .....	18
Figure 3.5: XML File Example .....	19
Figure 3.6: Overview of the methodology.....	20
Figure 3.7: Language Model data format .....	21
Figure 3.8: Classification Model data format.....	22
Figure 3.9: Evaluation Loss of Classification Model .....	24
Figure 3.10: Prediction result file content .....	25
Figure 3.11: Prediction results file format.....	25

## LIST OF TABLES

Table 2.1: Summary of Literature .....	10
Table 3.1: Summary of eRisk 2018 dataset .....	19
Table 3.2: Summary of data in each split .....	21
Table 3.3: Language Model hyperparameters .....	23
Table 3.4: Classification Model hyperparameters .....	23
Table 4.1: Language Model Output.....	28
Table 4.2: Performance metrics for each split.....	29
Table 4.3: ERDE metric for all chunks .....	29

## **APPENDICES**

APPENDIX A: eRisk2018 User Agreement .....	I
APPENDIX B: Source Code of the Language Model .....	III
APPENDIX C: Source Code of the Classification Model .....	IV
APPENDIX D: Source Code for Prediction .....	VI
APPENDIX E: Source Code for ERDE Calculation .....	VIII

# CHAPTER 1

## INTRODUCTION

This chapter presents the motivation for this study on early detection of depression, explains the state of problem, aims and objectives and scope of the work.

### 1.1 Motivation

Social media such as Twitter, Facebook, Reddit and Online discussion forums have become extremely popular as open and free communication platforms where people can self-express, share feelings and enjoy their experiences with others in the community. Many people who go through difficult times find it easy to verbalize their internal restlessness on online platforms when true feelings are difficult to articulate. Due to this reason, researchers discovered that social media can be used for detection, prevention and intervention of mental health illnesses (Coppersmith et al., 2015), so that professional help can be directed towards the people who are in need.

As depression is a major contributor for the increasing numbers of mental health issues, it will be a great help for professionals if depression can be identified via these online platforms. Further, it is a known fact that if depression is not treated in a timely manner, it can affect the daily lifestyle, ruin relationships, increase risky behavior and gradually make it difficult to overcome the illness. Therefore, early diagnosis is a crucial aspect of depression.

However, in most cases professionals do not have access to patients during early stages of diagnosis unless patients themselves or their relatives report the symptoms to a clinician. But, if there is an automated system to analyze the language in user posts on social media and alert the professionals, it would be beneficial for the society in general.

### 1.2 Statement of the problem

Depression is known to be one of the most common mental illnesses which has become a major issue for mental health practitioners as professional help is inaccessible to a larger population and the social stigma around depression prevents people from reaching out to medical experts. Further, professional help can be expensive, time consuming and depression can be often misdiagnosed. Depression can negatively affect daily activities of a person's life. It can cause individuals to suffer greatly and lose interest or pleasure in activities once

enjoyed, loose appetite, experience difficulty in sleeping or sleeping too much, feeling worthless and etc. When depression is long lasting with moderate and severe intensity, it can even lead to suicide. According to WHO, more than 264 million people are affected with depression worldwide and closer to 800 000 people die due to suicide every year.

With the advancements in Natural Language Processing (NLP), computational linguistics and text analysis methods, several studies have been carried out to detect depression using social media based on the posts the individuals share. Few areas of interest in this domain are identifying posts as depressive posts (Fatima et al., 2017), classifying users as depressed or healthy (Resnik et al., 2015; Singh and Wang, n.d.), detecting the degree of depression as mild, moderate and severe, detecting the degree of depression as a score (Schwartz et al., 2014; Stephen and P., 2019) and early detection of depression (Owen et al., 2020; Wald et al., 2012). Among these, early detection can be considered as the most beneficial for professionals because, if depression is detected early, proper treatments can be offered to reduce the negative effects before things start to escalate to a suicidal level. Hence, this study proposes a method for early detection of depression using social media, as they are excellent platforms for providing public records over a long period of time and it reveals lot of information about user's health and changes in their mental state.

Previous researches have mostly been carried out for early detection using clinically diagnosed/self-reported cases (De Choudhury et al., 2014, 2013). The problem in this scenario is that users themselves claim that they have been diagnosed with depression. Therefore, verifying the reliability of the information difficult. Most of the other studies based on early detection used linguistic, semantic and writing features to build their models (A. G. Reece et al., 2017) and few of them used deep learning techniques. (Orabi et al., 2018)

Even though these models performed well in predicting depressed and non-depressed samples for the task of early detection, there was no proper method to evaluate how early they diagnosed the cases. Previous studies were not able to assess early risk of depression properly.

Furthermore, the latest discovery in the field of NLP is the introduction to transformer models (Devlin et al., 2019). These models have not yet been widely used for mental health related tasks. Therefore, there is an absence in state-of-art models for NLP, tested with proper evaluation metrics to support early alert systems.

## **1.3 Research Aims and Objectives**

The primary focus of this research is expressed in terms of aims and objectives.

### **1.3.1 Aim**

This research project aims to propose a novel approach for early detection of depression for self-expressed cases in social media, specifically in Reddit platform, by using one of the latest deep learning models called BERT. Additionally, this study aims to evaluate the model using a new metric called ERDE (Early Risk Detection Error) and find out how well BERT models perform on early risk detection tasks. This approach will be helpful for professionals to detect users who are likely to be diagnosed with depression in the future, at a stage where even the users themselves are unaware of the instabilities in their mental state.

### **1.3.2 Objectives**

This study intent to achieve the following objectives.

1. To understand and perform a critical study on the relationship between depression and social media.
2. To identify the suitability of deep learning models like BERT for the early detection of depression
3. To develop a model to predict the likelihood of a user to be diagnosed with depression at an early stage.
4. To evaluate the model using early risk detection metric and discover how well the model performs during early stages.

## **1.4 Scope**

This study focuses on developing a deep learning model for early detection of depression using a publicly available dataset ('2018 CLEF eRisk') on self-expressed cases. Since the dataset is based on English language, the model will be applicable to only English-speaking users.

## **1.5 Structure of the Thesis**

The thesis gives an introduction to the problem followed by the aims and objectives of the research. Next, the literature review and a summary of papers is presented. The methodology includes a brief introduction to machine learning techniques, detailed description of the dataset and how the model is trained for the prediction task. Then the evaluation metrics are discussed and finally the results are presented with the conclusion and future work.



# **CHAPTER 2**

## **LITERATURE REVIEW**

### **2.1 Introduction**

Currently, medical practitioners evaluate depression mostly through surveys based on patients' self-reported experiences and statements. Thereafter, they analyze symptoms and classify them into distinct categories (absent, mild, moderate, severe) in order to prescribe treatments. With the evolution of machine learning and deep learning techniques in the field of Natural Language Processing, various studies have been carried out to explore the potential of detecting symptoms and characteristics of mental health illnesses through social media content. This information could serve as a guidance for clinicians and as well as for patients. Over the years, the studies on depression in particular, have been able to reveal significant markers for detecting the degree of depression in posts, detecting depression in users, and predicting depression of users at early stages.

### **2.2 Literature Review**

#### **2.2.1. Detection of Depression**

In an attempt to detecting posts and communities as depressive and non-depressive, (Fatima et al., 2017) performed a linguistic and semantic analysis with machine learning techniques to a dataset extracted from a popular blogging site called LiveJournal. Each post classified as depressive was then characterized according to the severity of depression as mild, moderate and severe. Despite the fact that this study focused more on classifying posts instead of detecting users with depression, it produced accurate results in identifying the severity of depression in posts.

Many researches have worked on detecting depression of users by analyzing their language on social media platforms. (De Choudhury et al., 2013) carried out a clinical survey to collect data using crowdsourcing method, then developed a supervised model based on emotional expression, linguistic style, user engagement, and egocentric social network properties. Further, a social media depression index (SMDI) was introduced to predict and compare depression rates in cities/states and at diurnal and annual scales across men and women. Additionally, the authors were able to verify that users with depression tend to use first-person pronouns more often in their writings. Another study (Resnik et al., 2015) worked on mainly two publically

available datasets and several sophisticated LDA models, topic modeling techniques, to detect depression by analysing linguistic style. These improved models were able to derive topics with high predictive value in user's language.

(Orabi et al., 2018) assessed a deep learning approach to identify users at risk with depression using a publically available Twitter dataset. This novel architecture introduced an optimized word embedding modal which could be incorporated in deep learning models to detect depression of twitter users with limited data. In another study (Singh and Wang, n.d.), the tweets scraped from various Twitter posts were fed into several deep learning models to predict depression of each user. A comparative analysis was carried out to examine the effects of character based vs word based models and pretrained embeddings vs learned embeddings. The results declared that word based GRU and CNN models performed well with high accuracy.

Then researchers became interested in detecting the degree of depression in online users. (Stephen and P., 2019) introduced a mechanism to calculate the level-of-depression /depression-score of Twitter users based on the emotional integrity of their tweets. Results were validated via manual intervention. It was disclosed that depression levels do coorelate with tweets and users with depression published tweets during specific hours of the day and during specific days of the week. Another study (Schwartz et al., 2014) presented an approach to predict the degree of depression of an individual entirely based on the laguage used in their Facebook updates. The degree of depression was assessed over time as a continuos value rather than classes. For that, authors built a regression model on survey responses and Facebook status updates. When estimating seasonal changes of users, it was found that degree of depression often increases from summer to winter.

While the task of detecting online users with depression was becoming a successful venture, researchers recently started investigating more on early detection of depression as it can be beneficial for clinicians and patients to identify symptoms at an early stage.

### **2.2.2. Early Detection of Depression**

This approach allows professionals to attend to individuals who are at risk, and provide treatments before depression can do any damage to their lives.

One of the earliest developments which provided a basis for predicting depression via social media was (Choudhury et al., 2013). This study emphasized on the potential of using Twitter as a tool to predict whether an individual is likely to suffer from depression in the future. Crowdsourcing was used to collect assessments from Twitter users with MDD (Major Depressive Disorder). By analysing the social media behavior of users during past one year (dating back from the reported onset or the day survey was taken), it was shown that users with depression have lowered social activity, greater negative emotion, high self-attentional focus, increased relational and medicinal concerns, and intense expression of religious thoughts. The authors undertook another research to understand the feasibility of using Facebook to detect and predict onset of Postpartum Depression (PPD) in new mothers (De Choudhury et al., 2014). An online survey was conducted to gather self-reported diagnoses of PPD, along with scores of a common depression screening tool. By analysing data available before childbirth, a series of statistical models were developed to predict a mother's likelihood of PPD.

The authors of (A. G. Reece et al., 2017) gathered the date of first depression diagnosis of Twitter users via crowdsourcing, then categorized Twitter data of the selected users into 'healthy' and 'affected' groups. Furthermore, a timeline for 'depression and onset recovery' was portrayed as per 'user-days'. This revealed that there is a high probability for depressed individuals to show symptoms even in the period of nine months prior to diagnosis. Three months before the diagnosis, there seemed to be a rise in depression and, post-diagnosis the probability of depression began to decrease after 3-4 months.

The crowdsourcing method, collect data from users who claimed to be diagnosed with depression or users who have already suffered through depressive episodes in the past. Hence, the reliability of the diagnosis dates gathered from these self-reported individuals is questionable. Further, crowdsourcing is a time consuming and a heavy process. Instead, it would be more useful if we can filter and extract data from social media platforms directly and manually annotate them with a professional's intervention. The reason being that sometimes users are unaware of their own mental

state, so when they express themselves explicitly on online platforms, the revealing information can be very beneficial.

An interesting turn in the field of early risk detection took place when CLEF (Conference and Labs for the Evaluation Forum) organized the ‘eRisk2018’ workshop (“CLEF eRisk: Early risk prediction on the Internet | CLEF 2018 workshop,” n.d.) where teams could participate in two tasks namely, ‘Early detection of Depression’ and ‘Early detection of Anorexia’. The organizers released a valid dataset which is manually labelled and the challenge was to detect the traces of depression by sequentially processing user posts in social media. eRisk2018 also proposed a new evaluation metric called ERDE (Early Risk Detection Error) to reward early detections and penalize late detections. As and when the submissions from users are pushed to the system as chunks, it has to make a decision as early as possible. The purpose of the suggested mechanism is to compare the effectiveness of proposed models which should maximise the F1 score for positive cases and minimize the overall ERDE value.

One of the participants (Cacheda et al., 2019), used eRisk2018 dataset extracted from Reddit to analyse the user behaviour based on textual, semantic and writing features and built two models for the task of prediction, singleton model to identify depressive cases and dual model to address non depressive cases. Both models produced moderate results for the ERDE metric, out of which dual model performed the best. Another study (Maupome and Meurs, 2019), incorporated topic modelling to identify early signs of depression and a simple neural network MLP (Multi Layer Perceptron) to predict results. The model seemed to favour non depressive samples and there was a noticeable delay in taking the decision as to user indicates depressive symptoms or not. One of the best performing systems that was submitted to the workshop was developed by (Ramírez-Cifuentes and Freire, 2018). The model employed linguistic information and depression related vocabulary with Logistic Regression classifier. The authors discovered that processing text, post by post rather than chunk by chunk can further reduce the ERDE value. This UPFA model produced lowest ERDE<sub>50</sub> score which is 6.41% among all candidates. (Bucur and Dinu, 2020) also used a topic modelling approach to extract embeddings from user posts and feed them to a fully connected neural network. Additionally, the model incorporates a confidence score to guide the decision process. This addresses the problem of having to wait till the last chunk of the dataset to make the decision. Hence, the model emits the decision, whether the user is likely to be diagnosed with depression, as early as possible.

As discussed above, most of the proposed systems for the task of detecting depression in eRisk2018 workshop, used features such as n-grams, topic modelling, Linguistic Inquiry and word Count (LIWC), emotion and semantic indicators and other user metadata. Most commonly used methods were Logistic Regression, Support Vector Machines (SVM), Random Forests and neural networks.

However, the latest deep learning techniques and algorithms have not been tested on this dataset. One of the emerging methods to solve various NLP tasks is transformer model. The most popular transformer model is known as BERT (Bidirectional Encoder Representations from Transformers) from Google. According to (González-Carvajal and Garrido-Merchán, 2021) who compared BERT model with different traditional machine learning algorithms (Logistic Regression, Voting Classifier and etc) retrieving TF-IDF (Term Frequency - Inverse Document Frequency) as feature inputs on different datasets, verified that BERT outperforms most of the traditional approaches. But as any other model, BERT also has its own limitations. Another study (Maslej-Krešňáková et al., 2020) compared BERT with already existing similar models (CNN with GloVe(CC) embeddings and BiLSTM and etc.) which again offered better results.

In the field of mental health, the authors (Owen et al., 2020) extracted tweets from Twitter API to predict depression and anxiety in Twitter users who have not been diagnosed with relavent symptoms yet. Three human annotators were involved in the process of labelling the posts. The results of the comparative analysis between pre-trained language models like BERT and ALBERT and traditional linear models suggests that LMs performed relatively well but when data was unbalanced, traditional models were on par.

Due to the promising results produced by BERT modal on various other NLP tasks, this study proposes a novel approach to detect the early stages of depression using BERT modal and evaluate the early detection with ERDE metric.

## 2.3 Summary

Table 2.1: Summary of Literature

<p>Analysis of user-generated content from online social communities of characterize and predict depression degree (Fatima et al., 2017)</p>	<ul style="list-style-type: none"> <li>• Data extracted from a blogging site named ‘LiveJournal’</li> <li>• Post and community classification using LIWC features and Random Forest classifier</li> <li>• Depression degree (mild, moderate, and severe) analysis using Live Journal mood tags</li> <li>• Above 92% accuracy in post, community and degree classification</li> <li>• User classification was not addressed</li> </ul>
<p>Social Media as a Measurement Tool of Depression in Populations (De Choudhury et al., 2013)</p>	<ul style="list-style-type: none"> <li>• Crowdsourcing to collect clinically diagnosed date along with data from Twitter. (Self-reported cases)</li> <li>• Engagement and ego network features, n-grams, linguistic style, emotion and time features were used.</li> <li>• Social Media Depression Index (SMDI) to predict degree of depression based on daily postings of users.</li> <li>• SVM Classifier</li> <li>• Analysis on population characteristics of depression using SMDI</li> </ul>
<p>Beyond LDA: Exploring Supervised Topic Modeling for Depression-Related Language in Twitter (Resnik et al., 2015)</p>	<ul style="list-style-type: none"> <li>• CLPsych 2015 publically available Twitter dataset (Self-reported)</li> <li>• Stream-of-consciousness publically available dataset for LDA learning process</li> <li>• Supervised LDA, Supervised Anchor Model and Supervised Nested LDA</li> </ul>
<p>Deep Learning for Depression Detection of Twitter Users (Orabi et al., 2018)</p>	<ul style="list-style-type: none"> <li>• CLPsych 2015 and Bell Let’s Talk datasets</li> <li>• Introduced an optimized word embedder for deep learning classifications</li> <li>• CNN (CNNWithMax, MultiChannelCNN, MultiChannelPoolingCNN) and RNN</li> </ul>
<p>Detecting Depression Through Tweets (Singh and Wang, n.d.)</p>	<ul style="list-style-type: none"> <li>• Filtered out Tweets and labelled manually</li> <li>• Word embeddings using Word2vec</li> <li>• Word based CNN, RNN and GRU models</li> <li>• Character based GRU model</li> <li>• Word based GRU model performed best.</li> </ul>
<p>Detecting the magnitude of depression in Twitter users using sentiment analysis (Stephen and P., 2019)</p>	<ul style="list-style-type: none"> <li>• Filtered out Tweets using tags like #abuse, #anxiety, #addict, #addiction, and #bullying</li> <li>• Sentiment analysis on tweets by calculating base emotions</li> <li>• Calculated sentiment scores for each user</li> <li>• Calculated final depression magnitude for each user by adding weighted scores for emotions indicated in their tweets</li> <li>• Revealed that tweets were posted during specific hours of the</li> </ul>

	day and specific days of the week amidst the depressive episodes
Towards Assessing Changes in Degree of Depression through Facebook (Schwartz et al., 2014)	<ul style="list-style-type: none"> <li>• Data collected through a Facebook application and a questionnaire</li> <li>• Features include n-grams, topics from LDA, Lexica from LIWC and no. of words</li> <li>• Regression modelling</li> <li>• Accuracy evaluation with Pearson correlation coefficient</li> </ul>
Predicting Depression via social media (Choudhury et al., 2013)	<ul style="list-style-type: none"> <li>• Crowdsourcing to collect clinically diagnosed date along with data from Twitter and CES-D screening test (Self-reported cases)</li> <li>• Features related to engagement, egocentric social graph, emotions, linguistic style, depression language, and diurnal activity</li> <li>• SVM model with linguistic features alone performed the best.</li> <li>• Findings show that users with depression have lowered social activity, greater negative emotion, high self-attentional focus, increased relational and medicinal concerns, and intense expression of religious thoughts</li> </ul>
Characterizing and Predicting Postpartum Depression from Shared Facebook Data (De Choudhury et al., 2014)	<ul style="list-style-type: none"> <li>• Online survey on Facebook based on PHQ-9 depression screening tool</li> <li>• Considered both prenatal (50 weeks) and postnatal (10 weeks)</li> <li>• Features related to user characteristics, social capital, emotions, linguistic style</li> <li>• Stepwise Logistic Model</li> <li>• Findings show that mothers with depression posted more about their concerns and questions on social media and less likely to show their depressive emotions online.</li> </ul>
Forecasting the onset and course of mental illness with Twitter data (A. Reece et al., 2017)	<ul style="list-style-type: none"> <li>• Crowdsourcing to collect clinically diagnosed date along with data from Twitter and CES-D screening test (Self-reported cases)</li> <li>• labMT, LIWC and ANEW unigrams to quantify the happiness expressed by tweets.</li> <li>• 1200-tree Random Forests classifier</li> <li>• Time series analysis using two state Hidden Markov Model.</li> <li>• Findings suggests that there's a high probability for depressed individuals to show symptoms even in the period of nine months prior to diagnosis.</li> </ul>
Early Detection of Depression: Social Network	<ul style="list-style-type: none"> <li>• eRisk2018 dataset</li> <li>• Singleton model to predict depression cases</li> </ul>

<p>Analysis and Random Forest Techniques (Cacheda et al., 2019)</p>	<ul style="list-style-type: none"> <li>• Dual model to predict nondepression cases</li> <li>• Random Forest Classifier</li> <li>• Features related to textual spreading (time gap, time span), text similarity features (Bag of words, IDF) and semantic similarity features(LSA)</li> <li>• Shrinking threshold value to classify depression samples</li> <li>• ERDE metric</li> </ul>
<p>Using Topic Extraction on Social Media Content for the Early Detection of Depression (Maupome and Meurs, 2019)</p>	<ul style="list-style-type: none"> <li>• eRisk 2018 dataset.</li> <li>• Topic extraction using LDA</li> <li>• Multi Layer Perceptron (MLP) for prediction</li> <li>• Shrinking threshold value to classify depression samples</li> <li>• ERDE<sub>5</sub> – 10.04% and ERDE<sub>50</sub> – 7.85%</li> </ul>
<p>UPF's Participation at the CLEF eRisk 2018: Early Risk Prediction on the Internet (Ramírez-Cifuentes and Freire, 2018)</p>	<ul style="list-style-type: none"> <li>• eRisk 2018 dataset</li> <li>• Features include depression related vocabulary (LIWC), N-grams and vocabulary with added weighted scores.</li> <li>• Logistic Regression and Random Forest</li> <li>• Threshold value to classify depression samples</li> <li>• Best performing model is Logistic Regression model developed using N-grams and LIWC</li> <li>• 5<sup>th</sup> place in the workshop for F1 Score – 0.55</li> <li>• Lowest value for ERDE<sub>50</sub> – 6.41%</li> </ul>
<p>Detecting Early Onset of Depression from Social Media Text using Learned Confidence Scores (Bucur and Dinu, 2020)</p>	<ul style="list-style-type: none"> <li>• eRisk 2018 dataset</li> <li>• Latent Semantic Indexing modal to extract embeddings from text to be used as input to the neural network.</li> <li>• Emits the decision if the out-of-distribution confidence score is above a certain threshold.</li> <li>• Start labelling users at early stages.</li> </ul>
<p>Towards Preemptive Detection of Depression and Anxiety in Twitter (Owen et al., 2020)</p>	<ul style="list-style-type: none"> <li>• Filtered tweets on terms like ‘depress’, ‘anxie’, or ‘anxio’, but not ‘diagnos’</li> <li>• Three human annotators labelled the dataset.</li> <li>• SVM with TD-IDF and/or word embeddings</li> <li>• BERT and ALBERT language models</li> <li>• BERT performed well despite being trained on pre-trained model</li> </ul>



## **CHAPTER 3**

### **METHODOLOGY**

This chapter provides a brief introduction to machine learning techniques used in the methodology, then explains the design overview, presents a detail description of the dataset and how the model is trained to predict the likelihood of a user to be diagnosed with depression.

#### **3.1 Machine Learning Techniques**

For this NLP based study, we have used machine learning to automate the task of identifying the early risk and classifying the users on social media as depressed and non-depressed.

##### **3.1.1 Deep Learning**

Deep learning is a subfield of machine learning which is also known as deep neural networks. This is due to the reason that it uses neural network architectures. The term “deep” denotes the number of hidden layers in the neural network. In contrast to traditional neural networks which only contain 2-3 hidden layers, deep networks can have hundreds of hidden layers. Deep learning models usually need large amount of labelled data to produce quality results. These models can learn features directly from the data without having to manually extract features from the dataset. Therefore, having more data can improve the deep neural network.

Furthermore, deep learning models have shown high levels of accuracy to the point where in some instances like classifying objects in images, deep learning has even outperformed humans. Convolutional Neural Network (CNNs), Long Short Term Memory Networks (LSTMs), Recurrent Neural Networks (RNNs) and Generative Adversarial Networks (GANs) are few example of deep learning algorithms.

But in the real world there’s a lack of large labelled data sets. Due to the huge number of parameters in the networks, training models on such networks with smaller datasets can cause overfitting. Further, the absence of transfer learning was also an issue.

As a solution to these issues the Transformer model was introduced by Google in 2018. Transformer model is the latest addition to deep learning

These models follow “Transfer Learning”, the technique of training a deep learning model on a large dataset which then can be used to perform similar tasks on another dataset. Such models are called “pre-trained models”.

### 3.1.2 BERT

BERT (Bidirectional Encoder Representations from Transformer) is an adaptation of transformer models mentioned above. BERT is known to be a cutting-edge technology in machine learning for Natural Language Processing. (Devlin et al., 2019)

BERT is pre-trained on a large unlabeled data set which includes the entire Wikipedia (2500 million words) and book corpus of 800 million words. Nowadays this model is reused on many applications of downstream tasks like text classification, name-entity recognition, language model fine-tuning, sentence pair tasks and regression and etc.

Previously, language models could only read text inputs sequentially, either from left-to-right or right-to-left. But BERT is “deeply bidirectional” meaning that it learns the text input from both directions. This mechanism helps the model to understand the meaning of ambiguous language.

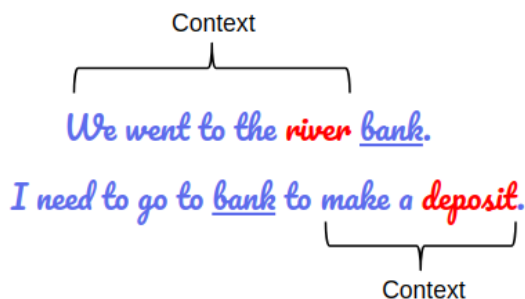


Figure 3.1: BERT captures the context of both left and right

BERT is pre-trained on two NLP tasks:

1. Masked Language Modelling (MLM)  
This is the task of replacing 15% of the words in each sequence with a [MASK] token, then feeding these sequences to the BERT model to predict the hidden/masked words by understanding the context of the other unmasked words in the sequence.
2. Next Sentence Prediction (NSP)

This task understands the relationship between sentences. Given two sentences, BERT model predicts whether the second sentence follows the first or if it's a random sentence.

```
Input = [CLS] the man went to [MASK] store [SEP]
        he bought a gallon [MASK] milk [SEP]
Label = IsNext

Input = [CLS] the man [MASK] to the store [SEP]
        penguin [MASK] are flight ##less birds [SEP]
Label = NotNext
```

Figure 3.2: Example of Next Sentence Prediction

Many versions of BERT have emerged over the last few years. RoBERTa, DistilBERT and ALBERT are few of them. When BERT is fine-tuned on downstream tasks, in most instances, it has proven to produce more accurate results over the other deep learning models.

### 3.1.3 K-Fold Cross Validation

K-Fold cross validation is a resampling technique which is used to ensure the stability of machine learning models.

In K-fold cross validation, the whole data set is divided into k sets which are of equal size. Then first set is selected as the testing set and the model is trained on the remaining k-1 sets. Again, the second set is selected as the testing set and the model is trained on the remaining k-1 sets. Similarly, the process continuous for the rest of the sets.

The importance of this technique is that each data point gets to be in a test set once and k-1 times in a training set. As the value of K increases the variance in test-error also decreases. k=5, k=10 is said to be the most suitable values for k as it shows less biasness and less variance.

## 3.2 Dataset

### 3.2.1 Introduction

According to literature, there can be three ways to find a depression related dataset for research purposes.

1. Using crowdsourcing method

A survey needs to be conducted based on a screening tool (CES-D, PHQ-9 etc) and collect the diagnosis dates from the users who stated that they have been depressed. Then select the eligible users and get access to their social media accounts to extract posts for further analysis. This method is more time consuming and the results will be based on self-reported cases, meaning these users themselves claimed that they have been diagnosed with depression. So the reliability of the user information is debatable. (Choudhury et al., 2013; De Choudhury et al., 2014)

2. Running searches on social media platforms

Researchers can either use filter words (#depres, #anxiety, #abuse, etc.) to select posts and retrieve data using relevant social media APIs, or they can search for communities related to depression on such platforms and collect posts of community members. (Owen et al., 2020) This method allows the researchers to find self-expressions of depression diagnoses and collect data from users who maybe even unaware of their own mental state. Finally, the resulting dataset needs to be manually annotated by professionals in the field of study.

3. Using a publicly available dataset

In this study, we have used a publicly available dataset which was initially created using the above mentioned second method.

### **3.2.2 Data**

‘eRisk2018’ is a research collection published by CLEF workshop. (Losada and Crestani, 2016) We gained access to the dataset by signing a user agreement with the owners of dataset. A copy of the user agreement is given in Appendix A.

The collection consists of 2 tasks (task1: depressed and non-depressed users; task2: anorexia and non-anorexia users). We considered only task 1 which is aimed towards early detection of signs of depression.

eRisk2018 dataset is created from the Reddit platform, particularly from depression subreddits. Subreddits are communities where redditors can submit content such as posts, comments or direct links on a specific subject of interest.

The owners of the dataset have collected data for depressed samples by running searches on subreddits. Then the posts have been manually reviewed to verify that they were real and acceptable. To avoid self-reported cases, expression like “I have depression”, “I think I have depression”, “I am depressed” have been removed from the dataset.

Moreover, large number of random redditors were selected as non-depressed samples. Few numbers of users who were active on depression subreddit but did not indicate any depression symptom were also included in this set.

For each user, the time period considered from first submission to last submission is approximately a year.

### **3.2.3 Structure**

The ‘eRisk2018’ for task 1: early detection of signs of depression provides a dataset which is divided into 10 chunks and ground truth of the corpus. The file structure is shown in Figure 3.

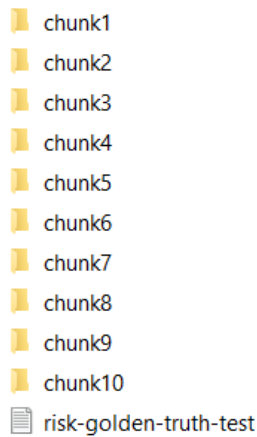


Figure 3.3: File Structure of eRisk2018 Dataset

The posts from users are ordered in chronologically and divided into 10 chunks. Each chunk contains a set of XML files per user. The first chunk holds the oldest 10% of the submissions, the second chunk next 10% of the submission and so forth. Each user is identified by the subject id.

For example, if subject314 had posted 200 posts over a year’s period of time, then all 200 posts were chronologically ordered, and first 20 posts were put into first chunk against its subject id, subsequently second 20 posts were put into the second chunk and so on. Hence, each chunk contains 10% of the total number of posts.

```
<INDIVIDUAL>
<ID> ... </ID>
<WRITING>
<TITLE> ... </TITLE>
<DATE> ... </DATE>
<INFO> ... </INFO>
<TEXT> ... </TEXT>
</WRITING>
<WRITING>
<TITLE> ... </TITLE>
<DATE> ... </DATE>
<INFO> ... </INFO>
<TEXT> ... </TEXT>
</WRITING>
....
</INDIVIDUAL>
```

Figure 3.4: XML File Format

Each XML file (per subject) contains

- ID - anonymous id
- TITLE - title of the post if available
- DATE – posted date
- INFO – type of content (post or comment)
- TEXT – body of the post or comment

```
<INDIVIDUAL>
  <ID>subject513</ID>
  - <WRITING>
    <TITLE> </TITLE>
    <DATE> 2017-04-16 16:59:36 </DATE>
    <INFO> reddit post </INFO>
    <TEXT> So why not stop playing if you hate it so much? </TEXT>
  </WRITING>
  - <WRITING>
    <TITLE> </TITLE>
    <DATE> 2017-04-16 16:45:29 </DATE>
    <INFO> reddit post </INFO>
    <TEXT> Lol whats entertaining is all you people crying on a subreddit about how unfair it is that dice is appealing to more gamers than you in your ivory towers. </TEXT>
  </WRITING>
  - <WRITING>
    <TITLE> </TITLE>
    <DATE> 2017-04-16 16:36:54 </DATE>
    <INFO> reddit post </INFO>
    <TEXT> People that whine about the automatico need to realize its supposed to be good up close. But nope, they'd rather cry to dice about the things that kill them downvote me I don't care. Someone has to call out the whining. </TEXT>
  </WRITING>
```

Figure 3.5: XML File Example

### 3.2.4 Summary

The ground truth contains total of 820 users with 79 labelled as 1 (depressed users) and 741 users as 0 (non-depressed users). The total number of writings from depressed users were 40,665 and non-depressed users were 503,782.

The average period of time from first submission of a user to the last submission covers approximately a year.

Table 3.1: Summary of eRisk 2018 dataset

	<i>Depressed</i>	<i>Non-Depressed</i>	<i>Total</i>
<i>Users</i>	79	741	820
<i>Writings</i>	40,665	503,782	544,447

### 3.2.5 Process

The dataset is divided into 10 chunks in order to support the early detection task. Therefore, during the testing stage, chunk by chunk is released. For example, once the first chunk is released, our model has to make one of the following decisions,

- 1 – user is depressed
- 0 – user is non-depressed
- 2 – no decision, waiting for more chunks to make the decision

Then the second chunk will be released. Then the third chunk and so forth.

Once the model emits the decision 1 or 0, then it is considered as the final decision. So that we can evaluate how early the decision is emitted by the model. (Losada et al., 2018)

### 3.3 Methodology

This study is focused on building a deep learning model to identify the users with depression during early stages.

After data preparation, we fine-tuned a pre-trained BERT model using the eRisk2018 dataset. Then we used this language model created from our eRisk2018 dataset and fine-tuned again for the prediction task.

As explained in 3.2.5, chunk by chunk was released to predict results for each user.

We used 5-fold cross validation to improve the accuracy. Therefore, we merged all the results from 5 folds and calculated the ERDE metric for each chunk.

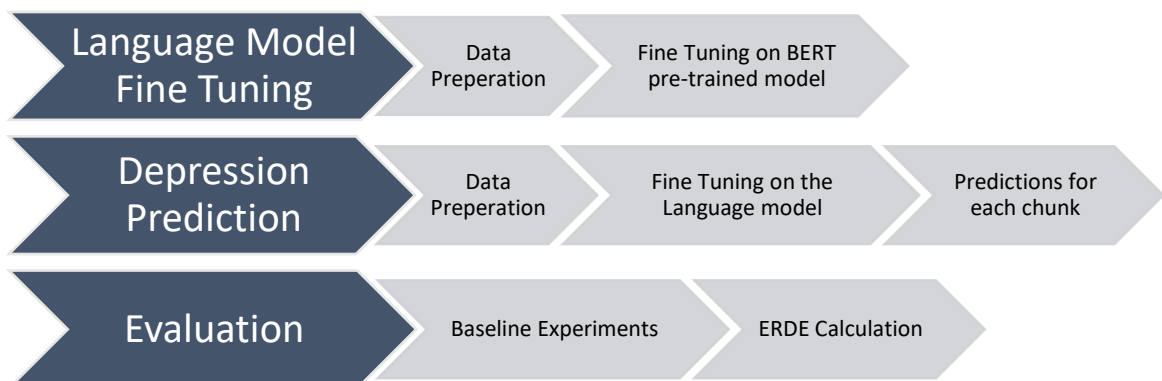


Figure 3.6: Overview of the methodology



### 3.3.1 Data Preparation

The ground-truth corpus was split into two sets: training set 80% and test set 20%

We used 5-fold cross validation to ensure less biasness in the dataset. Therefore, the dataset was divided into 5 splits where each split had following number of depressed and non-depressed users.

Table 3.2: Summary of data in each split

	<i>Training Set</i>	<i>Test Set</i>
<i>Depressed</i>	63	16
<i>Non-Depressed</i>	593	148
<i>Total</i>	656	164

BERT model takes text sequences as inputs. For each split data was prepared for the language model and classification model separately.

#### 3.3.1.1 Language Model

Language Model requires only texts. Labels were removed from the dataset and posts of all the users in the training set were merged into a file. Similarly, the test file was created to evaluate the Language Model.

```
Why Skyrim is my GOTY. *Lots of SPOILERS*I hated the Imperials. I had never played a Elder Scr
One of my first thoughts after being freed from my bonds was to teach these vile dictators a le
I was wrong of course. Bringing Skyrim back to the Nords meant freeing it from these immigrant
When I arrived, I presented the axe to Balgruuf, hoping to see him to come to a conclusion that
As I made my way back, I thought of all the options my character would have to stop this push f
The battle was ferocious, my character slaying man after man of the Whiterun guard, men pleadin
During the next few campaigns I began to embrace the warrior's life with more vigil. Ralof was
Skyrim is many things. A game in which you battle Dragons, Giants and Bandits. It has a simpl
Later, I was imprisoned within Markarth on a mission to bribe a Squire to sympathize with our c
By Talos, what have I done...
TIL that there was special unit of the soviet police that investigated and battled acts of cann
This is freaky now. Imagine seeing this in 1979.
TIL that Gladiator had a planned sequel and a script written by Nick Cave. But the story was s
Can we accuse Westboro Baptist Church of child abuse as a way to combat their protests?Ya know,
```

Figure 3.7: Language Model data format

### 3.3.1.2 Classification Model

For Classification Model, data needs to be arranged in a *Python* dataframe as follows. This was prepared for both training and test sets.

Text – All posts from all chunks of a particular user

Label – Risk label of the user

	text	label
0	Ah I was told to it here but ill move it\nEdit...	1
1	Moving to Omaha Any Advice?I live in NYC (the ...	1
2	I look at this comic whenever I need a good ch...	0
3	The Witch DON'T BREATHE - First International ...	0
4	How old is he? I can't imagine that process wo...	0

Figure 3.8: Classification Model data format

### 3.3.2 Language Model Fine-Tuning

There are four types of pre-trained models of BERT. We have used BERT-Base: 12-layer, 768-hidden-nodes, 12-attention-heads, 110M parameters. We used the “bert-base-uncase” pre-trained model which has the same configurations to fine-tune our language model.

The reason that we didn’t directly use the “bert-base-uncase” pre-trained model for the classification task is because sometimes, pre-trained language models can be less effective on data which has a highly specialized language such as health care data. Therefore, we fine-tuned the “bert-base-uncase” pre trained model on our dataset. The source code for creating a language model is given in Appendix B.

We used “Simple Transformers” library for BERT modelling.

The hyperparameters for the language model are as follows.

Table 3.3: Language Model hyperparameters

<i>Parameter</i>	<i>BERT</i>
Learning Rate	2e-5
No. of Training Epochs	3
Maximum Sequence Length	512
Batch Size	8
Warm up steps	400

It was advised to train the model for 2 to 4 number of epochs. Therefore, we fine-tuned the model for 3 epochs. After tweaking the hyperparameters for better results, the model seemed to perform well with the above given values. The learning rate and batch size can make a huge difference to training results. Also, this process consumed high power and long training times.

We fine-tuned language models for each split.

### 3.3.3 Fine-Tuning BERT for Text Classification

We used the language model that we previously fine-tuned on our dataset for the downstream task of binary classification. The source code for creating a language model is given in Appendix C.

The hyperparameters of the classification model are as follows.

Table 3.4: Classification Model hyperparameters

<i>Parameter</i>	<i>BERT</i>
Learning Rate	1e-5
No. of Training Epochs	10
Maximum Sequence Length	256
Batch Size	8
Warm up steps	10

After fine-tuning the language model for the classification task, we realized the model is suffering from overfitting. The evaluation loss was increasing from the start. This is due to the highly imbalanced dataset.

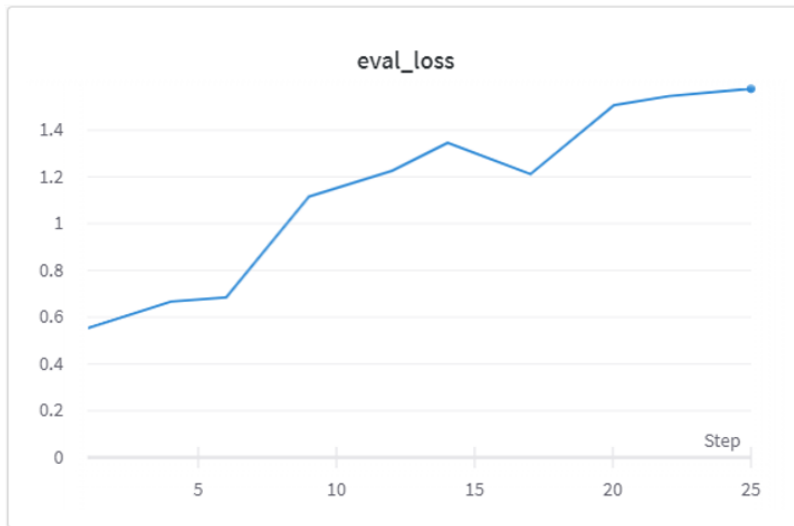


Figure 3.9: Evaluation Loss of Classification Model

After reducing the learning rate to  $1e-5$  the model performed somewhat better. But to reduce the evaluation loss even more, we added class weights to the model. We calculated the class weights using “Sklearn” library in python and the values were [0.55311973, 5.20634921]

Still the model was suffering from overfitting. Since the “SimpleTransformers” library did not support adding a dropout to the model, we couldn’t use the Dropout regularization method to reduce the loss.

We fine-tuned classification models for each split.

### 3.3.4 Prediction

As explained in 3.2.5. when the chunks are fed to the system one by one, model has to predict the user is either depressed, non-depressed or no decision (waiting for more chunks to decide)

Since our classification model is favoring negative samples, due to the imbalanced dataset and overfitting, we calculated an optimal threshold value for positive samples using ROC curve, before starting the prediction task. See Appendix D for the source code.

The prediction solution suggested by this study is as follows.

threshold = 0.75

For each chunk

    Get model outputs (model predictions)

    Calculate optimal threshold values for positive samples

    For each user

        If (model output for positives > optimal threshold)

            If (model output > threshold)

                Set 1

            Else

                Set 2

        Else

            If (model output > threshold)

                Set 0

            Else

                Set 2

We used a threshold value = 0.75 to decide whether to wait for more chunks or confirm the decision.

1 = Depressed, 0= No decision and 2 = Non-Depressed

Once the prediction is completed, we saved the results of each chunk into a file.

Each file contains the Subject ID, Label and Delay (No. of posts seen)

BERT_1	subject3642	0	197
BERT_2	subject7785	2	138
BERT_3	subject7891	0	31
BERT_4	subject6928	1	1
BERT_5	subject845	0	6
BERT_5	subject5161	1	1
BERT_6	subject1093	2	173
BERT_6	subject7770	2	6
BERT_7	subject1010	0	127
BERT_8	subject7831	0	23
BERT_9	subject8993	2	101
BERT_9	subject9825	0	153
BERT_10	subject4719	0	30
BERT_10	subject5156	0	126

Figure 3.11: Prediction results file format

Figure 3.10: Prediction result file content

## CHAPTER 4

### EVALUATION AND RESULTS

This chapter explains the evaluation plan of the research and reveals the results of our prediction model.

#### 4.1 Evaluation

In this study, I have evaluated both the language model and the classification model. Language model was evaluated using perplexity and the classification model was evaluated against two metrics: Performance metrics and the ERDE metric that was introduced by the eRisk2018 organizers to reward early diagnosis of depression.

##### 4.1.1 Perplexity

Most commonly used evaluation metric for language models is perplexity.

If there is a tokenized sequence  $X = (x_0, x_1, \dots, x_t)$  then the perplexity is defined by

$$\text{PPL}(X) = \exp \left\{ -\frac{1}{t} \sum_i^t \log p_{\theta}(x_i | x_{<i}) \right\}$$

Perplexity of a language model can be described as the uncertainty level for predicting the following symbol. Thus, lower the perplexity better the language model is. But for masked language models like BERT, perplexity is not the best evaluation metric.

##### 4.1.2 Performance Metrics

To evaluate the classification model performance, we applied standard metrics widely used for classification problems.

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

$$\text{Precision} = \frac{TP}{TP+FP}$$

$$Recall = \frac{TP}{TP+FN}$$

$$F1\ Score = 2 \times \frac{Precision*Recall}{Precision+Recall}$$

where:

TP – samples which model predicts as positive and are actually positive

TN – samples which model predicts as negative and are actually negative

FP – samples which model predicts as positive but are actually negative

FN – samples which model predicts as negative but are actually positive

### 4.1.3 ERDE Metric

ERDE stands for Early Risk Detection Error which rewards early detection of positive samples of a system. eRisk2018 organizers introduced ERDE metric due to the reason that standard classification metrics are time unaware.

This measure takes into account the correctness of the decision and delay in taking the decision. An early risk detection system should process the texts in order they were created and detect the risk samples as soon as possible. So that, the system can monitor social media evidence as and when they appear online. In real world domain, data can be highly imbalanced, therefore different predictions were given different weights.

$$ERDE_o(d, k) = \begin{cases} c_{fp} & \text{if } d = \text{positive AND ground truth} = \text{negative (FP)} \\ c_{fn} & \text{if } d = \text{negative AND ground truth} = \text{positive (FN)} \\ lc_o(k) \cdot c_{tp} & \text{if } d = \text{positive AND ground truth} = \text{positive (TP)} \\ 0 & \text{if } d = \text{negative AND ground truth} = \text{negative (TN)} \end{cases}$$

where:  $d = \text{binary decision taken by the system}$

$$c_{fn} = 1$$

$c_{fp} = \text{proportion of positive cases in the collection}$

$k = \text{delay (no of submissions seen before taking the decision)}$

$$lc_o(k) = 1 - \frac{1}{1+e^{k-o}}$$

$$c_{t_p} = 1$$

$$o = 5 \text{ for } ERDE_5 \text{ or } 50 \text{ for } ERDE_{50}$$

For each user, ERDE was calculated and the mean of all the values is taken as the final score.

In  $ERDE_5$  penalties grow after the first 5 submissions from the user. This metric is very sensitive to delays. In  $ERDE_{50}$  penalties grow after first 50 submissions.

Late detection of a positive case, i.e. depressed user, will be penalized. But late detection of a negative case, i.e. non depressed user, is not an issue for the system. Therefore, this metric measures the early detection of risk cases and detection of non-risk cases.

According to ERDE metric if a system takes fewer submissions to make the correct decision, then the system is better. Thus lower the ERDE value is better the system.

See Appendix E for the source code.

## 4.2 Results

### 4.2.1 Perplexity

The language model evaluation metric used in SimpleTransformers for BERT is the perplexity value. But as mentioned in 4.1.1, it is not the best metric to evaluate BERT language models. The output values for each split are given below.

Table 4.1: Language Model Output

	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>
<i>Perplexity</i>	7.6486	8.3565	8.4897	8.5396	8.0598

These values are comparatively low and better and shows that the uncertainty is low.



## 4.2.2 Performance Metrics

The evaluation results of the Classification Model are as follows.

Table 4.2: Performance metrics for each split

	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>
<i>Accuracy</i>	0.8476	0.8720	0.8841	0.8841	0.8780
<i>F1 Score</i>	0.1935	0.3636	0.3871	0.3871	0.2857
<i>Precision</i>	0.2	0.3529	0.4	0.4	0.3333
<i>Recall</i>	0.1875	0.3750	0.3750	0.3750	0.25

Due to the imbalanced dataset and overfitting of the model, F1 scores are very low. Similarly, Precision and Recall values are low. Accuracy is considerably high due to the large number of negative samples.

## 4.2.3 ERDE

To calculate the overall ERDE value in each chunk, we merged results from 5 folds and then calculated the ERDE.

Given below is the comparison of our results with one of the best performing models submitted for eRisk 2018 workshop. UPFA model has the lowest and best  $ERDE_{50}$  measure which is 6.41%

Table 4.3: ERDE metric for all chunks

	<i>1</i>	<i>2</i>	<i>3</i>	<i>4</i>	<i>5</i>	<i>6</i>	<i>7</i>	<i>8</i>	<i>9</i>	<i>10</i>
<i>ERDE<sub>5</sub>-BERT</i>	10.03%	10.03%	10.01%	9.85%	9.55%	9.31%	9.31%	9.31%	9.29%	<b>9.29%</b>
<i>ERDE<sub>50</sub>-BERT</i>	8.91%	8.26%	8.25%	8.05%	7.93%	7.93%	7.84%	7.74%	7.61%	<b>7.56%</b>
<i>ERDE<sub>5</sub>-UPFA</i>	9.26%	9.04%	9.04%	9.05%	9.06%	9.07%	9.08%	9.11%	9.11%	9.11%
<i>ERDE<sub>50</sub>-UPFA</i>	7.99%	7.16%	7.04%	6.72%	6.73%	6.62%	6.63%	6.65%	6.65%	6.41%

Compared to the UPFA models, our BERT models have produced moderate results. But when considering the F1 scores of the Classification Model, there was a considerable improvement in ERDE values. This could be due to the use of optimal threshold value for prediction. Threshold values have lowered the ERDE value. Our best ERDE value was obtained using ERDE<sub>50</sub> with 7.56% value.

## CHAPTER 5

### CONCLUSION AND FUTURE WORK

This chapter briefly describes the conclusion of our study and the future work that can be done to improve the results.

#### 5.1 Conclusion

In this paper, we have presented an experimental analysis of the eRisk2018 dataset collected from Reddit platform. We have developed a BERT model using the dataset which is highly imbalanced. We fine-tuned the “bert-base-uncase” pretrained model on our dataset and used that language model for the task of binary classification. Since our model was suffering from overfitting, we introduced two threshold values for prediction. The optimal threshold value was calculated using ROC curve for each chunk. The global threshold was a fixed value set to 0.75. As per the eRisk instructions, when we fed the system with each chunk in chronological order, and evaluated the system with ERDE metric, it produced moderate results.

The real-world data related to depression is also highly imbalanced. Therefore, this dataset is an ideal sample to train a model for early detection of depression.

#### 5.2 Future Work

In future work, we intend to utilize more training time when fine-tuning models and analyze the results. We can also train our data on other BERT models like RoBERTa, ALBERT and DistilBERT to see which model performs the best on this dataset.



# APPENDICES

## APPENDIX A: eRisk2018 User Agreement

Individual Application  
to use the

### eRisk 2018 Text Research Collection

I, A.T.L. Harini Yasodhya Samarasinghe, a person engaging in research and development of University of Colombo School of Computing (UCSC), Sri Lanka, and a member of, consultant to, or person providing service to the following organization:

Organization University of Colombo School of Computing (UCSC), Sri Lanka  
Corporation/Partnership/Legal Entity University Of Colombo (UOC), Sri Lanka  
Official mail address University of Colombo School of Computing  
UCSC Building Complex,  
35, Reid Avenue, Colombo 7, SRI LANKA  
Telephone +94 -11-2581245/ 7  
Facsimile +94 -11-2587239  
Electronic mail info@ucsc.cmb.ac.lk

apply(ies) to use the eRisk 2018 Text Research Collection subject to the following understandings, terms and conditions. These understandings, terms and conditions apply equally to all or to part of the information.

#### Permitted Uses

1. The information may only be used for research purposes. Portions of the data maybe copyrighted, and may also have commercial value as data, so you must be careful to use it only for research purposes.
2. Summaries, analyses and interpretations of the linguistic properties of the information may be derived and published, provided it is not possible to reconstruct the information from these summaries.
3. You may not try to identify the individuals whose texts are included into this dataset. You may not cross-reference individuals with the dataset against any other dataset or collection of data. You may not try to establish any kind of contact with the individuals of this dataset.
4. You are not permitted to publish any portion of the dataset (e.g. example post) other than summary statistics, or share it with anyone else.
5. We grant you the right to access the collection's content in the manner described in this agreement. You may not otherwise make unauthorized commercial use of, reproduce, prepare derivative works, distribute copies, perform, or publicly display the collection or parts of it.
6. You may present research findings concerning knowledge obtained using the collection provided that the aforementioned presentation is within the limits of this agreement. Any scientific publication derived from the use of this collection should explicitly refer to:

Losada D.E., Crestani F. A Test Collection for Research on Depression and Language Use. Conference and Labs of the Evaluation Forum (CLEF), Évora, Portugal, 2016.

Losada D.E., Crestani F., Parapar J. (2018) Overview of eRisk: Early Risk Prediction on the Internet. In: Bellot P. et al. (eds) Experimental IR Meets Multilinguality, Multimodality, and Interaction. CLEF 2018. Lecture Notes in Computer Science, vol 11018. Springer, Cham

7. You shall not use results obtained through the use of the collection for profitable purposes including advertisement and/or defamatory or slanderous purposes.
8. If we or the copyright holders request you to discontinue the use of the collection, or your use of the collection is deemed to be in violation of this agreement, you shall immediately discontinue use of the collection and promptly delete the collection and all data obtained by processing it from any computer or media onto which it has been copied.

#### Copyright

1. The copyright holders retain ownership and reserve all rights pertaining to the use and distribution of the information.
2. Except as specifically permitted above and as necessary to use and maintain the integrity of the information on computers; the display, reproduction, transmission, distribution or publication of the information is strictly prohibited. Violations of the copyright restrictions on the information may result in legal liability.
3. Copyright holders of the information contained in the collection include a wide variety of online Internet users.

#### By the Individual:

Signature  \_\_\_\_\_

Date 2021-06-08

Name (*please print*) A.T.L. Harini Yasodhya Samarasinghe

Title Research Student

## APPENDIX B: Source Code of the Language Model

```
from simpletransformers.language_modeling import LanguageModelingModel
import logging
import configparser
from eriskhandler.handler import Task
configs = configparser.ConfigParser()
configs.read('config.cfg')
```

```
logging.basicConfig(level=logging.INFO)
transformers_logger = logging.getLogger("transformers")
transformers_logger.setLevel(logging.WARNING)
```

```
#Prepare data
task = Task(configs['Data']['task'])
split = configs['Data']['split']
```

```
train_args = {
    "output_dir": "outputs/" + split,
    "cache_dir": "cache/" + split,
    "num_train_epochs": 3,
    "train_batch_size": 8,
    "max_seq_length": 512,
    "learning_rate": 2e-5,
    "do_lower_case": True,
    "warmup_steps": 400,
    "reprocess_input_data": True,
    "overwrite_output_dir": True,
}
modelpath = 'bert-base-uncase'
model = LanguageModelingModel('bert', modelpath , args=train_args)

trainfile = "posts/" +split + "/train.txt"
evalfile = "posts/" +split + "/test.txt"
model.train_model(trainfile, eval_file= evalfile)
```

```
model.eval_model(evalfile)
```

## APPENDIX C: Source Code of the Classification Model

```
from simpletransformers.classification import ClassificationModel
import pandas as pd
import sklearn
import logging
import configparser
from eriskhandler.handler import Task
configs = configparser.ConfigParser()
configs.read('config.cfg')
```

```
logging.basicConfig(level=logging.INFO)
transformers_logger = logging.getLogger("transformers")
transformers_logger.setLevel(logging.WARNING)
```

```
#Prepare data
task = Task(configs['Data']['task'])
split = configs['Data']['split']
```

```
#TRAINING SET
train_set = task.get_split(split, part='train', chunks=10)
#subject ids, lables, user posts
train_ids, train_labels, train_users = map(list, zip(*train_set))
print(len(train_users))
```

```
train_users = [' '.join(user) for user in train_users]
train_labels = list(map(int, train_labels))
```

```
train_df = pd.DataFrame(
    {'text': train_users,
     'label': train_labels
    })
train_df.head()
```

```
train_df['label'].value_counts()
```

```
test_set = task.get_split(split, part='test', chunks=10)
#subject ids, lables, user posts
test_ids, test_labels, test_users = map(list, zip(*test_set))
print(len(test_users))
```

```
test_users = [' '.join(user) for user in test_users]
test_labels = list(map(int, test_labels))
```

```
test_df = pd.DataFrame(
    {'text': test_users,
     'label': test_labels
    })
```

```
test_df.head()
```

```
test_df['label'].value_counts()
```

```
#create a ClassificationModel
```

```
train_args = {
    "output_dir": "outputs_bert/" + split,
    "cache_dir": "cache_bert/" + split,
    "num_train_epochs": 10,
    "train_batch_size": 8,
    "max_seq_length": 512,
    "learning_rate": 1e-5,
    "do_lower_case": True,
    "warmup_steps": 10,
    "wandb_project": "BERT-eRisk-a",
    "evaluate_during_training": True,
    "reprocess_input_data": True,
    "overwrite_output_dir": True,
}
modelpath = 'outputs/'+split
model = ClassificationModel('bert', modelpath , args=train_args , weight = [0.55311973,5.20634921])
```

```
model.train_model(train_df, eval_df = test_df)
```

```
# evaluate model
```

```
model.eval_model(test_df,acc=sklearn.metrics.accuracy_score, f1=sklearn.metrics.f1_score)
```

```
#predict
```

```
predictions, raw_outputs = model.predict(test_users)
```

```
from sklearn.metrics import classification_report, confusion_matrix
print(classification_report(test_labels, predictions))
print(confusion_matrix(test_labels, predictions))
```



## APPENDIX D: Source Code for Prediction

```
from simpletransformers.classification import ClassificationModel
from scipy.special import softmax
import pandas as pd
import sklearn
import logging
import configparser
import os
from os.path import join
import sys

from eriskhandler.handler import Task
configs = configparser.ConfigParser()
configs.read('config.cfg')
```

```
logging.basicConfig(level=logging.INFO)
transformers_logger = logging.getLogger("transformers")
transformers_logger.setLevel(logging.WARNING)
```

```
#Prepare data
task = Task(configs['Data']['task'])
split = configs['Data']['split']

model_dir = 'models/'+split
pred_dir = 'results/'+split
threshold = float(configs['Model']['threshold'])

model = ClassificationModel('bert', model_dir)
```

```
from sklearn.metrics import roc_curve
from matplotlib import pyplot
from numpy import sqrt
from numpy import argmax
```

```

for chunk in range(1, 11): #number of chunks to predict for
    print(chunk)

    #Prepare test data
    test_set = task.get_split(split, part='test', chunks=chunk)
    test_ids, test_labels, test_users = map(list, zip(*test_set))
    user_posts = list(map(len, test_users)) #posts
    test_users = [' '.join(user) for user in test_users]
    test_labels = list(map(int, test_labels))

    predictions, raw_outputs = model.predict(test_users)
    probabilities = softmax(raw_outputs, axis=1)

    #Calculate threshold

    # Calculate roc curves
    fpr, tpr, thresholds = roc_curve(test_labels, probabilities[:, 1])
    # Calculate the g-mean for each threshold
    gmeans = sqrt(tpr * (1-fpr))
    # Locate the index of the largest g-mean
    max_gmean = argmax(gmeans)
    print('Best Threshold=%f, G-Mean=%.3f' % (thresholds[max_gmean], gmeans[max_gmean]))

    opt_threshold = thresholds[max_gmean]

    with open(os.path.join(pred_dir, 'BERT_{}.txt').format(chunk), 'w') as f:
        for uid, prob, num_of_posts in zip(test_ids, probabilities, user_posts): # ids, results , posts
            value = 2
            if prob[1] >= opt_threshold:
                if prob[1] >= threshold:
                    value = 1
                else:
                    value = 2

            else:
                if prob[0] >= threshold:
                    value = 0
                else:
                    value = 2
            f.write('{}\t{}\t{}\n'.format(uid, value ,num_of_posts)) # decision made
        f.close()

```

## APPENDIX E: Source Code for ERDE Calculation

```
data_golden = pd.read_csv(goldenTruth_path, sep="\t", header=None, names=['subj_id','true_risk'])
data_result = pd.read_csv(algorithmResult_path, sep=" ", header=None, names=['subj_id','risk_decision','delay'])

# Merge tables (data) on common field 'subj_id' to compare the true risk and the decision risk
merged_data = data_golden.merge(data_result, on='subj_id', how='left')

# Add column to store the individual ERDE of each user
merged_data.insert(loc=len(merged_data.columns), column='erde',value=1.0)

# Variables
risk_d = merged_data['risk_decision']
t_risk = merged_data['true_risk']
k = merged_data['delay']
erde = merged_data['erde']

# Count of how many true positives there are
true_pos = len(merged_data[t_risk==1])

# Count of how many positive cases the system decided there were
pos_decisions = len(merged_data[risk_d==1])

# Count of how many of them are actually true positive cases
pos_hits = len(merged_data[(t_risk==1) & (risk_d==1)])

# Total count of users
total_users = len(merged_data)

# ERDE calculus
for i in range(total_users):
    if(risk_d[i] == 1 and t_risk[i] == 0):
        erde.ix[i] = float(true_pos)/total_users
    elif(risk_d[i] == 0 and t_risk[i] == 1):
        erde.ix[i] = 1.0
    elif(risk_d[i] == 1 and t_risk[i] == 1):
        erde.ix[i] = 1.0 - (1.0/(1.0+np.exp(k[i]-o)))
    elif(risk_d[i] == 0 and t_risk[i] == 0):
        erde.ix[i] = 0.0

# Calculus of F1, Precision, Recall and global ERDE
precision = float(pos_hits)/pos_decisions
recall = float(pos_hits)/true_pos
F1 = 2 * (precision * recall) / (precision + recall)
erde_global = erde.mean() * 100

indiv_erde = merged_data.ix[:,['subj_id','erde']]
print indiv_erde.to_string()
print 'Global ERDE (with o = %d): %.2f' % (o, erde_global), '%'
```

## REFERENCES

- Bucur, A.-M., Dinu, L.P., 2020. Detecting Early Onset of Depression from Social Media Text using Learned Confidence Scores. ArXiv201101695 Cs Stat.
- CACHEDA, F., FERNANDEZ, D., NOVOA, F.J., CARNEIRO, V., 2019. Early Detection of Depression: Social Network Analysis and Random Forest Techniques. *J. Med. Internet Res.* 21, e12554. <https://doi.org/10.2196/12554>
- Choudhury, M.D., Gamon, M., Counts, S., Horvitz, E., 2013. Predicting Depression via Social Media 10.
- CLEF eRisk: Early risk prediction on the Internet | CLEF 2018 workshop [WWW Document], n.d. URL <https://early.irlab.org/2018/index.html> (accessed 9.12.21).
- Coppersmith, G., Dredze, M., Harman, C., Hollingshead, K., 2015. From ADHD to SAD: Analyzing the Language of Mental Health on Twitter through Self-Reported Diagnoses. Presented at the Proceedings of the 2nd Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality, pp. 1–10. <https://doi.org/10.3115/v1/W15-1201>
- De Choudhury, M., Counts, S., Horvitz, E., 2013. Social media as a measurement tool of depression in populations, in: Proceedings of the 5th Annual ACM Web Science Conference, WebSci '13. Association for Computing Machinery, New York, NY, USA, pp. 47–56. <https://doi.org/10.1145/2464464.2464480>
- De Choudhury, M., Counts, S., Horvitz, E.J., Hoff, A., 2014. Characterizing and predicting postpartum depression from shared facebook data, in: Proceedings of the 17th ACM Conference on Computer Supported Cooperative Work & Social Computing, CSCW '14. Association for Computing Machinery, New York, NY, USA, pp. 626–638. <https://doi.org/10.1145/2531602.2531675>
- Devlin, J., Chang, M.-W., Lee, K., Toutanova, K., 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. ArXiv181004805 Cs.
- Fatima, I., Mukhtar, H., Ahmad, H., Rajpoot, K., 2017. Analysis of user-generated content from online social communities to characterise and predict depression degree. *J. Inf. Sci.* 44, 016555151774083. <https://doi.org/10.1177/0165551517740835>
- González-Carvajal, S., Garrido-Merchán, E.C., 2021. Comparing BERT against traditional machine learning text classification. ArXiv200513012 Cs Stat.

- Losada, D.E., Crestani, F., 2016. A Test Collection for Research on Depression and Language Use, in: Fuhr, N., Quaresma, P., Gonçalves, T., Larsen, B., Balog, K., Macdonald, C., Cappellato, L., Ferro, N. (Eds.), *Experimental IR Meets*
- Multilinguality, Multimodality, and Interaction, *Lecture Notes in Computer Science*. Springer International Publishing, Cham, pp. 28–39. [https://doi.org/10.1007/978-3-319-44564-9\\_3](https://doi.org/10.1007/978-3-319-44564-9_3)
- Losada, D.E., Crestani, F., Parapar, J., 2018. Overview of eRisk 2018: Early Risk Prediction on the Internet (extended lab overview) 20.
- Maslej-Krešňáková, V., Sarnovský, M., Butka, P., Machová, K., 2020. Comparison of Deep Learning Models and Various Text Pre-Processing Techniques for the Toxic Comments Classification. *Appl. Sci.* 10, 8631. <https://doi.org/10.3390/app10238631>
- Maupome, D., Meurs, M.-J., 2019. Using Topic Extraction on Social Media Content for the Early Detection of Depression 5.
- Orabi, A.H., Buddhitha, P., Orabi, M.H., Inkpen, D., 2018. Deep Learning for Depression Detection of Twitter Users. Presented at the Proceedings of the Fifth Workshop on Computational Linguistics and Clinical Psychology: From Keyboard to Clinic, pp. 88–97. <https://doi.org/10.18653/v1/W18-0609>
- Owen, D., Collados, J.C., Espinosa-Anke, L., 2020. Towards Preemptive Detection of Depression and Anxiety in Twitter. *ArXiv201105249 Cs*.
- Ramírez-Cifuentes, D., Freire, A., 2018. UPF's Participation at the CLEF eRisk 2018: Early Risk Prediction on the Internet.
- Reece, A., Reagan, A., Lix, K., Dodds, P., Danforth, C., Langer, E., 2017. Forecasting the onset and course of mental illness with Twitter data. *Sci. Rep.* 7. <https://doi.org/10.1038/s41598-017-12961-9>
- Resnik, P., Armstrong, W., Claudino, L., Nguyen, T., Nguyen, V.-A., Boyd-Graber, J., 2015. Beyond LDA: Exploring Supervised Topic Modeling for Depression-Related Language in Twitter. Presented at the Proceedings of the 2nd Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality, pp. 99–107. <https://doi.org/10.3115/v1/W15-1212>
- Schwartz, H.A., Eichstaedt, J., Kern, M., Park, G., Sap, M., Stillwell, D., Kosinski, M., Ungar, L., 2014. Towards Assessing Changes in Degree of Depression through Facebook. Presented at the Proceedings of the Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality, pp. 118–125. <https://doi.org/10.3115/v1/W14-3214>

- Singh, D., Wang, A., n.d. Detecting Depression Through Tweets.
- Stephen, J., P., P., 2019. Detecting the magnitude of depression in Twitter users using sentiment analysis. *Int. J. Electr. Comput. Eng. IJECE* 9, 3247.  
<https://doi.org/10.11591/ijece.v9i4.pp3247-3255>
- Wald, R., Khoshgoftaar, T., Sumner, C., 2012. Predicting Depression via Social Media. *Proc. 2012 IEEE 13th Int. Conf. Inf. Reuse Integr. IRI 2012* 2, 109–115.  
<https://doi.org/10.1109/IRI.2012.6302998>

