COMMON N-GRAM METHOD: A PROMISING APPROACH TO DETECTING MENTAL HEALTH DISORDERS ON SOCIAL MEDIA

by

Harshit Agarwal

Submitted in partial fulfillment of the requirements
for the degree of Master of Computer Science

at

Dalhousie University
Halifax, Nova Scotia
April 2023

This is dedicated to my family.

Contents

List of Tables

List of Figures

Abstract

This paper addresses the mental health challenges posed by the COVID-19 pandemic and the lack of reliable and accessible diagnostic tools for mental health conditions. The dataset used in this research consists of posts from 10,000 users with over 2 million posts categorized into 7 labels from the social media platform called Reddit. To enhance the realism of our model, we created a biased dataset that reflects the real-world ratios of mental illness prevalence. This approach makes our findings more relevant and applicable in real-world settings. The proposed solution is Common N-gram (CNG) method that offers comparable results to the state-of-the-art CNN-LSTM model and is less resource-intensive. The study demonstrates the ability of common N-gram method to classify and differentiate between illnesses in classification tasks. The CNG method shows better performance in comparison to the CNN-LSTM model and SVM, baseline model, in multi-classification tasks. The CNN-LSTM surpasses performance in binary tasks compared to the best score reported in the previous study with the same dataset, with an f1-score consistently higher than 0.60 reported for all classes. The study also highlights the usefulness of the Relative N-Gram Signature method to analyze the classification decision of the common N-gram technique. Furthermore, interpretation of the CNN-LSTM model is accomplished through the use of word embeddings that were generated by the model. Interpretation of the model allows us to test its viability for application in the real world as well as to identify interesting patterns within the text written by individuals with mental health conditions. The proposed solutions offer practical and accessible options for individuals seeking reliable and accurate mental health support. The goal of this research is to contribute towards better mental health outcomes and improve the quality of life for those who are struggling.

List of Abbreviations and Symbols Used

ADHD Attention-Deficit / Hyperactivity Disorder. 1, 22, 39, 44

AI Artificial Intelligence. 7

BERT Bidirectional Encoder Representations from Transformers. 16

CNG Common N-gram. 2, 17, 19, 20, 27, 36, 38, 39, 46–49

CNN Convolutional Neural Network. 6, 8, 11, 12, 25

CNN-LSTM Convolutional neural network and long short-term memory network. 15, 25, 26, 37, 38, 44, 46, 47

COVID-19 Coronavirus Disease of 2019. 1, 15

FN False Negative. 33

IRL Inductive Rule Learning. 7

LSTM Long Short Term Network. 13–15, 25, 26

ML Machine Learning. 6–8, 16, 17, 23, 38

NLP Natural Language Processing. 7, 48

PTSD Post-traumatic stress disorder. 6

RBF Radial Basis Function. 9

RSDD Reddit Self-reported Depression Diagnosis. 22

SMHD Self-Reported Mental Health Diagnoses. 22, 23

SVM Support Vector Machine. 6, 8–10, 25, 38

TP True Positive. 33

Acknowledgements

I am deeply grateful to my loving family and dear friends for their unwavering support during this exciting chapter in my life. A special shout-out to the remarkable Dr. Vlado Keselj, whose guidance and mentorship were invaluable during my Master's studies. Thank you all for being a part of my journey.

Chapter 1

Introduction

1.1 Motivation

The Coronavirus Disease of 2019 (COVID-19) pandemic has presented our mental health with many difficulties. With lockdowns and social distancing practices, people have felt isolated and alienated, leading to an increase in mental health conditions like acute stress disorder, anxiety, and depression [120]. According to the Centers for Disease Control, more than 50% of people in North America have mental health problems at some point in their life [32]. This is a critical issue that demands our attention and despite the growing need for mental health support, access to accurate and reliable diagnostic tools remains limited.

As technology and social media have become more ubiquitous, particularly during the COVID-19 pandemic, people have increasingly relied on them to stay connected with others and express their feelings and opinions. However, this dependence on digital communication also underscores the importance of identifying and addressing mental health issues, as individuals may be more susceptible to mental health conditions due to the lack of face-to-face interaction and other stressors associated with online communication.

Neurodevelopmental disorders and mental illnesses are prevalent conditions that impact individuals of all ages and backgrounds in their daily life, relationships, and overall well-being. This study aims to shed light on the six most common mental illnesses. Attention-Deficit / Hyperactivity Disorder (ADHD) is one of the most common neurodevelopmental disorders of childhood. People with ADHD may have trouble paying attention, controlling impulsive behaviors, or being overly active [33]. Anxiety is a feeling of fear, dread, and uneasiness which can be often triggered by stress [80]. Autism spectrum disorder (ASD) is a developmental disability caused by differences in the brain. People with ASD often have problems with social communication and interaction, and restricted or repetitive behaviours or interests [34].

Bipolar disorder is a mental illness that causes unusual shifts in mood, energy, activity levels, concentration, and the ability to carry out day-to-day tasks [78]. Depression is a common but serious mood disorder that causes a persistent feeling of sadness and loss of interest [16]. Schizophrenia is a mental disorder characterized by disruptions in thought processes, perceptions, emotional responsiveness, and social interactions [79]. Mental illnesses can significantly impact a person's writing and speaking patterns, and gaining an in-depth understanding of the characteristics and effects of these disorders is crucial for improving early identification, management, and treatment options [94, 15]. Therefore, this study aims to provide valuable insights into these mental illnesses and contribute towards achieving better mental health outcomes.

This research discusses novel applications of natural language processing and machine learning techniques which can help develop more effective, accessible, and integrable diagnostic tools for mental health illnesses using text-based data. The objective is to enrich the well-being of people who are dealing with mental health problems and to give people the tools they need to better understand and manage their mental health. The proposed solutions offer comparable results to state-of-the-art systems and are less resource-intensive, making them a practical and accessible option for individuals seeking reliable and accurate mental health support. By implementing these solutions, we hope to contribute to a future where mental health resources are widely available and mental health conditions can be detected and treated with greater efficiency and accuracy.

1.2 Research Problem

The major challenge with creating accessible diagnostic tools for mental health diagnosis is that they are too resource intensive to be deployed on current smartphone devices and the deep learning model's black box nature makes them less reliable to use without professional supervision. We address this problem with Common N-gram (CNG) method which has demonstrated high performance in style-oriented text classification tasks across multiple languages, including English.

Further using relative n-gram signature we can gain more insights into the features that contribute towards final prediction results. These insights help us detect trends and patterns in different illnesses ultimately helping to establish a system that utilizes consistent reasoning similar to that used by professionals in patient diagnosis.

In previous works, data balance between the illness group and control has always been closer to 1 : 1 ratio [26, 95]. Eichstaedta [31] used a ratio of 1:5 for patients with depression and without depression to better simulate real-world depression prevalence and concluded that social media has the potential to diagnose mental illness. We draw inspiration from this approach and mimic the ratio of illness based on real-world prevalence to train the models in a biased manner.

1.3 Contribution

The main contributions of this thesis are:

1. Novel CNG implementation:
   - Implementing efficient authorship attribution method for classification of mental disorder comparing its performance to deep learning methods which were adapted and implemented in this research.
   - Incorporates innovative character n-gram approaches to the problem for the classification task of multiple categories in the space of mental health detection, making the research unique.
2. Preparing and using a biased dataset in the real-world ratios of mental illness to simulate true prevalence:
   - Enables the results to be more representative of the real-world scenario.
   - Increases the realism of the model. A model trained on a dataset that accurately reflects real-world statistics will provide more realistic predictions, making it more useful in real-world applications.
3. Applying relative n-gram signature method for analysis:
   - Improves the interpretability of the model by providing a more meaningful representation of the data.
   - Enhances the ability to detect patterns and correlations in the data, leading to improved understanding and potential breakthroughs.

- Demonstrated utility of relative n-gram signature method and word embeddings in language analysis for disorders like ADHD and depression.

1.4 Thesis Outline

The remaining parts of this thesis are outlined in this section as follows:

Chapter 2 discusses the background and related work on natural language processing, and machine learning techniques, and introduces common N-gram related and relative n-gram signature.

Chapter 3 provides the dataset description, data preprocessing steps, processes, and design of different the models used. It further explains model analysis techniques and model evaluation methods in detail.

Chapter 4 discusses the experimental setup that was utilized to compare the effectiveness of the various models. The models are analyzed for interpretation and results are carefully examined to see which model produces the best performance.

Chapter 5 outlines the main findings of this thesis and its limitations and provides directions for future work.

Chapter 2

Background and Related Work

2.1 Social Media as a Tool for Identifying and Analyzing Mental Health Illness

In the field of psychology, prior research work has been conducted to study the relationship between social media and mental health and how they affect each. Pantic [83] assessed that determining the cause-and-effect relationship between the two can be challenging, but the correlation is substantial and credible. The study specifically investigated the relationship between Facebook usage and symptoms of depression and found that certain online behaviours may serve as predictive indicators in identifying and assessing depression. Other studies [111, 69] have proposed that self-presentation is the main cause of low-self esteem in Facebook users. Kuss and Griffiths [61] have also concluded that overuse of social networking sites can have detrimental effects on relationships, academic performance, and face-to-face social interactions, all of which can be symptoms of possible internet addiction.

The accessibility of widespread language on social media has intrigued researchers studying the linguistic expressions of individuals with mental health conditions [18, 122]. Many studies have turned to Facebook, Twitter, and Reddit for data collection due to the cost and bias issues present in survey-based studies. Twitter's character limit for messages can pose a challenge for deep learning techniques that require substantial amounts of data, as the concise nature of the messages may not allow for adequate information to be captured [21]. Various studies have been conducted to use social media data to identify and analyze specific mental illnesses such as depression [20, 122, 49], schizophrenia [75], post-partum emotions in mothers [25], suicide ideation [29, 60, 18, 27, 51]. Cohan et al. [17] expanded upon the work of Yates et al. [122] and created a more extensive dataset that incorporated nine classes of illnesses and a targeted control group from Reddit.

De Choudhury et al. [26] were successful in identifying themes and features in the text of users with depression and were able to predict the onset of depression prior to its occurrence. This study's identified features were also evident in other languages, including Japanese, showing comparable cross-language trends [110]. Coppersmith et al. [22] compared various approaches for modeling mental health-related language from social media using data from Twitter users with depression or Post-traumatic stress disorder (PTSD) and demographically matched control participants. Three binary classification tasks were performed and average precision was compared for performance.

Garla and Brandt [36, 37] presented novel feature engineering techniques that involved leveraging Unified Medical Language systems to improve text classification (UMLS) using Machine Learning (ML) methods by handling ambiguous terms and feature ranking. Both studies reported improved clinical document classification. Domain knowledge in the classification of medical text has been shown to play a major role in the inductive learning process and performance of the model [102, 117, 121]. The successful implementation of a simple rule-based classifier that removed misleading information from text for semantic classification highlights the unique challenges presented by the medical text. Unlike general English text, medical text requires specialized knowledge to not only enhance performance but also fully utilize the representation learning capabilities of machine learning methods.

Cohan et al. [17] conducted an in-depth study on binary and multi-label classification of various illness classes, along with a control group, using models such as logistic regression, Support Vector Machine (SVM), Convolutional Neural Network (CNN), and supervised Fasttext. Their best f1-score of 0.54 for binary classification and 0.27 for multi-label classification serves as a benchmark for similar research in this domain. On the other hand, Kim et al. [59] focused on binary classification tasks, and their CNN-based model produced an impressive average f1-score of 0.85, which significantly outperformed XGBoost's average score of 0.50. It is worth noting that, unlike Cohan's dataset, Kim's data was extracted from subreddits that were dedicated to each illness thus text included an explicit declaration by the user about their condition and their experience, while Cohan's dataset excluded such text which makes it more applicable for creating models that can be deployed in real work. Other positive attempts at using Reddit posts to detect depression and anxiety came to the same conclusion that Natural Language Processing (NLP) models which are based on N-gram models as well as deep learning systems can be a practical solution for the detection of mental health disorder and early prevention [100, 52].

2.2 Text Classification

Text classification is the fundamental technique in natural processing to categorize texts according to established criteria, context, or a shared subject. "Document retrieval, categorization, routing, filtering, and clustering, as well as natural language processing tasks such as tagging, word sense disambiguation, and some aspects of understanding, can be formulated as text classification" [63].

There are different text classification systems such as rule-based systems, machine learning systems, and hybrid systems.

2.2.1 Rule-Based System

The rule-based system falls under classical or knowledge-based Artificial Intelligence (AI). They operate on the tenet of classifying text based on handcrafted linguistic rules which are predefined and fixed. These rules guide the system based on semantically important textual elements of a text to identify the category of the group.

These systems are highly interpretable and tuneable based on needs over time but required deep knowledge of the domain that is being analyzed. Chua et al. [14] studied eight different strategies to include negated features within the process of Inductive Rule Learning (IRL). Their findings indicated that the option to include negated features within the IRL process produces more effective classifiers. Rule-based, context-dependent word clustering methods have shown improved classification accuracy as well efficient dimensionality reduction [39, 6, 28].

2.2.2 Machine Learning Based System

Machine Learning (ML) models have a probabilistic approach to their work. Unlike the rule-based system where each rule for classification is provided to the system, ML learns to classify new data based on past observations, which is called the training phase. Data and their associated labels are given to the model during the training phase. Based on predetermined parameters and using a trial-and-error approach, the model infers the relationship between the data and the label.

Although ML models are quite effective at identifying relationships in big data, they are difficult to interpret and validate. This topic is further discussed in detail in Section 2.3.

2.2.3 Hybrid System

Hybrid systems combine machine learning base classifiers with rule-based systems to boost the model's performance. By combining machine learning techniques with rule-based systems, hybrid models can achieve better performance than either approach on its own. These models are meticulously tuned on a case-by-case basis to incorporate specific rules for conflicts that weren't modeled correctly by the base classifier. These models are more appropriate for use case cases with unique requirements that cannot be generalized by using conventional methods.

Asogwa et al. [3] employed the use of Naive Bayes model and Support Vector Machine (SVM) model to produce their hybrid system which boost their model performance to 96.76% accuracy as against the 61.45% and 69.21% of the Naïve Bayes and SVM models respectively. Different variations of logical and probabilistic models have been combined to design a hybrid system [56, 88, 92] have observed significant improvement in performance compared to the performance of each individual model.

2.3 Machine Learning Models

Similar research has been done in the field to detect mental illness using text and by studying online social media behavior using machine learning and deep learning methods. Jina et al. [53] implemented a deep learning model using XGBoost and CNN-based classification on Reddit posts to accurately identify mental disorders like depression, anxiety, bipolar, borderline personality disorder, schizophrenia, and autism.

Ivan et al. [48] have analyzed how a Hierarchical Attention Network (HAN) can be better at analyzing social media posts under certain conditions than traditional models. The study found that the performance of ML models is highly dependent on the availability of data, with results worsening as the amount of available data decreases.

2.3.1 Support Vector Machines

Linear classifiers are the simplest form of classifiers that operate by determining the linear border between the classes but are unable to resolve a non-linear relationship using conventional techniques. Support Vector Machine (SVM) model is a supervised learning algorithm that is an extension of linear classier which has the ability to model non-linear class boundaries. SVM creates a hyperplane in N dimensions, which is used to classify data points. The hyperplane is selected on the basis that the distance between the nearest point of each class and the hyperplane is maximum — called margin. The best hyperplane is defined by Equ. 2.1.

$$\langle \overrightarrow{w}, \overrightarrow{x} \rangle + b = 0 \tag{2.1}$$

where $\overrightarrow{w}$ represents normal vector to the hyperplane and $\overrightarrow{x}$ is input vector of n dimensions.

If the classes are completely linearly separable then a hard margin is used but SVM can also perform non-linear classification using a soft margin, which allows some points to be misclassified or fall under the margin. For nonlinear classification, the hinge loss function is the more optimal function [96]. Hinge loss is defined by Equ. 2.2.

$$\max(0, 1 - y_i(\overrightarrow{w} x_i - b)) \tag{2.2}$$

where y_i is is the i-th target and $\overrightarrow{w} x_i - b$ is the i-th output.

SVM can solve non-linear problems by using Radial Basis Function (RBF) as the kernel. A kernel is a function that takes the non-linear problem and converts it into a non-linear form. This kernel however has issues with scaling extremely large datasets. RBF kernel is defined by Equ. 2.3.

$$K\left(x, x'\right) = e^{-\gamma \|x - x'\|^2} \tag{2.3}$$

where $\|x - x'\|^2$ is the squared Euclidean distance between two data points and γ is a scalar that defines the influence of a data point.

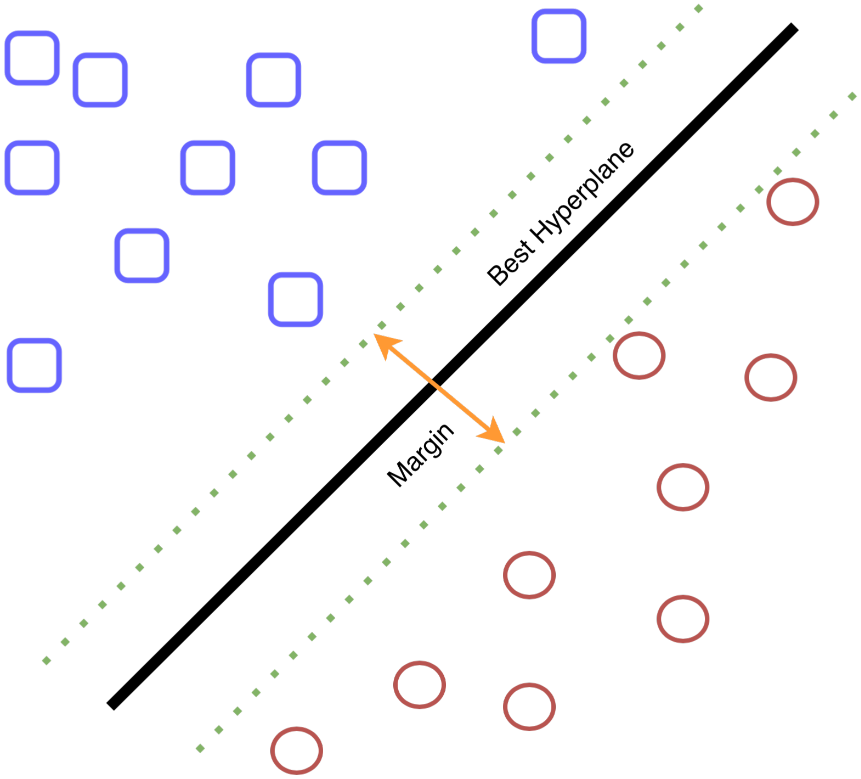


Figure 2.1: Support Vector Machine Illustration

Since there is virtually no limit to the dimensions of the hyperplane, SVM is highly effective in extracting features that may not be visible or too complex to compute for other models. It is also robust again sparse data which is frequently a problem with massive text-based data.

The base algorithm for SVM was designed by Vapnik et al. [10] in 1992. SVM [23, 112, 113] was initially applied in the field of pattern recognition and regression but later found useful application in text classification. Joachims et al. [54] conclude that SVM is well suited for the task of text classification due to its capabilities of dealing with sparse data and high dimensional features. However, its inability to consider semantic relations makes it unsuitable for information retrieval. Several advanced versions of SVM have been proposed [67, 105, 68] where SVM is able to generate multiple parallel hyperplanes by utilizing a pair of generalized Eigen-value problems for the purpose of classification of data.

2.3.2 Convolution Neural Network

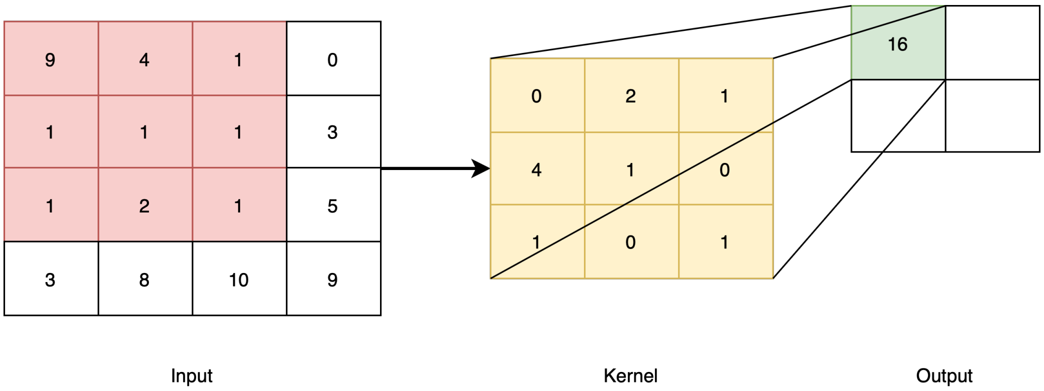
Neural network models are inspired by the human brain and the working of neurons and their structure. Recently, Deep Learning (DL) has attracted a lot of attention because of its capabilities and high performance for both supervised and unsupervised tasks. The simplest neural network is the feed-forward neural network or a fully connected network. Each working node or neuron in a different layer performs affine transformation of the input by multiplying it with the “weight” and adding “bias” to it, which are the learning parameters of the model. It is the base structure for all types of neural networks. It has single input and output layer with one or more hidden layers. Equ. 2.4 shows the affine transformation in a neural network.

$$y = \sigma\left(W \cdot x + b\right) \tag{2.4}$$

where x and y are the input and output vectors, W is the weight and b is the bias vector. σ represents the activation function.

A network is a set of these weights and biases and networks model the relationship between input and output by optimizing these parameters. The optimization is done through gradient descent to iteratively minimize the loss function.

Convolutional Neural Network (CNN) is a type of neural network that has shown great potential with working multi-dimensional and spatial data such as images and thus is extensively used in the computer vision domain. The main building block of the CNN model is called the convolutional layer or kernel. A kernel takes a matrix as an input and outputs another matrix proportional to the shape of the input using element-wise multiplication and addition. The output matrix is calculated by the kernel moving across (towards the right side) the image according to a fixed stride across the image starting from the top left of the image. The convolutional operation of a kernel on the matrix is demonstrated in Fig. 2.2. The kernel has the shape of 3×3 operating over an image of 4×4 with a stride of 1. This operation results in an output that is 2×2 in shape.



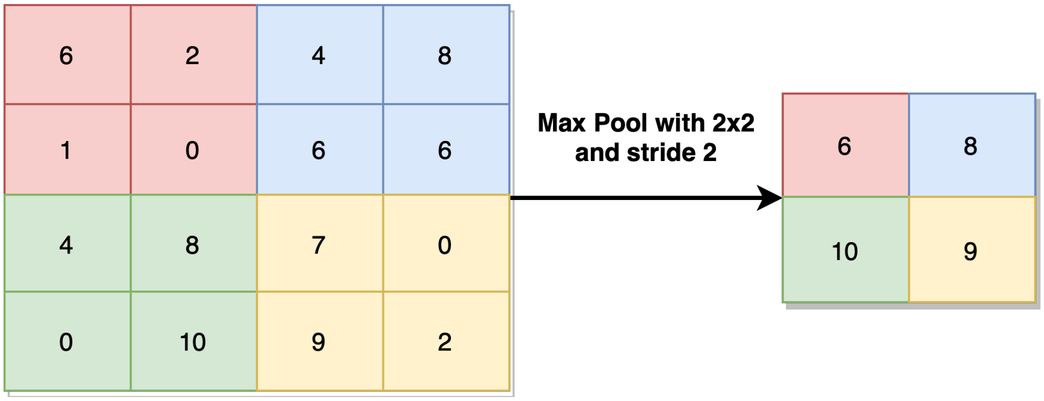
9	4	1	0
1	1	1	3
1	2	1	5
3	8	10	9

0	2	1
4	1	0
1	0	1

16	

Figure 2.2: Illustration of Convolutional operation

Another important operation of the CNN model is pooling. This operation is used to reduce the dimensionality of the data and allows the network to work with reduced parameters and pass on only the important information from the data. The most common type of pooling operating is used in max pooling where the max value from the pool or the section is selected and passed on to the next layer. Fig. 2.3 demonstrates the max pooling operation of 4 × 4 input with max pool layer of 2 × 2 window and stride of 2 across the data.



6	2	4	8
1	0	6	6
4	8	7	0
0	10	9	2

6	8
10	9

Figure 2.3: Illustration of Max Pooling operation

Fig. 2.4 illustrates the general architecture of all CNN models where a single image flows through different layers of the network. The image is passed through a number of convolution and pooling layers where each layer obtains different information about the image. The first layer learns about simple shapes like vertical or horizontal lines and more complicated information like colours and complicated structures like circles or the presence of different color shades is assessed by further layers. This information is then passed to fully connected layers where classification is performed using output activation functions such as softmax [7].

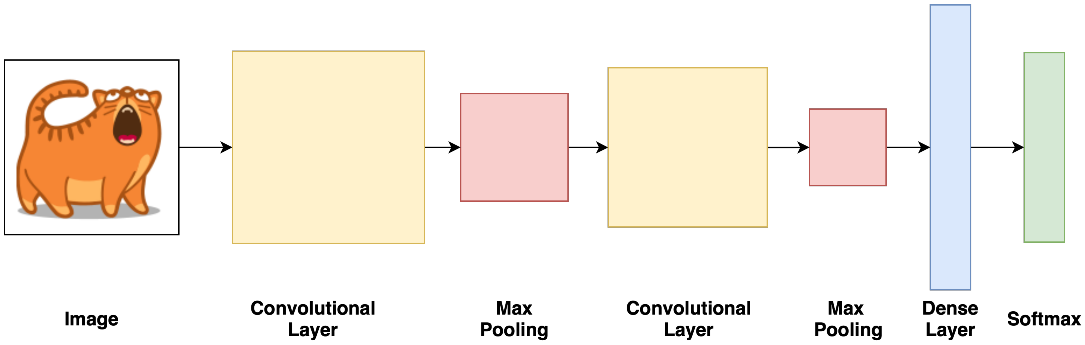


Figure 2.4: CNN architecture Illustration

Long Short Term Memory

Recurrent Neural Networks (RNN) are another type of neural network that are able to handle a temporal relationship in sequential data which makes them extremely useful for working time series data and modeling sentences. It is a feed-forward neural network where for each time step its hidden layer is connected to the hidden layer for the previous timestamp. Such recurrent connections allows the network to create memory through time. However, RNNs have major difficulties maintaining long-range dependencies because of the vanishing gradient problem which causes the gradient to become exponentially small while being propagated back across time steps.

Long Short Term Network (LSTM) was designed by Hochreiter and Schmidhuber [45] in order to solve the problem of long-range dependency. A weight multiplication and non-linearity combination known as a "gate" is how an LSTM cell abstracts its mathematical processes. Input, output, and forget gates are the three types of gates found in an LSTM node. The input gate process determines how much of the new input is required to modify the cell state. The output gate determines how the current cell state is added to the current hidden state. The forget gate determines how much of the previous cell state to forget. The input, output, and forget gate are shown by Equ. 2.5, Equ. 2.6 and Equ. 2.7 respectively.

$$i_i = \sigma\left(W_i \cdot [h_{t-1}, x_t] + b_i\right) \tag{2.5}$$

$$o_i = \sigma\left(W_o \cdot [h_{t-1}, x_t] + b_o\right) \tag{2.6}$$

$$f_i = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{2.7}$$

where f_t, i_t and o_t represent the outputs of the forget, input, and output gates respectively and σ represents the sigmoidu activation function.

Fig. 2.5 illustrates LSTM cell and the inner working of the hidden state [12].

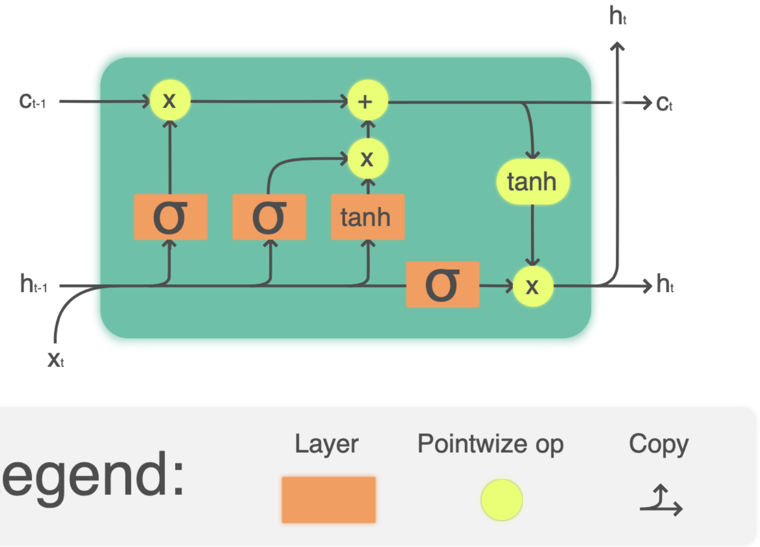


Figure 2.5: Illustration of LSTM Cell (Chevalier, 2018, Fig. 1)

The current cell state in language modeling is updated by multiplying the forget gate output with the previous cell state and adding the previous hidden state determined by the input gate. Information about specific topics is stored in segments of the cell state and updated by the forget and input gates as new input is received. This process is given by Equ. 2.8.

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tanh\left(W_C \cdot [h_{t-1}, x_t] + b_C\right) \tag{2.8}$$

The hidden state is updated by passing the current cell state through a hyperbolic tangent function and limiting it with the output gate. Output is generated by passing h_t through an appropriate output activation function such as sigmoid [40] or softmax.

Text classification has been the focus of study in the field of computer science since the very beginning and with the emergence of deep learning methods, this field has seen significant advancements and breakthroughs [66, 90, 2, 74]. LSTM models have better performance compared to linear models and have also shown good performance when working with unstructured free text [42, 9, 104]. Khan [58] employed CNN-LSTM in the field of medicine to predict mortality in ICU and found CNN-LSTM outperformed the baseline model and unlike the baseline model didn't underpredict mortality. They also suggested by changing the architecture of the CNN-LSTM accuracy of the model could be further improved. Computer vision is another field where CNN-LSTM has excelled by achieving an f1-score of 98.9% for detecting COVID-19 from X-ray and scoring 90.8% accuracy for detecting tempted video files [47, 97].

2.3.3 Word Embeddings

In order to model language, text-based data must be represented mathematically in a way that reflects the word's meaning and its relationships to other words in the corpus. There are several ways to represent data in numerical form such as bag-of-words [91] or discrete encoding. However, these methods are arbitrary and do not represent any relationship that exists among different words in the corpus. The model won't have any contextual knowledge regarding the relationship between man and woman and animals and humans if the words are merely encoded numerically, such as 1, 2, and 3 for man, woman, and dog, respectively. As a consequence, the model is unable to comprehend natural language and process them. There is a need to convert this information and capture this knowledge in mathematical expression and this can be done with the help of word embeddings.

Word embeddings allow the mapping of words in form of mathematical vectors in a manner such that similar-meaning words are closer to each other in vector space [58]. The embeddings created for the words boy, girl, prince, and princess, for instance, should be executed to perform the arithmetic described in Equ. 2.9 and illustrated in Fig. 2.6.

$$x_{\text{prince}} - x_{\text{boy}} + x_{\text{girl}} \approx x_{\text{princess}} \tag{2.9}$$

where x_w denotes the vector for the word w.

Embeddings for ML are divided into two categories: non-context-sensitive and context-sensitive techniques. Contextual embeddings provide each word a representation based on its context, capturing word usage across a range of situations and encoding cross-linguistic knowledge.

Popular embeddings such as word2vec [73] and GloVe [87] are non-context sensitive while Bidirectional Encoder Representations from Transformers (BERT) [30] and Context2vec [70] are context sensitive. Pennington et at. [87] introduced GloVe in 2014 which is an unsupervised learning algorithm for obtaining vector representations for words. The model works by training elements in a word-word cooccurrence matrix in a large corpus. The models outperformed similar models on word analogy tasks and have since their introduction been used in many studies.

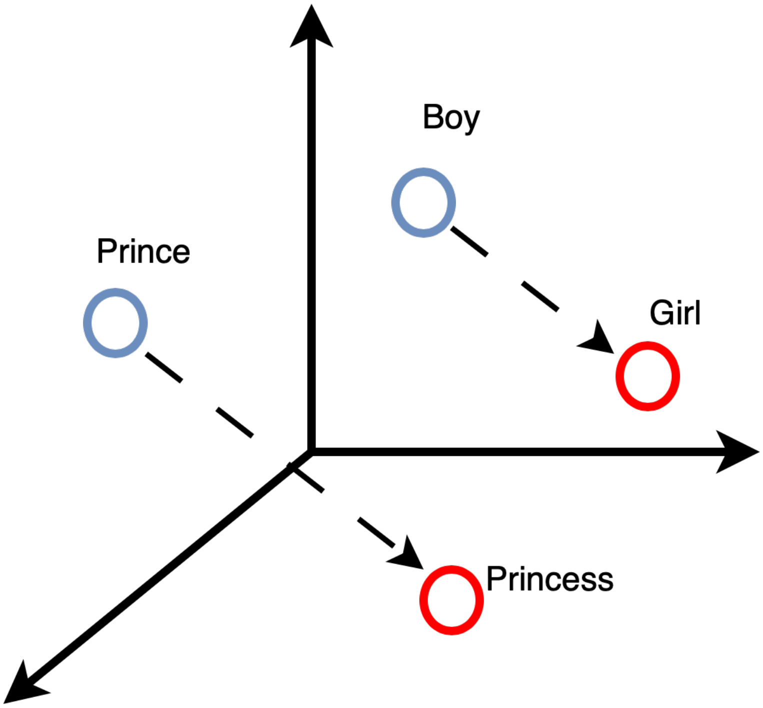


Figure 2.6: Word Embedding Illustration

Miaschi and Dell'Orletta [72] studied linguistic knowledge encoded in the internal representation of BERT and Word2vec. The study concluded both models encode sentence-level properties similarly, but BERT excels at syntax and raw text while Word2vec is better at morphosyntax. BERT can encode sentence-level phenomena within single-word embeddings, and the most informative word representation is often the last token of each input sequence.. Naili et al. [76] investigated the most effective method to learn word vector representation in the English language using Word2vec, GloVe and Latent Semantic Analysis and concluded word2vec had better representations in small dimensional semantic space.

2.4 Common N-gram Method

Common N-gram (CNG) method used in this study is based on character n-gram distribution. Many ML models and word embeddings are solely languages [11] dependant and are difficult to generalize and transfer knowledge to other languages. Asian languages such as Chinese and Japanese do not have explicit word boundaries. By using character-level n-grams the authors this problem can be solved. The frequency and consistency with which a person uses punctuation, characters, character-casing, and whitespace can play a significant role in their psychological makeup [17, 15, 64] which can be lost in traditional ML-based systems since these models treat this information as redundant. Using the character n-gram instead of the word n-gram helps us include and analyze these minute morphological features that play an important role.

2.4.1 N-gram

N-gram is a contiguous sequence of n items from a given text. N-grams can be letter or word-based. N defines the length of the sequence. Unigram will contain one item, bigram will have 2 items, and so on. Character n-grams are language-independent and hence character-level n-gram language models can be easily applied to any language and even non-language sequences such as DNA and music. It also has further applications in data compression and cryptanalysis.

2.4.2 N-gram Derivation

Characters and words can both be retrieved from n-grams. Character-based elements are referred to as character n-grams and word-based ones are referred to as word n-grams. Character n-gram considers all symbols along with alphabets such as whitespace, punctuations, or newlines. The scope of this study is limited to character n-gram only. Fig. 2.7 illustrates the extraction of character n-gram via sliding window where the value of N is four.

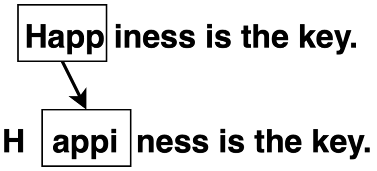


Figure 2.7: Character N-gram Extraction Illustration

When the n-gram extractions are for words it is known as word n-gram. Unlike character n-gram, word n-gram extraction ignores white space. Fig. 2.8 illustrates the extraction of the word n-gram where the value of N is two.

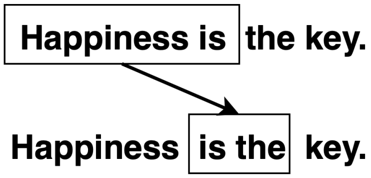


Figure 2.8: Word N-gram Extraction Illustration

For the authorship attribution task Equ. 2.10 was used in the original study by Bennet [8] where only bigrams were used which can be attributed to 1970's limited technological capabilities.

$$\sum_{n,m} \left[f_1\left(n,m\right) - f_2\left(n,m\right)\right]^2 \tag{2.10}$$

where n, m are indices over the range 1 to 26 and f_1 and f_2 are normalized character bigram frequencies for different authors.

N-grams have application in several tasks such as spelling correction as most of these typographical errors are minor and can be easily anticipated with character n-grams. These errors can be fixed with simple operations such as insertion, deletion, and substitution [103]. N-grams have also been proven good at capturing the structure of human language [93] and can be used for language detection. N-grams are also able to perform document clustering due to their ability to identify the technical words that are domain-specific [71].

Bennet [8] was the first author who successfully implement the character bigram-based method for authorship attribution and obtained promising results. Building on this work, Keselj et al. [57] implemented common N-gram for authorship attribution task in three different languages. This study implements the CNG method with a larger n-gram and profile length, unlike the previous studies which were limited to computational resources available and obtained 100% accuracy for the English language, and the Greek dataset obtained higher scores than previously reported. The N-gram method has been successful in other languages, including Korean. The authors tackle the challenge of using weighted document indexing in the Korean language by using a combination of word-based and n-gram-based indexing, using bigrams and trigrams of Korean syllables [62]. They conclude their system is likely more effective than word-based indexing and could approach the performance of morpheme-based indexing. Successful implementation of CNG method lead to its application for other classification tasks such as disease detection, identification of source code author, and music composer attribution tasks. [107, 35, 119].

Calvin et al. [107] studied the impact of Alzheimer's-type dementia on patients' spontaneous speech using the CNG approach. They found that using byte-level n-grams to construct class profiles, combined with minimal training instances, produced successful models in detecting and rating the severity of the disease.

2.4.3 Relative N-gram Signature

Jankowska et al. [50] proposed a visualization system called Relative N-gram Signature analysis methods the based on common N-gram method for text classification. The system was designed to conduct a detailed investigation of the characteristics of n-gram in the documents and provide greater insights into the working of the classifier in an intuitive manner. The relative N-gram signature method was inspired by the works of Collins et al. [19] which presented subsets of a faceted corpus through the extraction of distinctive words, which are then displayed as visually appealing tag clouds in a parallel format. For purpose of illustration of the n-gram signature visualization technique Fig. 2.9 was generated from the dataset used in this study.

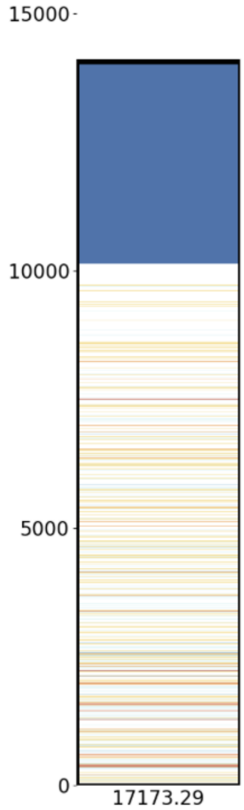


Figure 2.9: Illustration of N-gram Signature

This visualization method helps in discovering patterns in n-grams for a given profile to understand similar themes in different classes and interpret models working. The original study also foresees modification of the n-gram profile of a class on the task-dependant decision of a user for classification which can boost CNG's performance. The system is based on representing the difference in usage of frequency of n-gram between two classes to understand misclassification or similarity between them. Cohan et al. [17] employed LIWC lexicon [85] in order to categorize language through the examination of psycholinguistic attributes. This analysis sheds light on the nuanced elements present in writing that can indicate potential mental health issues. The findings of this study can assist researchers in verifying their model interpretations prior to testing in real-world scenarios. Eichstaedta et al. [31] performed LIWC analysis on Facebook posts and discovered people with depression have higher tendencies to use first-person pronouns in their speech. Ramos et al. [94] conducted validation of mHealth app for depression screening using 4 psychological depression instruments and came to the conclusion that even though mental health screening relied on subjective measurements. By implementing a novel approach using mobile technology, the validation process is augmented with credibility and increases confidence in the automated system.

Previous studies on character n-gram visualization used Latent Semantic Indexing, where character n-grams served as the terms of a document, and visualized the first LSI dimension to identify similar clusters [101]. Wolkowicz et al. [118] designed a music visualization tool to detect themes in musical pieces. This tool was built on the concept of self-similarity matrices, which encode a musical piece through character n-gram similarity of sequences.

Chapter 3

Methodology

In this chapter, we discuss various methods that were designed and implemented in this study.

3.1 Dataset Description

This study uses Self-Reported Mental Health Diagnoses (SMHD) dataset. SMHD data collection approach is based on Reddit Self-reported Depression Diagnosis (RSDD) dataset [122] and further builds on RSDD by including synonyms in matching patterns and adding data for eight additional diseases in addition to depression [17].

The SMHD dataset comprises posts made on Reddit by users (referred to as "diagnosed users") who acknowledge having been given a diagnosis for one or more of nine mental health problems, as well as posts made by matched control users. The data of diagnosed users had every post made to a mental health-related subreddit or having a keyword associated with a mental health issue removed; as a result of the selection process, the data of control users do not have such postings.[17].

The entire SMHD dataset consists of 116 million posts from 335,000 users [17] but due to limited computation and memory resources, this study uses a smaller sample dataset.

This study focuses on six mental illnesses included in DSM-5 standards [4]. Four conditions are top-level DSM-5 disorders: schizophrenia spectrum disorders (*schizophrenia*), bipolar disorders (*bipolar*), depressive disorders (*depression*), and anxiety disorders (*anxiety*). The 2 other conditions are one rank lower: autism spectrum disorders (*autism*) and *Attention-Deficit / Hyperactivity Disorder (ADHD)* under neurodevelopmental disorders.

For the sampled dataset used in this study, patients with and without a different diagnosis of mental illness were balanced at 5% for depression, 3.8% for anxiety, 2.8% for ADHD, 1% for autism, 0.6% for Bipolar Disorder, 0.45% for schizophrenia and rest 86.35 % for the control group to simulate true mental illness prevalence [81, 24, 99, 82]. Sampling was done with help of an in-built sampling function provided by *Pandas* [106]. Table 3.1 describes detailed statistics for each label including the total numbers of users and posts.

Label	**# of Users**	**# of posts**	**Total Word Count**	**Total # of char**	**Total # of Non Alpha char**
Control	8636	1,754,943	64,720,509	357,904,732	17,146,263
Depression	500	253,894	4,437,815	22,847,497	677,775
Anxiety	380	39,850	2,361,869	12,752,601	530,154
ADHD	280	31,695	1,827,233	10,056,686	403,166
Austim	100	12,184	767,305	4,209,406	181,642
Bipolar	60	6,138	390,626	2,108,075	82,527
Schizophrenia	45	5,361	314,569	1,749,242	79,053
Total	**10,000**	**2,104,065**	**74,819,926**	**411,628,239**	**19,100,580**

Table 3.1: Label wise Data Statistics

The dataset consists of 2 main columns which are *label* which indicates the illness profile or if the user belongs to the control group and *posts* written by respective users in various subreddits.

All usernames were replaced with random identifiers to prevent users' identities from being known without the use of external information. We strictly adhere to the Data Usage Agreement in obtaining the SMHD dataset. We ensure that no attempts are made to identify or establish connections between individual users within the dataset and any other users.

3.2 Data Preprocessing

Data preprocessing is a vital step that transforms raw data which can consist of false, corrupt, duplicate, or inconsistent data into a more quickly and effectively processed format. It greatly improves analysis and boosts the performance of ML algorithms. Fig. 3.1 shows preprocessing pipeline used in this study.

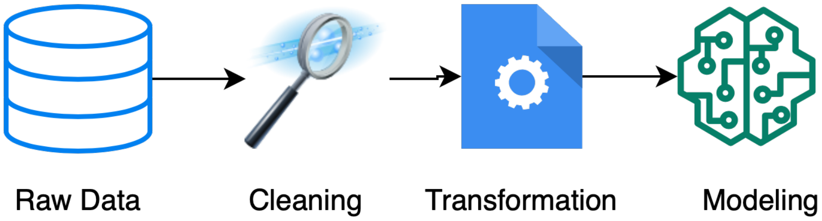


Figure 3.1: Data Preprocessing Pipeline

In the cleaning phase, the raw data is cleaned by removing data points with missing text or labels and by removing corrupt data. Since letter case does not necessarily provide important information, all letters were converted to lowercase to combat data sparsity. Furthermore, in order to ensure that the dataset contains sufficiently informative posts for analysis, any posts with fewer than 50 words were dropped. In the data transformation phase, the data is tokenized and padded. Tokenization is a preprocessing technique that splits a stream of text into words, phrases, symbols, or other meaningful elements called tokens [38, 114]. For example, consider the sentence :

“I am tired of this work.”

The tokenization of this sentence:

“I”, “am”, “tired”, “of”, “this”, “work”, “.”

Tokens obtained are vectorized in a text corpus, by turning each text into either a sequence of integers or into a vector where the coefficient for each token could be binary, based on word count or based on term frequency-inverse document frequency (TF-IDF). TF-IDF was proposed to numerically reflect how important a word is to a document in a corpus and reduces the effect of common words which do not provide useful information [55]. The weight of each term by TF-IDF is calculated using Equ. 3.1.

$$W\left(d,t\right)=TF\left(d,t\right)\times\log\left(\frac{N}{df\left(t\right)}\right) \tag{3.1}$$

where N is the number of documents present and *df(t)* is the number of documents containing the term t in the entire corpus.

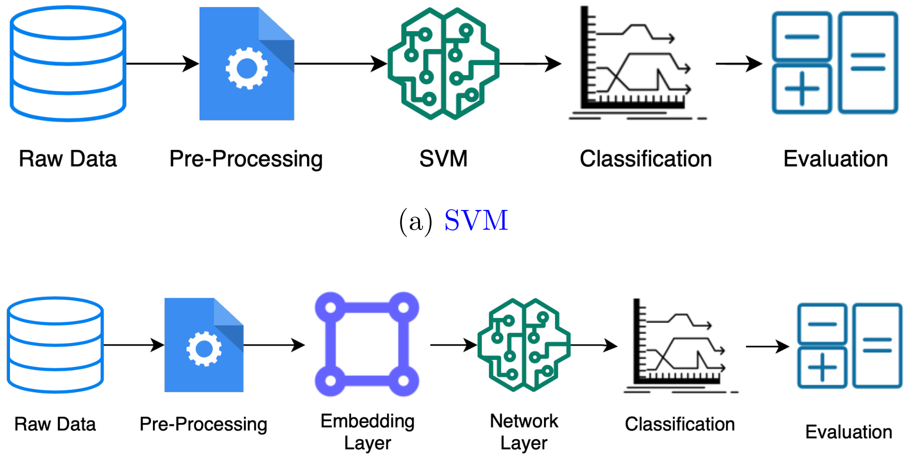
Padding transforms sequences to be equal to the desired length by adding 0 before or after the sequence. If the sequences are longer then they are truncated so that they fit the desired length [13]. Many studies that implement deep learning models for classification tasks remove stopwords [65, 98] from the dataset. Still, for this research, this step was not done as stopwords and punctuation usage and frequency can be indicators of mental health illness [86]. After the preprocessing was completed, transformed data was passed to classification models.

3.3 Classification Methods for Detection of Mental Disorders from Text

In this section, we will discuss classification methods from this research that are used to detect mental health illnesses from social media posts. It will briefly describe the data flow and classification algorithm.

3.3.1 Supervised Learning Model

The two supervised learning algorithms used in this study are SVM and CNN-LSTM models. Process pipelines for both models have similar phases as shown in Fig. 3.2a and Fig. 3.2b for SVM and CNN-LSTM respectively. For the SVM model, no embedding layer is present and a linear kernel is chosen as a nonlinear kernel such as Radial Basis Function does not scale well with the big dataset. CNN layers for feature extraction on input data are paired with LSTMs to facilitate sequence prediction in the CNN-LSTM architecture.



(a) SVM

(b) CNN-LSTM

Figure 3.2: Process Phases for Supervised Learning Models

For the CNN-LSTM model, the data is passed to an embedding layer before it is passed on to the first convolutional layer which is followed by the max pooling layer. The embedding layer helps us convert words into vectors of fixed length which can help the model better understand words and reduce dimensionality. Embedding layer values can later be extracted to study how the model interpreted each word in relation to the corpus given to it. After the max pooling layer data is passed to the LSTM layer which is connected to a fully connected layer and has softmax as the output activation function. The model also consists of a dropout layer [13] which deactivates random neurons during training to avoid overfitting. Fig. 3.3 shows an overview of CNN-LSTM model.

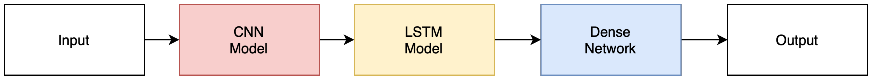


Figure 3.3: Overview of CNN-LSTM model

3.3.2 Common N-gram Method

This section will introduce and explain the profile similarity algorithm for the classification task used in this study. Unknown texts are classified using this profile similarity algorithm.

Profile Similarity Algorithm

This study uses the same algorithm proposed by Keselj et al. [57] and evaluates its performance on a different genre of data. In order to keep the profile of limited size to avoid memory explosion and rare and uncommon n-grams influencing the results we design an illness profile to be set of L the most frequent n-grams with their normalized frequencies. By limiting profile size L most common n-gram a profile can be defined as a set of L pairs $\{(x_1, f_1), (x_2, f_2), ...(x_L, f_L)\}$, n-grams and their normalized frequency. Profile length limiting also helps with computational performance as well as reduces the change of model overfitting [108]. Previous works [57] have shown values for $n \leq\ 8$ can be before computational and memory limitations and performance decreases are observed.

The original study done by Bennett [8] used Equ. 2.10 which gave simple Euclidean distance with equal weight to all n-gram frequency differences included in the profile since the profile size was smaller and data was not sparse. An extension of Bennett's work was done by Keselj et al. [57] where they have to use Equ. 3.2 for normalized similarity metric. The normalization of n-gram frequencies was done because if the absolute difference measure is used the more frequent n-grams would be highlighted more since the absolute differences in their frequencies are bigger. In order to normalize these differences they were divided by the average frequency for a given n-gram. As explained in the original study, "a difference of 0.1 between two profiles of an n-gram with frequencies of 0.9 and 0.8 will be less weighted than a similar difference between two profiles of an n-gram with frequencies of 0.2 and 0.1".

$$\sum_{n \in profile} \left(\frac{f_1(n) - f_2(n)}{\frac{f_1(n)+f_2(n)}{2}} \right)^2 = \sum_{n \in profile} \left(\frac{2 \cdot (f_1(n) - f_2(n))}{f_1(n) + f_2(n)} \right)^2 \tag{3.2}$$

where $f_1(n)$ and $f_2(n)$ are frequencies of an n-gram n in the unknown text and the illness profile.

Algo. 1 outlines the algorithm used in CNG method for calculating the dissimilarity metric between unknown text and known illness profile. The algorithm will always return a positive value, the measure of dissimilarity. For texts that are identical the algorithm will return 0, i.e if the L most frequent n-grams in a profile are all present in the unknown text profile and are truly similar. We can assign text based on the least dissimilarity value to either an illness class or a control group. In essence, the CNG classifier utilizes the k-nearest neighbor algorithm with k set to 1, using a dissimilarity measure between profiles to classify data instances

Algorithm 1: Profile Dissimilarity ($profile_1$, $profile_2$)

```
$sum \leftarrow 0$
**for all** n-grams $x$ contained in $profile_1$ and $profile_2$ **do**
  let $f_1$ and $f_2$ be frequencies of $x$ in $profile_1$ and $profile_2$ (zero if they are not included)
  add square of the normalized difference of $f_1$ and $f_1$ to sum:
  $sum \leftarrow sum + (2 \cdot (f_1 - f_2)/(f_1 + f_2))^2$;
**end for**
**return** $sum$
```

where $f_1(n)$ and $f_2(n)$ are frequencies of an n-gram n in the illness and the document profile.

3.4 Visualization Techniques

This study implements an n-gram visualization technique called Relative N-gram Signature as discussed in Section 2.4.3. This section will discuss its implementation and briefly discuss examples of this method.

Single Relative Signature

A relative n-gram signature is built based on two n-gram profiles that reflect the usage frequency of n-grams between them. Suppose P_1 and P_2 are two n-gram size L profiles to be analyzed. The signature is created in the following way: First, all the n-grams in P_1 (which is the base document that serves as the background for the signature) are ordered based on their frequency followed by the n-grams that appear only in P_2 in a similar manner preserving their order in P_2. The n-gram at i-th index will be the i-th most common n-gram in the base document when $i \leq L$. If $i > L$, it means the i-th n-gram is i-th most common n-gram in the second document since the n-gram with a number greater than L doesn't appear in the base document. The lower the value of i the more common the n-gram is. The dissimilarity index is calculated using Equ. 3.2 for n-grams used to create the signature.

A single relative n-gram signature is represented in Fig. 3.4 is used when only two classes are to analyzed. The dual contrast color mapping represents the distance between the n-gram frequency of the profile to the base documents. The white color represents that distance is zero or close to zero and the n-gram has an equal frequency in both documents. The red scale presents that n-gram is more frequent in the base document i.e P_1 document. While the blue scale presents the n-gram as less frequent in the base document and more in the other document. The darker the strip, the higher distance between the profiles with respect to the particular n-gram.

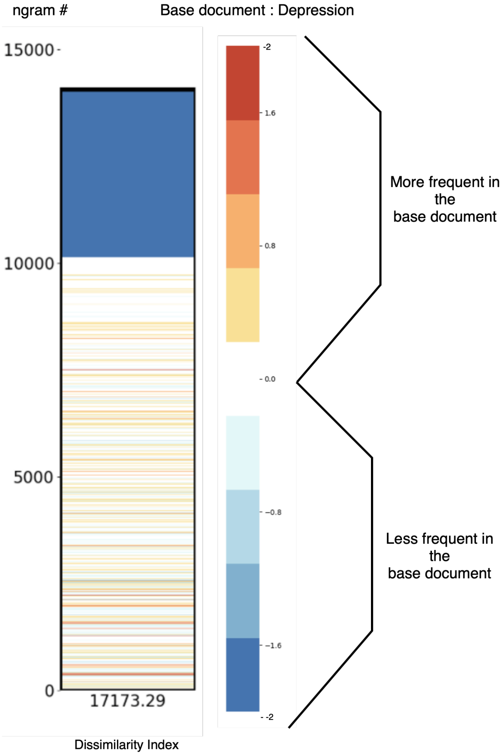


Figure 3.4: A single relative n-gram signature of Schizophrenia profile with Bipolar as a base document with parameters n = 4 and L = 10000

The top part of the signature presents n-grams higher than L. These n-grams do not appear in the base profile and have a dark blue color which is the maximum distance. This part provides information about how many n-grams appear in P_2 only. Similarly, bottom part of the signature contains more red strips and blue because it represents n-grams no higher than L, meaning they appear only in the base document (P_1). The taller the signature the less similar the profile are to one another since more n-grams appear only in one profile i.e P_2. Similarly, darker signatures either blue or red also indicate high dissimilarity between the two profiles.

Fig. 3.5 shows a “zoomed-in” version of Fig. 3.4 extracted using only sample data for the purpose of illustration, shows that n-grams “_YOU”, “T_TH” and “WORK” are more often used in text labeled for *"bipolar"* class than in *"schizophrenia"*. While n-grams "_MY" and "_OF_" are more common in *"schizophrenia"*. N-grams like "ING_" and "THE_" have an equal frequency in both profiles. In the visualization "_" presents whitespace.

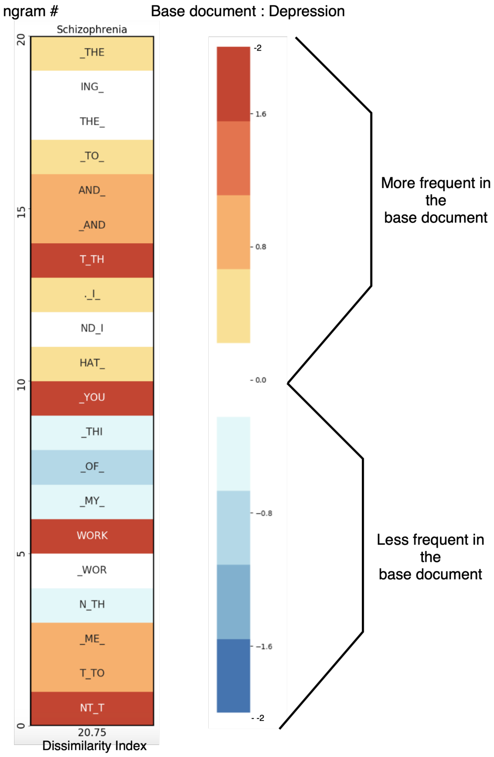


Figure 3.5: First 20 n-gram of Relative N-gram signature of Schizophrenia with depression as a base document with parameters n = 4 and L = 10000

A relative signature of a document with a base document of itself would be completely white since the distance between all n-grams will be zero indicating the identical profile and the dissimilarity index will be zero.

Series of Relative Signature

The use of a series of relative n-gram signatures can be highly beneficial in gaining an understanding of n-gram signatures when comparing multiple profiles with one another on one single base document. This can be extremely useful when analyzing multi-classification tasks and understanding the model's performance. It allows analyzing one profile with base documents and also comparing multiple profiles with one another to understand the dissimilarity between them.

Fig. 3.6 illustrates the relative signature of 6 illness profiles with the control group as the base document for n = 4 and N = 10000. From the figure, it is readily apparent that depression is the least similar to the control group while schizophrenia is most similar to the base document which could make classification between these two groups difficult. Since control is the base document for all the profiles it is also useful to see n-grams that are close to index 10,000 i.e 10,000 most common n-grams are closer to the control profile and the respective illness profile (P_2).

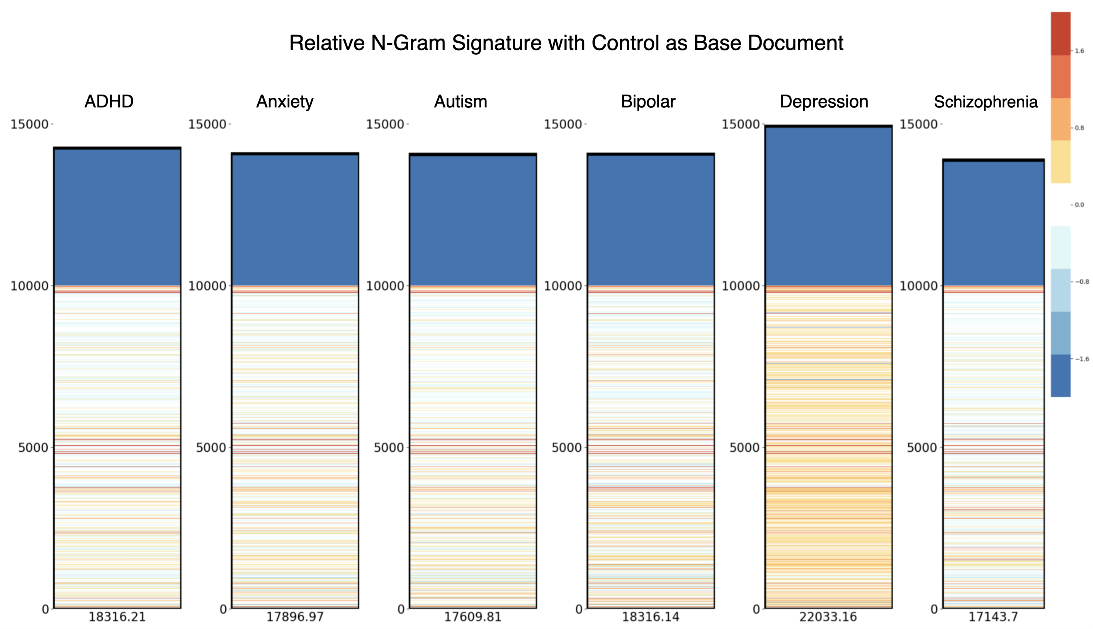


Figure 3.6: Series of relative signatures with control as a base Document and parameters n = 4 and L = 10000

As shown in Fig. 3.7 is a “zoomed in” version of Fig. 3.6. It helps us see if there are common or emerging themes between two or more profiles. N-grams “_TO_”, “AND_”, and “THAT” are more frequent in the control group. N-grams “_THIN” and “HINK” helps us deduces that the word “think” is more frequently used in depression profile and sentences with the word “think” could be further analyzed to understand the written statement of people suffering from depression.

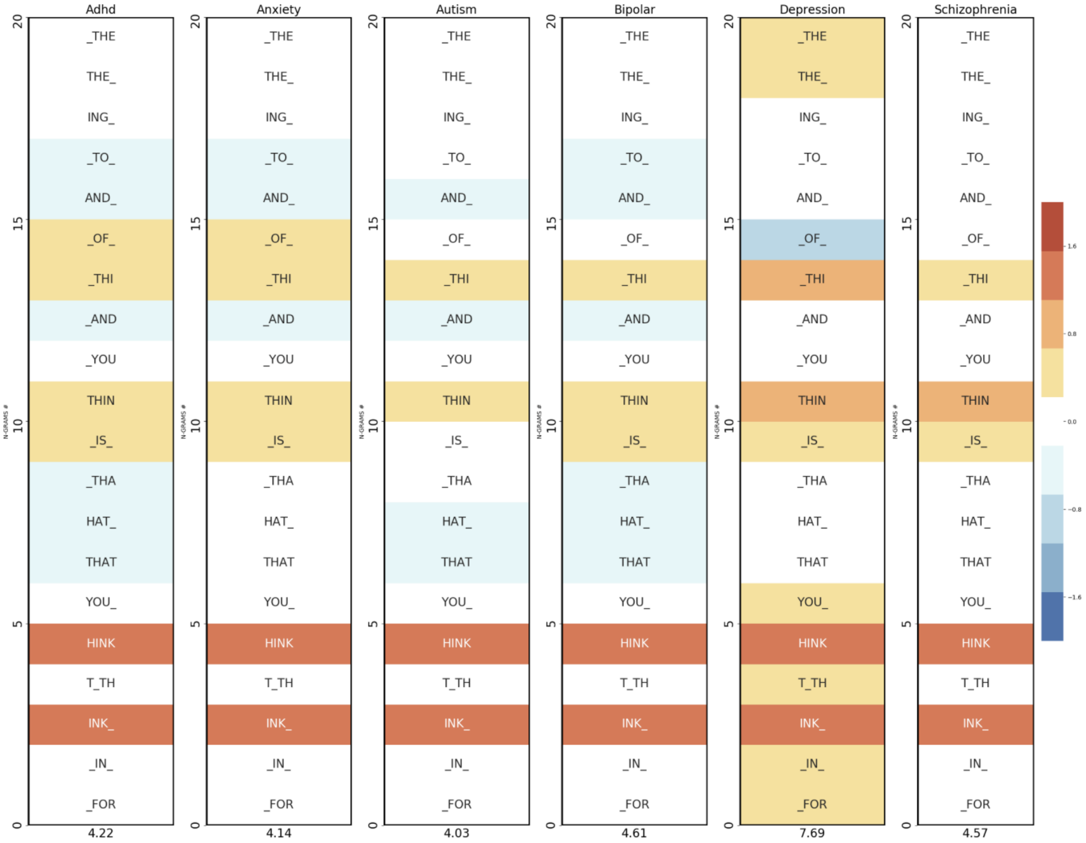


Figure 3.7: First 20 n-gram in Series of Relative N-gram signature with control as a base document with parameters n = 4 and L = 10000

3.5 Evaluation Matrices

Various techniques can be employed to evaluate the potential of different classification models. Accuracy is the one of most intuitive and frequently used evaluation criterion. Accuracy is defined as the ratio of correct classification to total predictions [89]. Due to the unbalanced nature of the dataset used and its sampling, using accuracy as an evaluation criterion is an unreliable indicator of model performance. If the model is biased towards the majority class and cannot classify the minority class the accuracy of the model will still be high creating a deceptive impression of model performance. F1-score, precision, and recall are more robust against the imbalanced dataset and provide more useful insights into the findings.

To measure the performance of the models and maintain the metrics with other prior research for easy comparison, this study uses the F1-score, precision, recall, and confusion matrix.

3.5.1 Precision

Precision provides the understanding of the proportion of positive identifications that was correct. It is the ratio of the true positive (TP) to the total positive classified by the model [89]. This is calculated using Equ. 3.3.

$$Precision = \frac{TP}{TP + FP} \tag{3.3}$$

3.5.2 Recall

Recall provides the understanding of the proportion of actual positives identified correctly. It is the ratio of true positives to actual true positives, which is the sum of true positives and false negatives (FN), in the dataset [89]. Recall is employed when it is more important to avoid a false negative than a false positive, which is typically the case in the healthcare industry. This is calculated using Equ. 3.4.

$$Recall = \frac{TP}{TP + FN} \tag{3.4}$$

3.5.3 F1-score

F1-score is the harmonic mean of precision and recall [89]. The value range of the f1-score is between 0 and 1. A high f1-score indicates high precision and recall values and vice versa while a mediocre d1-score indicates that one of the values is low and the other is high. This is calculated using Equ. 3.5.

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{3.5}$$

For this study, the focus is on the macro F1-score which is computed by calculating metrics for each label and finding their unweighted mean which does not take label imbalance into account, unlike the weighted F1-score. Macro F1-score is preferred instead of weighted f1-score which would introduce the same problem discussed as accuracy in Section 3.5 due to an imbalanced dataset [84].

$$Macro\, F1 = \frac{1}{n} \sum_{i=0}^{n} 2 \times \frac{Precision_i \times Recall_i}{Precision_i + Recall_i} \tag{3.6}$$

where n is the total number of labels present.

3.5.4 Confusion Matrix

The confusion matrix provides us with an understanding of the classification capability of the model for each label. A confusion matrix is a table of $n \times n$, where n is the number of labels in the dataset. The diagonal elements of the table represent the number of points for which the predicted label is equal to the true label, whilst the incorrect labels assigned by the classifier are represented by the off-diagonal elements. The higher the confusion matrix's diagonal values, the better, indicating correct predictions [84].

A confusion matrix offers a straightforward understanding of the model's strengths and limitations for each label which can not only be used to understand the main characteristics of different models but also labels that are extremely similar or dissimilar.

Chapter 4

Experiments and Results

4.1 Experimental Setup

Python programming language was used to implement and execute the experiments for this research. Python*v3.7.6* was employed for the implementation of the entire project.

Pandas and *Numpy* framework were used to import, handle and manipulate the dataset [106, 41]. *Scikit-learn* package provided implementation of Support Vector Machine classifier [84].

Text processing such as tokenization and padding of sequencing is done with the help of *Keras* API [13]. The neural network model is implemented with *Keras* API, which is a high-level API of *TensorFlow* [1] that implements neural network models and offers fundamental abstractions and components for creating and delivering machine learning solutions at rapid iteration rates.

Matplotlib is a cross-platform library for data visualization and graphical plotting. It is used to generate a confusion matrix for classification results [46]. *Seaborn* another high-level API based on *Matplotlib* [116], is used to generate relative N-gram signature for algorithm analysis in Section 4.3d. More details of implementation are provided in Appendix A.

4.2 Results and Discussion

In this section, we will discuss the performance and interpret models discussed in Section 2 briefly.

4.2.1 Model Comparison and Discussion

To find out the best-performing model we compare the evaluation metrics. We utilized precision, recall, and f1-score to assess how well the model predicted and distinguished each illness class and to determine whether it was biased towards any class. We employed the f1-score to compare the models' overall performance, as detailed in Section 3.5. We evaluated these models on test data, which was not used while training the models. Table 4.2 and Fig. 4.2 shows the performance of the three models tested in this experiment and Supervised FastText which was the best-performing model in the study [17] which created the dataset used in the thesis.

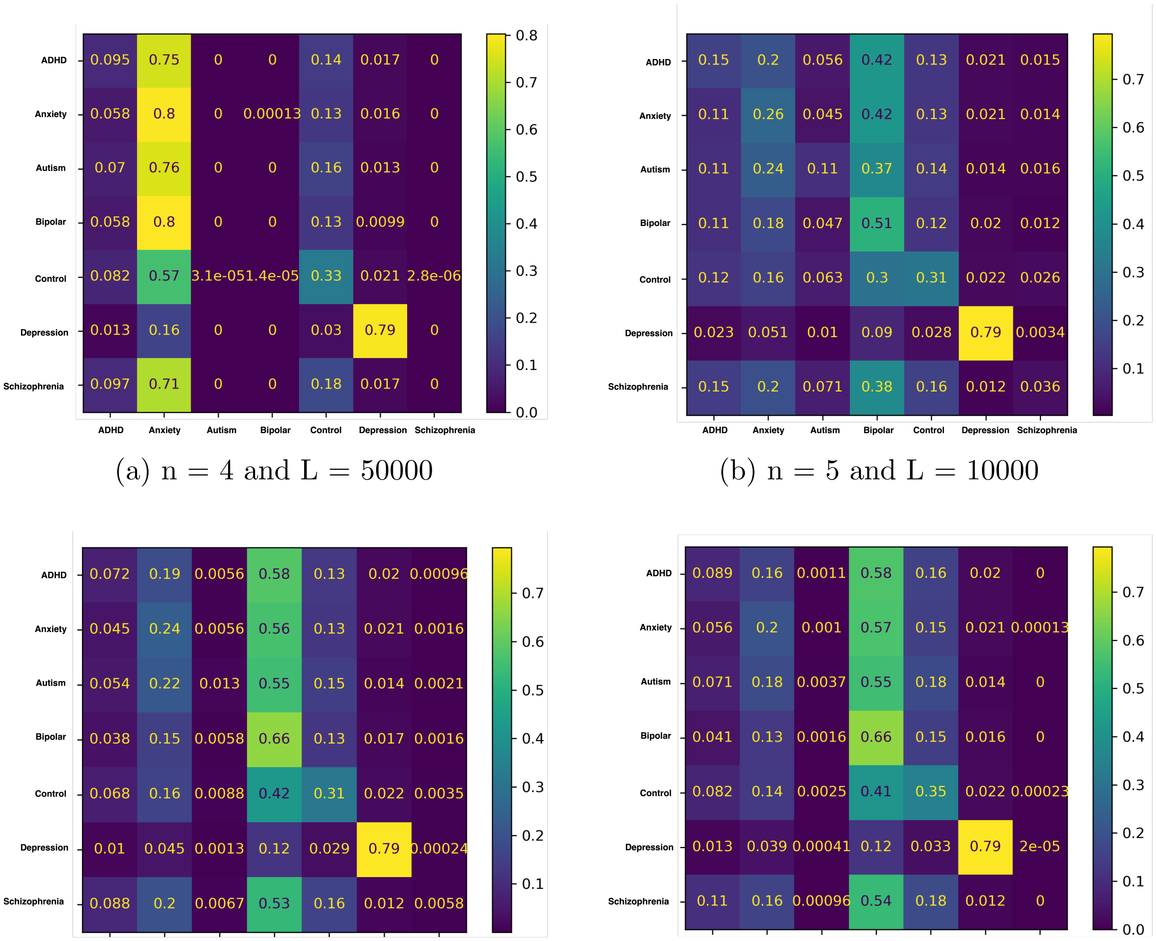
A grid search for finding optimal values of n and L was done for the CNG model, $3 \leq n \leq 5$ and $L = [10000, 20000, 50000, 100000]$. The reason for these limited values was, first as observed from past studies and experiments done with sample dataset, the highest performance was seen when $3 \leq n \leq 6$. Second, for $n \geq 7$ computation is slow and the memory requirement is high. Table 4.1 shows the f1-score obtained from the grid search.

	N-gram size		
Profile Size	**3**	**4**	**5**
10,000	0.18	0.19	**0.20**
20,000	0.13	0.19	**0.20**
50,000	0	**0.20**	**0.20**
100,000	0	0.16	0.23

Table 4.1: Pilot study F1-score for the multi-classification task using CNG

Even though our best f1-score (0.23) is for parameter n = 5 and L = 100,000 the model has a zero prediction rate for autism, bipolar control, and schizophrenia and overpredicts anxiety and control. The scope of this study is to find a model that can best distinguish and detect different mental illnesses and the CNG model with these parameters is not suitable for this purpose. However, it does provide enough evidence to test the CNG model with higher threshold values to assess the impact of less frequent n-grams on classification results in future studies.

F1-score of 0.20 was obtained for four different pairs of n and L but from the confusion matrix Fig. 4.1 we can see that for parameters n = 5 and L = 10,000, the model predicts 6 out of 7 labels with decent accuracy. While in other instances model has a similar f1-score for depression and control but a very low score for other classes. Due to this reason, CNG model's optimal parameters were selected as n = 5 and L= 10,000.



(a) n = 4 and L = 50000

(b) n = 5 and L = 10000

(c) n = 5 and L = 20000

(d) n = 5 and L = 50000

Figure 4.1: CNG model's Confusion Matrices Comparision

Model	Label						
	Depression	Anxiety	ADHD	Autism	Bipolar	Schizophrenia	Multi-class
Support Vector Machine	**P = 0.91** **R = 0.89** **F = 0.89**	P = 0.60 R = 0.60 F = 0.60	P = 0.59 R = 0.59 F = 0.59	P = 0.60 R = 0.60 F = 0.60	P = 0.62 R = 0.62 F = 0.61	P = 0.61 R = 0.60 F = 0.60	P = 0.20 R = 0.24 F = 0.21
Common N-Gram Method	P = 0.61 R = 0.55 F = 0.47	P = 0.62 R = 0.55 F = 0.48	P = 0.61 R = 0.55 F = 0.47	P = 0.59 R = 0.54 F = 0.46	P = 0.65 R = 0.54 F = 0.42	P = 0.58 R = 0.53 F = 0.45	**P = 0.27** **R = 0.31** **F = 0.20**
CNN Long Short-Term Memory Network	P = 0.82 R = 0.85 F = 0.83	**P = 0.65** **R = 0.65** **F = 0.65**	**P = 0.64** **R = 0.64** **F = 0.64**	**P = 0.65** **R = 0.65** **F = 0.65**	**P = 0.65** **R = 0.64** **F = 0.64**	**P = 0.66** **R = 0.66** **F = 0.66**	P = 0.29 R = 0.27 F = 0.16
Supervised FastText	P = 0.66 R = 0.44 F = 0.53	P = 0.67 R = 0.44 F = 0.53	P = 0.62 R = 0.37 F = 0.46	P = 0.68 R = 0.39 F = 0.49	P = 0.62 R = 0.42 F = 0.50	P = 0.69 R = 0.33 F = 0.45	P = 0.23 R = 0.44 F = 0.27

Table 4.2: Models Classification Performance

Binary classification task is performed between each illness and control group. For the binary classification task except for the depression class, CNN-LSTM outperforms

SVM and CNG models. For the depression class, SVM has the highest f1-score of 0.89 and CNN-LSTM has 0.83. CNN-LSTM provides consistent results across all labels for binary labels. While CNG's performance in the binary classification task may not be ideal, it still exhibits a noteworthy precision score. Although the f1-score falls below 0.5 for all classes, this does not detract from the fact that the model is able to correctly identify a significant number of positive samples.

For multiclass classification tasks, however, CNG method (0.20) outperforms CNN-LSTM (0.16). CNN-LSTM model is unable to predict any other class other than anxiety and depression. Even though SVM, which is acting as our baseline model, has a similar f1-score as CNG method it suffers from the same problem as CNN-LSTM does and is only able to predict depression and control class. This is most likely due to SVM overfitting on two majority classes and being unable to find boundaries for the minority classes which is evident from higher precision and recall rate but lower f1-score.

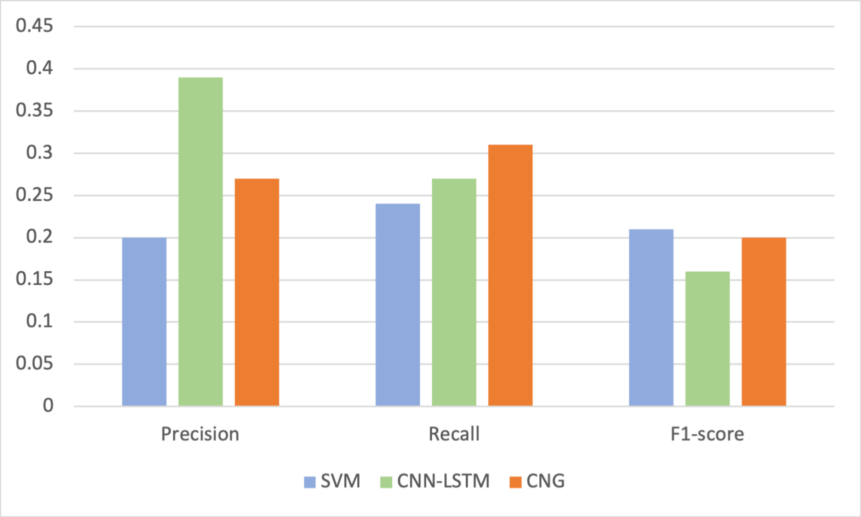


Figure 4.2: Model Comparision

In a previous study, [17], ML models that outperformed our method in terms of f1-score also showed inconsistency between precision and recall measures. Models would often report high precision and less recall rate or vice versa which is evidence that their models had an inconsistent performance where models would either return many results which are incorrect or very few results but they are correct.

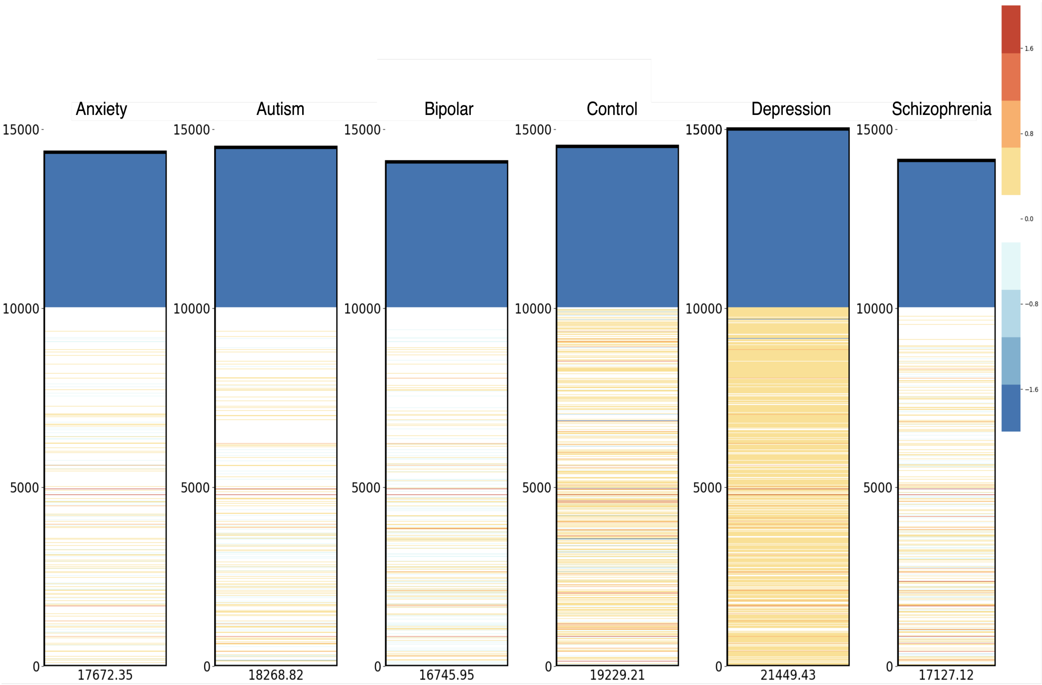
4.2.2 Model Analysis

This section presents the model analysis used in this research and interprets the model's learning and justifies its classification decisions.

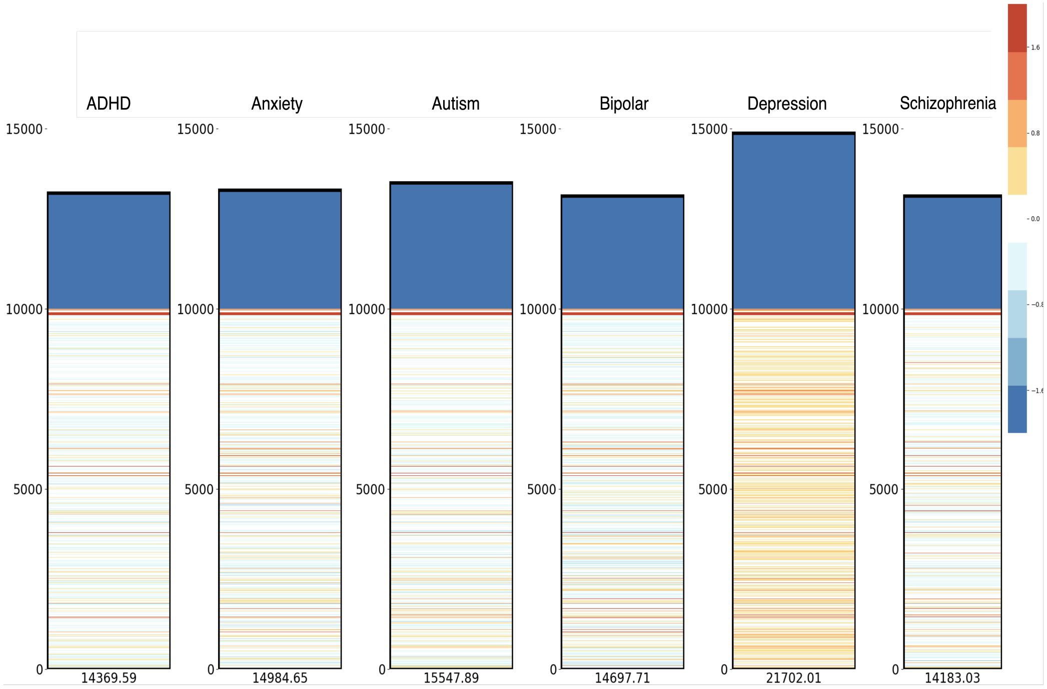
Common N-gram Method

Relative n-gram signatures detect document properties, they can provide a visual explanation of a CNG classifier's reasoning. Fig. 4.3 shows the relative n-gram signature for each of the labels as a background profile for parameters n = 5 and L = 10,000 for which CNG achieved the best results. From the confusion matrix 4.1c it can be observed that "'depression" was the label with the highest classification rate which can be attributed to the fact the depression's profile is least similar to the rest of the labels. Depression's candlestick is the longest and darkest of all the signatures plotted which can be observed from Fig. 4.3b. "Schizophrenia" has the lowest classification rate and is misclassified as control, bipolar, and anxiety labels. Schizophrenia has the lowest dissimilarity index of all when these labels act as base documents. This explains why the model is unable to distinguish between schizophrenia and other labels.

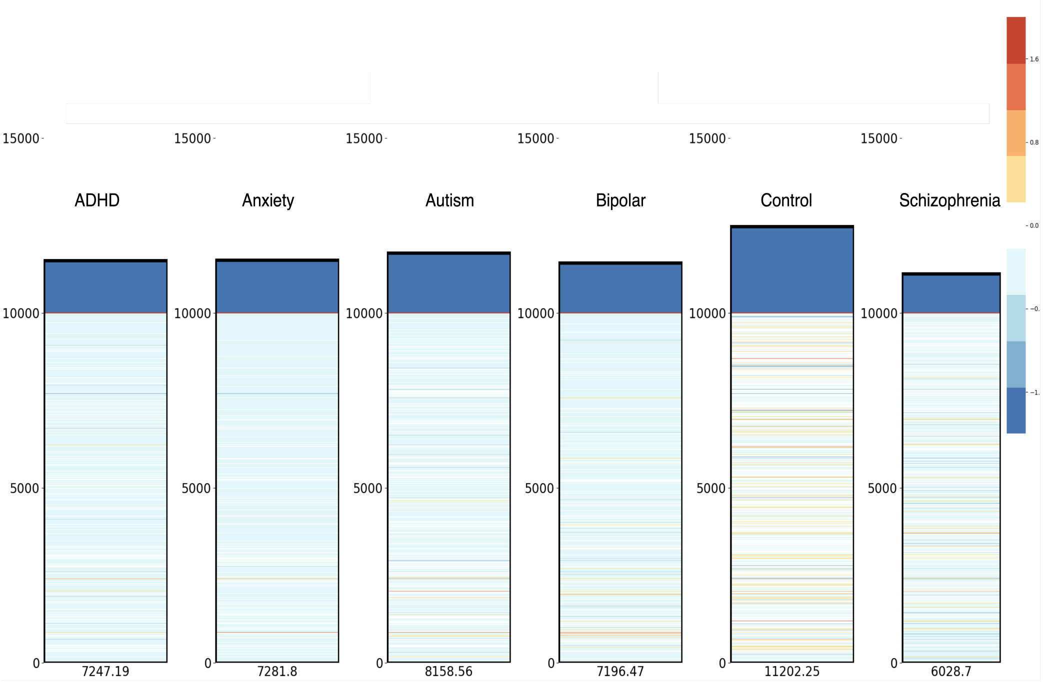
While the signature does help us understand model reasoning for classification to an extent it fails in the case of "ADHD" where even though for most of the documents it is not the least dissimilar profile, many of the profiles are misclassified as ADHD. This phenomenon was also noted by the original authors of system [50] and can be seen in Fig. 4.3a.



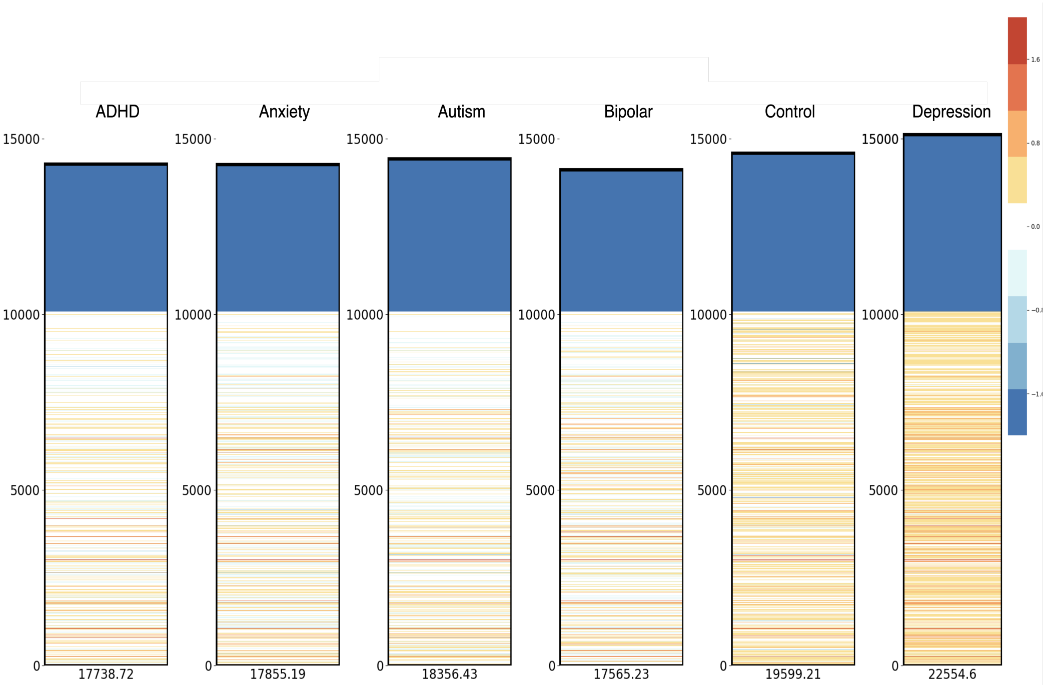
(a) Relative N-gram Signature with ADHD as Base Document



(b) Relative N-gram Signature with Control as Base Document



(c) Relative N-gram Signature with Depression as Base Document

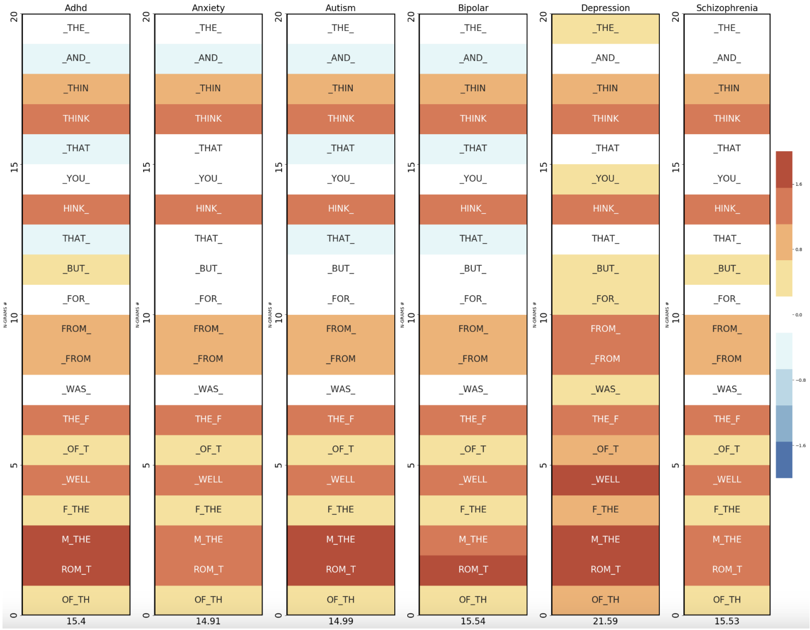


(d) Relative N-gram Signature with Schizophrenia as Base Document

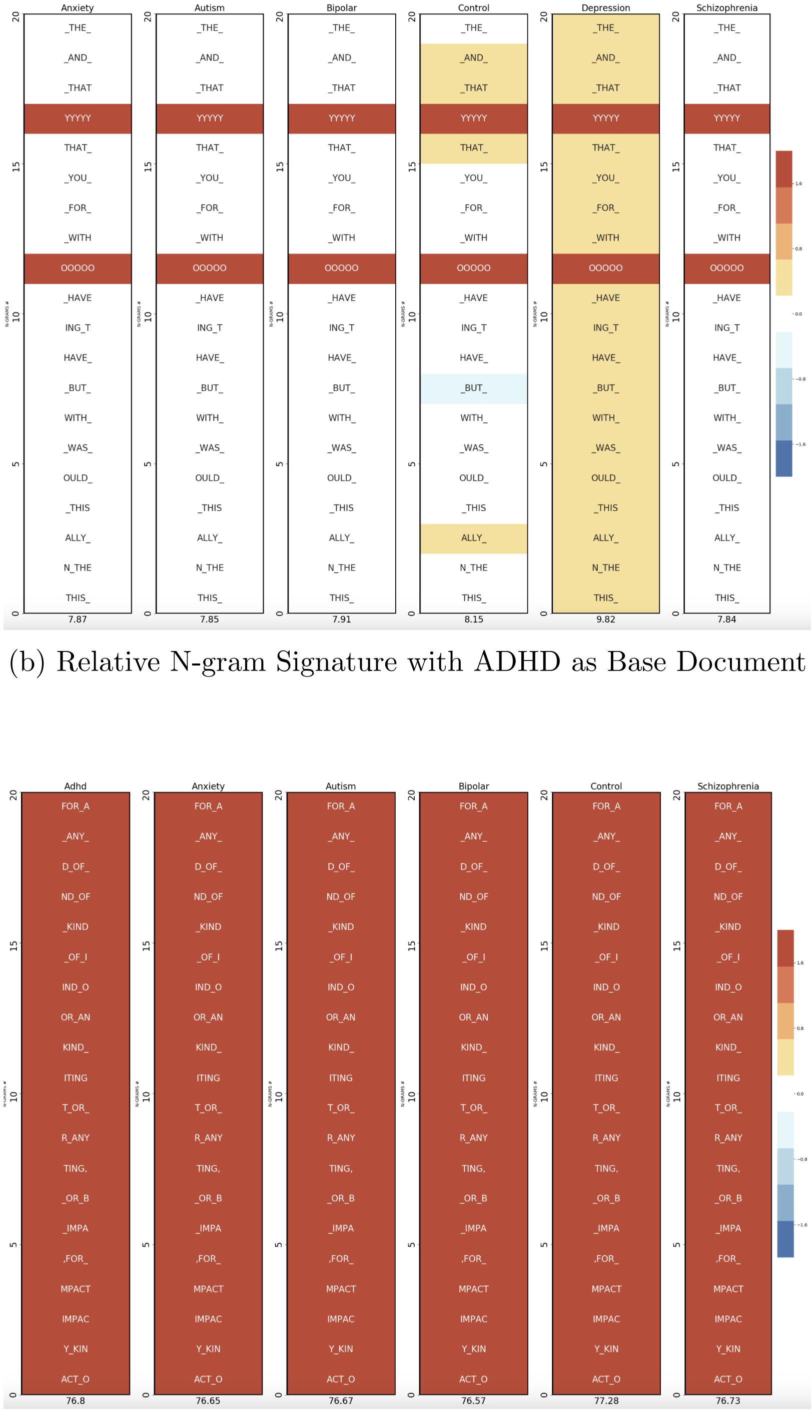
Figure 4.3: Relative N-gram Signatures for parameter n = 5 and L = 10,000

When observing the top 20 n-grams using the signature method it was observed that the top n-grams for anxiety, autism, bipolar, and schizophrenia are similar to one another and have almost the same frequencies. This is not an uncommon thing observed since stopwords were not removed which are usually in abundance in text written in the English language. A strange phenomenon when using depression as base class is that the top 20 n-grams are more common in depression profiles than in any other label.

From Fig. 4.4a it can be seen n-gram " YOU " which is a second-person pronoun is a more depressed profile than in the control group. It is often noticed that first and second-person pronouns are more commonly used by people with mental health illnesses. Phrases such as "act of kindness" and "impact of betrayal" generate n-grams such as "ACT O", "IMPAC" and "KIND " which are in the high-frequency depression profile suggesting that people suffering from depression were describing situations with extreme emotions. N-gram "ITING" was observed in sentences conveying emotions of excitement or worry, as words such as "waiting", "visiting", and "limiting" were present. These finds are consistent were previous work where similar emotions were observed from other data sources such as Facebook [31].



(a) Relative N-gram Signature with Control as Base Document



(b) Relative N-gram Signature with ADHD as Base Document

(c) Relative N-gram Signature with Depression as Base Document

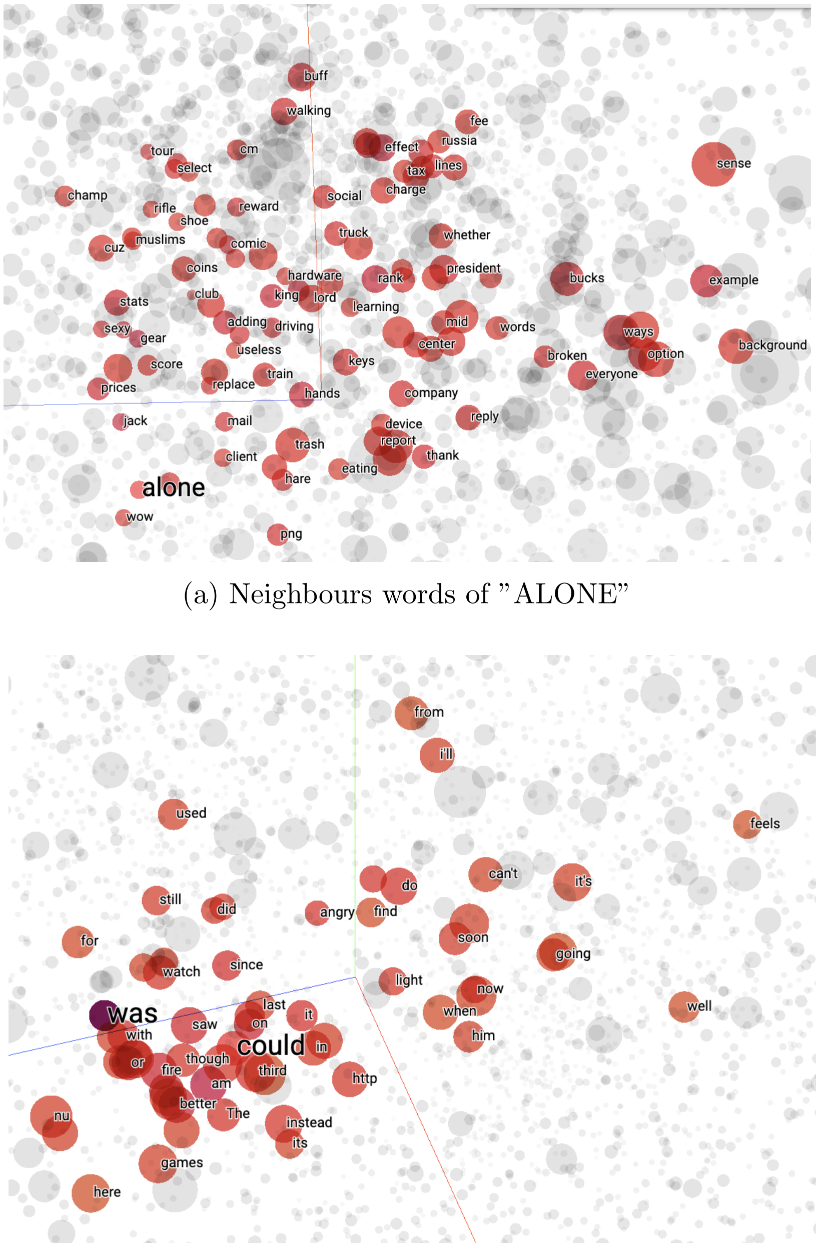
Figure 4.4: Relative N-gram Signatures of Top 20 N-grams

People suffering from mental disorders have also different usage of grammar and comparatively higher usage of past tense for sentence formation [115]. N-gram “OULD ” suggests the presence of verbs such as “could”, “would” and “should”. N-grams “YYYYY” and “OOOOO” are present in ADHD profiles with high frequency and mostly absent in other profiles. These particular n-grams were extracted from words like ”heyyyyy” and ”sooooo” and are thought to be linked to the repetitive behavior that is often observed in people with ADHD.

CNN-LSTM

To interpret the CNN-LSTM model this research analyzed the embedding layer of the model. The embeddings were trained from scratch instead of using a pre-trained embedding model. This was done because pre-trained embedding will have certain relations established based on norms present in English literature, which may not be true for social media and how people express their feelings and opinions on online platforms.

The embeddings are visualized using Tensoflow’s Embedding Projector [1]. The embeddings used in this experiment had a dimension of 128, which means the length of single vector space used for a word was 128. To visualize embedding in 3D space dimensionality reduction was done with help of t-SNE algorithm [44] learning rate was set at 1 and perplexity at 49. Ill-defined clusters appeared after 1400 iterations and then the shape stabilized.



(a) Neighbours words of ”ALONE”

(b) Neighbours words of ”COULD”

Figure 4.5: Word Embeddings from CNN-LSTM model

The deep learning model is also able to link words that are related to feelings of loneliness and sadness and other related words such as “useless”, “alone”, “social” and “trash”. The model is also able to build a relation between words such as “king” and “president” and words related to some fee structure such as “tax”, “charge”, “fee” and “line”. These words are part of the sentences there described bills or tax payments as the reason for distress and worry.

Word Embeddings extracted from CNN-LSTM model have some similarity with top n-grams that were observed from the CNG method as seen in Fig. 4.4b. From Fig. 4.5b words like “the”, “could”, “with”, “for”, and “this” are close neighbours in 3D space. CNN-LSTM model’s good recall rate for depression class can be explained by this phenomenon that the deep learning model was able to learn that these words are often present in text written by people with mental disorders.

These interesting findings suggest that deep learning models are highly capable of detecting mental illness with great accuracy and model language in a manner that aligns with the psychological markers commonly observed in mental disorders.

Chapter 5

Conclusion and Future Work

This chapter concludes this research by providing the conclusion, limitation, significance, and suggestions for future work to continue and improve this research.

5.1 Conclusion

To our knowledge, this is the first study to employ the character-level common N-gram method for the classification of DSM-5 mental disorders, while also incorporating a control group for evaluation To achieve this objective and assess CNG's performance comparison was done with state-of-the-art machine learning models.

The research attempted binary classification for each illness label with the control group and multi-classification using all models. Various metrics like precision, recall, and f1-score were used to evaluate performance and with help of the confusion matrix best model was identified when similar performance was achieved across different various models. The labels in the dataset were kept in ratio to simulate their true prevalence in the real world to accurately asses the model's potential real-world application.

This research demonstrated CNG's ability to work with complex, unbalanced, and large datasets, unlike deep learning models that may over or underfit with such data and are sensitive to their hyper-parameters. The CNG method had better able performance in terms of both f1-score, 0.20 compared to CNN-LSTM's score of 0.16, and the ability to classify and differentiate between illnesses in multi-classification tasks as explained in model analysis. Even though the CNG model is unable to beat the CNN-LSTM model in binary classification tasks, it is able to match the score of the Supervised FastText method used in the original study.

The Relative N-Gram Signature method was used to analyze the classification decision of the n-gram technique and provide interesting and useful insights. By analyzing the CNG model it was observed that it made classification decisions based on features that are consistent with findings in previous studies and that are used by professional psychologists to diagnose their patients. This provided enough evidence that NLP models can be used used to augment the process of automated systems that detect mental illness in people who can then seek professional help and strive for a better life. The CNG model has the potential to be integrated into individuals' personal devices, allowing for real-time analysis of their text and immediate notification of any indications of mental illness. By running the system directly on the local machine, users can benefit from rapid diagnosis without the need for an internet connection or remote servers, allowing for quick turnaround times and increased convenience. Secondly, by keeping sensitive medical data offline, users can ensure the confidentiality of their information and minimize the risk of data breaches or unauthorized access. This is especially important in the healthcare industry, where patient privacy is a critical concern [43].

5.2 Ethical Challenges

As the sharing of private information on public platforms and data analysis continues to increase, it is becoming easier to deduce personal traits such as health conditions or opinions such as political affiliations or sexual orientations of individuals from this data. While these techniques were developed with the intention of improving quality of life, there is a looming danger of their unethical utilization by private companies for discriminatory hiring practices or even by a totalitarian regime. To prevent misuse, strict government regulations surrounding data and data mining should be implemented, and adherence to HIPAA laws [5] should be enforced in their research and use. Additionally, increasing public awareness about the potential consequences of disclosing personal information can help individuals better understand the risks and benefits of sharing information publicly [109]. By doing so, we can take steps toward protecting individual privacy while still leveraging the benefits of data analysis. For this study we maintain all data privacy and usage laws as discussed in Section 3.1.

5.3 Limitations

The CNN-LSTM model was able to surpass the results of Logistic Regressoin, XGBoost, Linear SVM and CNN for the binary tasks from the original study that designed this dataset. Cohan et al. [17] used the same dataset in their experiment and were not able to achieve an f1-score higher than 0.55 in most cases for binary classification. However, due to limited computational resources as discussed in Section 3.1 which resulted in the use of the partial dataset, there is a threat to the external validity due to selection bias in this research that the selected subset of data used in this research happens to contain specific patterns that the model was able to effectively learn from, resulting in better classification performance. To validate these results, further experimentation on the entire dataset would be necessary. Furthermore, it should be noted that the original study utilized clinical subreddit groups to identify individuals with mental disorders and extract their data. However, this approach does not represent the most accurate or reliable method of diagnosis, and as such, it is not possible to independently verify the claims made by these individuals. It remains unclear whether these individuals were formally diagnosed with the disorder by a healthcare professional or if it was a self-diagnosis.

5.4 Future Work

This section offers a few possible research trajectories that can carry forward the work done in this thesis:

- CNG's performance can be further boosted by the creation of profiles tailored specifically to markers of an illness which can be facilitated using the n-gram visualization technique. Further investigation can be done by using large threshold values to study the effect of rare n-grams on the classification decision of the model.

- Considering our work showed binary models are better at classifying mental disorders, using Ensemble methods to combine multiple binary classification models to create a multi-class and multi-label classification system should be an interesting avenue of research since illnesses like depression or anxiety are symptoms or secondary illness accompanied when suffering from ADHD or schizophrenia. Another advantage of this kind of system will analysis for each model would be easier.

- Reddit-based data did not include any social-demographic-based information which can greatly affect a person's background and their response to mental disorders. A survey-based study can gather more information about this information along with writing samples from social media information to generate models that can better capture and understand language models of different regions and cultures of the world.

- Longitudinal data across a few months or years can be used to study the evolution of the language of people suffering from mental illness. LSTM models have shown great performance with such data and the relative n-gram signature method can be useful for gaining insights into a shift in words and language used over time.

Bibliography

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems. https://www.tensorflow.org, 2015. Accessed: 2023-2-12.

[2] Abdullah Alqahtani, Habib Ullah Khan, Shtwai Alsubai, Mohemmed Sha, Ahmad Almadhor, Tayyab Iqbal, and Sidra Abbas. An efficient approach for textual data classification using deep learning. *Front. Comput. Neurosci.*, 16:992296, 2022.

[3] DC Asogwa, SO Anigbogu, IE Onyenwe, and FA Sani. Text classification using hybrid machine learning algorithms on big data. *International Journal of Trend in Research and Development*, 6(5), 2021.

[4] American Psychiatric Association. *Diagnostic and Statistical Manual of Mental Disorders*. American Psychiatric Association, May 2013.

[5] Brian K. Atchinson and Daniel M. Fox. From the field: The politics of the health insurance portability and accountability act. *Health Affairs*, 16(3):146–150, May 1997.

[6] L. Douglas Baker and Andrew Kachites McCallum. Distributional clustering of words for text classification. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, page 96–103, New York, NY, USA, 1998. Association for Computing Machinery.

[7] Kunal Banerjee, Vishak Prasad C, Rishi Raj Gupta, Karthik Vyas, Anushree H, and Biswajit Mishra. Exploring alternatives to softmax function. In *Proceedings of the 2nd International Conference on Deep Learning Theory and Applications*, DeLTA 2021, pages 81–86. Science and Technology Publications, 2020.

[8] William Ralph Bennett. *Scientific and engineering problem-solving with the computer*. Prentice Hall series in automatic computation. Prentice-Hall, Englewood Cliffs, N.J., 1976.

[9] Willie Boag, Dustin Doss, Tristan Naumann, and Peter Szolovits. What's in a note? unpacking predictive value in clinical note representations. *AMIA Summits Transl. Sci. Proc.*, 2017:26–34, 2018.

[10] Bernhard E Boser, Isabelle M Guyon, and Vladimir N Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 144–152, 1992.

[11] Albin Byström. Extending a text classifier to multiple languages. https://www.diva-portal.org/smash/get/diva2:1608874/FULLTEXT01.pdf. Accessed: 2023-2-7.

[12] Guillaume Chevalier. Larnn: linear attention recurrent neural network. *arXiv preprint arXiv:1808.05578*, 2018.

[13] François Chollet. Keras. https://keras.io, 2015. Accessed: 2023-2-12.

[14] Stephanie Chua, Frans Coenen, and Grant Malcolm. Classification inductive rule learning with negated features. In *ADMA (1)*, volume 6440, pages 125–136, 11 2010.

[15] Lee Anna Clark, Bruce Cuthbert, Roberto Lewis-Fernández, William E Narrow, and Geoffrey M Reed. Three approaches to understanding and classifying mental disorder: ICD-11, DSM-5, and the national institute of mental health's research domain criteria (RDoC). *Psychol. Sci. Public Interest*, 18(2):72–145, 2017.

[16] Mayo Clinic. Depression (major depressive disorder). https://www.mayoclinic.org/diseases-conditions/depression/symptoms-causes/syc-20356007, October 2022. Accessed: 2023-2-12.

[17] Arman Cohan, Bart Desmet, Andrew Yates, Luca Soldaini, Sean MacAvaney, and Nazli Goharian. SMHD: a large-scale resource for exploring online language usage for multiple mental health conditions. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1485–1497, Santa Fe, New Mexico, USA, August 2018. Association for Computational Linguistics.

[18] Arman Cohan, Sydney Young, Andrew Yates, and Nazli Goharian. Triaging content severity in online mental health forums. *Journal of the Association for Information Science and Technology*, 68(11):2675–2689, 2017.

[19] Christopher Collins, Fernanda B. Viegas, and Martin Wattenberg. Parallel tag clouds to explore and analyze faceted text corpora. In *2009 IEEE Symposium on Visual Analytics Science and Technology*, pages 91–98, 2009.

[20] Glen Coppersmith, Mark Dredze, and Craig Harman. Quantifying mental health signals in Twitter. In *Proceedings of the Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality*,

pages 51–60, Baltimore, Maryland, USA, June 2014. Association for Computational Linguistics.

[21] Glen Coppersmith, Mark Dredze, Craig Harman, and Kristy Hollingshead. From ADHD to SAD: Analyzing the language of mental health on Twitter through self-reported diagnoses. In *Proceedings of the 2nd Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality*, pages 1–10, Denver, Colorado, June 5 2015. Association for Computational Linguistics.

[22] Glen Coppersmith, Mark Dredze, Craig Harman, Kristy Hollingshead, and Margaret Mitchell. CLPsych 2015 shared task: Depression and PTSD on Twitter. In *Proceedings of the 2nd Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality*, pages 31–39, Denver, Colorado, June 5 2015. Association for Computational Linguistics.

[23] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20:273–297, 1995.

[24] Saloni Dattani, Hannah Ritchie, and Max Roser. Mental health. https://ourworldindata.org/mental-health, 2021. Accessed: 2023-2-4.

[25] Munmun De Choudhury, Scott Counts, and Eric Horvitz. Major life changes and behavioral markers in social media: Case of childbirth. In *Proceedings of the 2013 Conference on Computer Supported Cooperative Work*, CSCW '13, page 1431–1442, New York, NY, USA, 2013. Association for Computing Machinery.

[26] Munmun De Choudhury, Michael Gamon, Scott Counts, and Eric Horvitz. Predicting depression via social media. *Proceedings of the International AAAI Conference on Web and Social Media*, 7(1):128–137, Aug. 2021.

[27] Munmun De Choudhury and Emre Kıcıman. The language of social support in social media and its effect on suicidal ideation risk. *Proc. Int. AAAI Conf. Weblogs Soc. Media*, 2017:32–41, 2017.

[28] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

[29] Bart Desmet and Véronique Hoste. Online suicide prevention through optimised text classification. *Information Sciences*, 439-440:61–78, 2018.

[30] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[31] Johannes C. Eichstaedt, Robert J. Smith, Raina M. Merchant, Lyle H. Ungar, Patrick Crutchley, Daniel Preoţiuc-Pietro, David A. Asch, and H. Andrew Schwartz. Facebook language predicts depression in medical records. *Proceedings of the National Academy of Sciences*, 115(44):11203–11208, 2018.

[32] Centers for Disease Control and Prevention. About mental health. https://www.cdc.gov/mentalhealth/learn/index.htm, November 2021. Accessed: 2023-2-10.

[33] Centers for Disease Control and Prevention. What is ADHD? https://www.cdc.gov/ncbddd/adhd/facts.html, August 2022. Accessed: 2023-2-12.

[34] Centers for Disease Control and Prevention. Signs and symptoms of autism spectrum disorder. https://www.cdc.gov/ncbddd/autism/signs.html, January 2023. Accessed: 2023-2-12.

[35] Georgia Frantzeskou, Efstathios Stamatatos, Stefanos Gritzalis, and Sokratis Katsikas. Effective identification of source code authors using byte-level information. In *Proceedings of the 28th International Conference on Software Engineering*, ICSE '06, page 893–896, New York, NY, USA, 2006. Association for Computing Machinery.

[36] Vijay N Garla and Cynthia Brandt. Ontology-guided feature engineering for clinical text classification. *Journal of biomedical informatics*, 45(5):992–998, 2012.

[37] Vijay N Garla and Cynthia Brandt. Knowledge-based biomedical word sense disambiguation: an evaluation and application to clinical document classification. *Journal of the American Medical Informatics Association*, 20(5):882–886, 2013.

[38] Gaurav Gupta and Sumit Malhotra. Text document tokenization for word frequency count using rapid miner (taking resume as an example). *Int. J. Comput. Appl*, 975:8887, 2015.

[39] Hui Han, Eren Manavoglu, C. Lee Giles, and Hongyuan Zha. Rule-based word clustering for text classification. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, SIGIR '03, page 445–446, New York, NY, USA, 2003. Association for Computing Machinery.

[40] Jun Han and Claudio Moraga. The influence of the sigmoid function parameters on the speed of backpropagation learning. In *Lecture Notes in Computer Science*, pages 195–201. Springer Berlin Heidelberg, 1995.

[41] Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian

Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020.

[42] Hrayr Harutyunyan, Hrant Khachatrian, David C Kale, Greg Ver Steeg, and Aram Galstyan. Multitask learning and benchmarking with clinical time series data. *Scientific data*, 6(1):96, 2019.

[43] John Havens, Jared Bielby, and Miguel Angel Pérez Alvarez. ETHICALLY ALIGNED DESIGN a vision for prioritizing human wellbeing with artificial intelligence and autonomous systems. *ETHICALLY ALIGNED DESIGN A Vision for Prioritizing Human Wellbeing with Artificial Intelligence and Autonomous Systems*, 2016.

[44] Geoffrey E Hinton and Sam Roweis. Stochastic neighbor embedding. *Advances in neural information processing systems*, 15, 2002.

[45] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[46] J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007.

[47] Md. Zabirul Islam, Md. Milon Islam, and Amanullah Asraf. A combined deep cnn-lstm network for the detection of novel coronavirus (covid-19) using x-ray images. *Informatics in Medicine Unlocked*, 20:100412, 2020.

[48] Sekulic Ivan and Strube Michael. Adapting deep learning methods for mental health prediction on social media. pages 322–327, Hong Kong, China, November 2019. Association for Computational Linguistics.

[49] Zunaira Jamil, Diana Inkpen, Prasadith Buddhitha, and Kenton White. Monitoring tweets for depression to detect at-risk users. In *Proceedings of the Fourth Workshop on Computational Linguistics and Clinical Psychology — From Linguistic Signal to Clinical Reality*, pages 32–40, Vancouver, BC, 2017. Association for Computational Linguistics.

[50] Magdalena Jankowska, Vlado Kešelj, and Evangelos Milios. Relative n-gram signatures: Document visualization at the level of character n-grams. In *2012 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pages 103–112, 2012.

[51] Jared Jashinsky, Scott H Burton, Carl L Hanson, Josh West, Christophe Giraud-Carrier, Michael D Barnes, and Trenton Argyle. Tracking suicide risk factors through twitter in the US. *Crisis*, 35(1):51–59, 2014.

[52] Shaoxiong Ji, Tianlin Zhang, Luna Ansari, Jie Fu, Prayag Tiwari, and Erik Cambria. MentalBERT: Publicly available pretrained language models for mental healthcare. *arXiv [cs.CL]*, 2021.

[53] Kim Jina, Lee Jieon, Park Eunil, and Han Jinyoung. A deep learning model for detecting mental illness from user content on social media. *Scientific Reports*, 10(11846(2020)), 2020.

[54] Thorsten Joachims. Text categorization with support vector machines: Learning with many relevant features. In *Machine Learning: ECML-98*, pages 137–142. Springer Berlin Heidelberg, Berlin, Heidelberg, 1998.

[55] Karen Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21, January 1972.

[56] SM Kamruzzaman and Farhana Haider. A hybrid learning algorithm for text classification. *arXiv preprint arXiv:1009.4574*, 2010.

[57] Vlado Kešelj, Fuchun Peng, Nick Cercone, and Calvin Thomas. N-gram-based author profiles for authorship attribution. *Proceedings of the Conference Pacific Association for Computational Linguistics PACLING'03: 2003*, 09 2003.

[58] Mohammad Hashir Khan. *A CNN-LSTM for predicting mortality in the ICU.* PhD thesis, University of Tennessee, 2019.

[59] Jina Kim, Jieon Lee, Eunil Park, and Jinyoung Han. A deep learning model for detecting mental illness from user content on social media. *Scientific Reports*, 10(1), July 2020.

[60] Rohan Kshirsagar, Robert Morris, and Sam Bowman. Detecting and explaining crisis. *arXiv preprint arXiv:1705.09585*, 2017.

[61] Daria J. Kuss and Mark D. Griffiths. Online social networking and addiction—a review of the psychological literature. *International Journal of Environmental Research and Public Health*, 8(9):3528–3552, August 2011.

[62] Joo Ho Lee and Jeong Soo Ahn. Using n-grams for korean text retrieval. In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '96, page 216–224, New York, NY, USA, 1996. Association for Computing Machinery.

[63] David D. Lewis and William A. Gale. A sequential algorithm for training text classifiers. In Bruce W. Croft and C. J. van Rijsbergen, editors, *SIGIR '94*, pages 3–12, London, 1994. Springer London.

[64] Kristen A. Lindquist, Jennifer K. MacCormack, and Holly Shablack. The role of language in emotion: predictions from psychological constructionism. *Frontiers in Psychology*, 6, 2015.

[65] Edward Loper and Steven Bird. NLTK: The natural language toolkit. In *Proceedings of the ACL Interactive Poster and Demonstration Sessions*, pages 214–217, Barcelona, Spain, July 2004. Association for Computational Linguistics.

[66] Hongxia Lu, Louis Ehwerhemuepha, and Cyril Rakovski. A comparative study on deep learning models for text classification of unstructured medical notes with various levels of class imbalance. *BMC Med. Res. Methodol.*, 22(1):181, 2022.

[67] Olvi L Mangasarian and David R Musicant. Lagrangian support vector machines. *Journal of Machine Learning Research*, 1(Mar):161–177, 2001.

[68] Olvi L Mangasarian and Edward W Wild. Multisurface proximal support vector machine classification via generalized eigenvalues. *IEEE transactions on pattern analysis and machine intelligence*, 28(1):69–74, 2005.

[69] Soraya Mehdizadeh. Self-presentation 2.0: Narcissism and self-esteem on facebook. *Cyberpsychology, behavior, and social networking*, 13(4):357–364, 2010.

[70] Oren Melamud, Jacob Goldberger, and Ido Dagan. context2vec: Learning generic context embedding with bidirectional LSTM. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, pages 51–61, Berlin, Germany, August 2016. Association for Computational Linguistics.

[71] Yingbo Miao, Vlado Kešelj, and Evangelos Milios. Document clustering using character n-grams: A comparative evaluation with term-based and word-based clustering. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, CIKM '05, page 357–358, New York, NY, USA, 2005. Association for Computing Machinery.

[72] Alessio Miaschi and Felice Dell'Orletta. Contextual and non-contextual word embeddings: an in-depth linguistic investigation. In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 110–119, 01 2020.

[73] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[74] Shervin Minaee, Nal Kalchbrenner, Erik Cambria, Narjes Nikzad, Meysam Chenaghlu, and Jianfeng Gao. Deep learning based text classification: A comprehensive review. arxiv e-prints, page. *arXiv preprint arXiv:2004.03705*, 2020.

[75] Margaret Mitchell, Kristy Hollingshead, and Glen Coppersmith. Quantifying the language of schizophrenia in social media. In *Proceedings of the 2nd Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality*, pages 11–20, Denver, Colorado, June 5 2015. Association for Computational Linguistics.

[76] Marwa Naili, Anja Habacha Chaibi, and Henda Hajjami Ben Ghezala. Comparative study of word embedding methods in topic segmentation. *Procedia Computer Science*, 112:340–349, 2017. Knowledge-Based and Intelligent Information Engineering Systems: Proceedings of the 21st International Conference, KES-20176-8 September 2017, Marseille, France.

[77] Faculty of Computer Science. Technical services. https://www.dal.ca/faculty/computerscience/for-faculty-staff/technical-services.html. Accessed: 2023-2-12.

[78] National Institute of Mental Health (NIMH). Bipolar disorder. https://www.nimh.nih.gov/health/topics/bipolar-disorder. Accessed: 2023-2-12.

[79] National Institute of Mental Health (NIMH). Schizophrenia. https://www.nimh.nih.gov/health/statistics/schizophrenia. Accessed: 2023-2-12.

[80] National Institute of Mental Health (NIMH). Anxiety. https://www.nimh.nih.gov/health/topics/anxiety-disorders, 1998. Accessed: 2023-2-12.

[81] World Health Organization. Depression. https://www.who.int/news-room/fact-sheets/detail/depression. Accessed: 2023-2-12.

[82] World Health Organization. Schizophrenia. http://www.who.int/news-room/fact-sheets/detail/schizophrenia. Accessed: 2023-2-12.

[83] Igor Pantic. Online social networking and mental health. *Cyberpsychology, Behavior, and Social Networking*, 17(10):652–657, 2014.

[84] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[85] James W Pennebaker, Ryan L Boyd, Kayla Jordan, and Kate Blackburn. The development and psychometric properties of LIWC2015. https://repositories.lib.utexas.edu/bitstream/handle/2152/31333/LIWC2015_LanguageManual.pdf. Accessed: 2023-2-8.

[86] James W Pennebaker, Ryan L Boyd, Kayla Jordan, and Kate Blackburn. The development and psychometric properties of liwc2015. Technical report, 2015.

[87] Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, October 2014. Association for Computational Linguistics.

[88] Adriana Pietramala, Veronica L. Policicchio, Pasquale Rullo, and Inderbir Sidhu. A genetic algorithm for text classification rule induction. In Walter Daelemans, Bart Goethals, and Katharina Morik, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 188–203, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

[89] David MW Powers. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. *arXiv preprint arXiv:2010.16061*, 2020.

[90] Sunil Kumar Prabhakar, Harikumar Rajaguru, Kwangsub So, and Dong-Ok Won. A framework for text classification using evolutionary contiguous convolutional neural network and swarm based deep neural network. *Front. Comput. Neurosci.*, 16:900885, 2022.

[91] Wisam A Qader, Musa M Ameen, and Bilal I Ahmed. An overview of bag of words; importance, implementation, applications, and challenges. In *2019 international engineering conference (IEC)*, pages 200–204. IEEE, 2019.

[92] Chowdhury Mofizur Rahman, Ferdous Ahmed Sohel, Parvez Naushad, and S. M. Kamruzzaman. Text classification using the concept of association rule of data mining. *CoRR*, abs/1009.4582, 2010.

[93] Carlos Ramisch. N-gram models for language detection. *M2R Informatique-Double diplˆome ENSIMAG–UJF/UFRIMA*, 2008.

[94] Roann Munoz Ramos, Paula Glenda Ferrer Cheng, and Stephan Michael Jonas. Validation of an mhealth app for depression screening and monitoring (psychologist in a pocket): Correlational study and concurrence analysis. *JMIR MHealth UHealth*, 7(9):e12051, 2019.

[95] Andrew G. Reece, Andrew J. Reagan, Katharina L. M. Lix, Peter Sheridan Dodds, Christopher M. Danforth, and Ellen J. Langer. Forecasting the onset and course of mental illness with twitter data. *Scientific Reports*, 7(1), October 2017.

[96] Lorenzo Rosasco, Ernesto De Vito, Andrea Caponnetto, Michele Piana, and Alessandro Verri. Are loss functions all the same? *Neural Computation*, 16(5):1063–1076, May 2004.

[97] Muhammad Salihin Saealal, Mohd Zamri Ibrahim, David J Mulvaney, Mohd Ibrahim Shapiai, and Norasyikin Fadilah. Using cascade CNN-LSTM-FCNs to identify AI-altered video based on eye state sequence. *PLoS One*, 17(12):e0278989, 2022.

[98] Serhad Sarica and Jianxi Luo. Stopwords in technical language processing. *PLOS ONE*, 16(8):1–13, 08 2021.

[99] Psy D Sharon Saline. ADHD statistics: New ADD facts and research. https://www.additudemag.com/statistics-of-adhd/, October 2006. Accessed: 2023-2-12.

[100] Judy Hanwen Shen and Frank Rudzicz. Detecting anxiety through reddit. In *Proceedings of the Fourth Workshop on Computational Linguistics and Clinical Psychology — From Linguistic Signal to Clinical Reality*, pages 58–65, Stroudsburg, PA, USA, 2017. Association for Computational Linguistics.

[101] Ian M. Soboroff, Charles K. Nicholas, James M. Kukla, and David S. Ebert. Visualizing document authorship using n-grams and latent semantic indexing. In *Proceedings of the 1997 Workshop on New Paradigms in Information Visualization and Manipulation*, NPIV '97, page 43–48, New York, NY, USA, 1997. Association for Computing Machinery.

[102] Illés Solt, Domonkos Tikk, Viktor Gál, and Zsolt T Kardkovács. Semantic classification of diseases in discharge summaries using a context-aware rule-based classifier. *Journal of the American Medical Informatics Association*, 16(4):580–584, 2009.

[103] David Sundby and Lund. Spelling correction using n-grams. 2010.

[104] Harini Suresh, Jen J Gong, and John V Guttag. Learning tasks for multitask learning: Heterogenous patient populations in the icu. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 802–810, 2018.

[105] Johan AK Suykens and Joos Vandewalle. Least squares support vector machine classifiers. *Neural processing letters*, 9:293–300, 1999.

[106] T Team. Pandas development pandas-dev/pandas: Pandas. *Zenodo*, 21:1–9, 2020.

[107] C. Thomas, Vlado Keselj, N. Cercone, Kenneth Rockwood, and Elissa Asp. Automatic detection and rating of dementia of alzheimer type through lexical analysis of spontaneous speech. volume 3, pages 1569 – 1574 Vol. 3, 02 2005.

[108] Calvin MacDonald Thomas. Rating dementia of alzheimer type through lexical analysis of spontaneous speech. Master's thesis, Dalhousie University, 2019.

[109] Robert Thorstad and Phillip Wolff. Predicting future mental illness from social media: A big-data approach. *Behavior research methods*, 51:1586–1600, 2019.

[110] Sho Tsugawa, Yusuke Kikuchi, Fumio Kishino, Kosuke Nakajima, Yuichi Itoh, and Hiroyuki Ohsaki. Recognizing depression from twitter activity. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, page 3187–3196, New York, NY, USA, 2015. Association for Computing Machinery.

[111] John H Tucker. Status update:" i'm so glamorous". *Scientific American*, 303(5):32–32, 2010.

[112] Vladimir N Vapnik. An overview of statistical learning theory. *IEEE transactions on neural networks*, 10(5):988–999, 1999.

[113] Vladimir N. Vapnik. *Methods of Pattern Recognition*, pages 123–180. Springer New York, New York, NY, 2000.

[114] Tanu Verma, R. Sam Renu, and Deepti Gaur. Tokenization and filtering process in rapidminer. *International Journal of Applied Information Systems*, 7:16–18, 2014.

[115] Matthew Walenski, Thomas W. Weickert, Christopher J. Maloof, and Michael T. Ullman. Grammatical processing in schizophrenia: Evidence from morphology. *Neuropsychologia*, 48(1):262–269, January 2010.

[116] Michael L. Waskom. seaborn: statistical data visualization. *Journal of Open Source Software*, 6(60):3021, 2021.

[117] Adam B Wilcox and George Hripcsak. The role of domain knowledge in automating medical text report classification. *Journal of the American Medical Informatics Association*, 10(4):330–338, 2003.

[118] Jacek Wolkowicz, Stephen Brooks, and Vlado Kešelj. Midivis: Visualizing music pieces structure via similarity matrices. In *Proceedings of the 2009 International Computer Music Conference, ICMC'09*, pages 53–6, 2009.

[119] Jacek Wołkowicz, Zbigniew Kulka, and Vlado Kešelj. N-gram-based approach to composer recognition. *Archives of Acoustics*, 33(1):43–55, 2008.

[120] Jiaqi Xiong, Orly Lipsitz, Flora Nasri, Leanna M.W. Lui, Hartej Gill, Lee Phan, David Chen-Li, Michelle Iacobucci, Roger Ho, Amna Majeed, and Roger S. McIntyre. Impact of COVID-19 pandemic on mental health in the general population: A systematic review. *Journal of Affective Disorders*, 277:55–64, December 2020.

[121] Liang Yao, Chengsheng Mao, and Yuan Luo. Clinical text classification with rule-based features and knowledge-guided convolutional neural networks. *BMC Medical Informatics and Decision Making*, 19(S3), April 2019.

[122] Andrew Yates, Arman Cohan, and Nazli Goharian. Depression and self-harm risk assessment in online forums. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2958–2968. Association for Computational Linguistics, 2017.

Appendix A

Implementation details

The list of modules and their respective version required for the experiments are mentioned in Table A.1.

Packages	**Version**
Keras	2.8.0
Matplotlib	3.1.1
NLTK	3.4.5
NumPy	1.21.2
Pandas	1.3.5
Regex	2022.6.2
Scikit-learn	1.0.2
Seaborn	0.9.0
TensorFlow	2.8.0

Table A.1: List of packages and their respective version

Brookside, *Calvert*, and *Waverley* servers were used that share a centralized disc space and carried out code execution. The servers were provided by Computer Science Department at Dalhousie University [77].