Active Depression Monitoring and Alert System in Reddit through Machine Learning

Group 14:

Goh Wen Wei Victor - **A0067229Y**

Ken Cheah - **A0218906Y**

Krishnan Ananth - **A0218894M**

Tommy Kangdra - **A0218866N**

Xiao Yidi - **A0218962W**

Abstract

In this paper, we aim to design a social media depression monitoring and alert system using a combination of machine learning and heuristics algorithms. In Phase 1, we attempt to classify social media comments from Reddit into three distinct categories: Normal, Depressed, and Suicidal. In Phase 2, we use these classification predictions in our rule-based algorithms to identify, monitor and intervene should a user's mental state deteriorate past a certain threshold. We compare five models and find that, with our limited dataset, the Bag-of-Words Logistic Regression model performs the best. While preliminary results are promising, more and better medically-curated / medically-informed data is needed to accurately assess the feasibility of such an undertaking. Link to Github can be found in Appendix A.

1. Introduction

Mental health is observed as the largest cause of disability globally. In a shocking report from the World Health Organization (WHO) [1], around 300 million people suffer from depression in 2015 globally. Some signs accompanying people with depression are characterized by affective symptoms (e.g., depression moods, flat emotions, heightened irritability, low self-esteem), cognitive symptoms (e.g., difficulties in concentrating and do work properly) and somatic symptoms (e.g., change in appetite, sleeping pattern). As depression progresses, it may lead to more severe consequences such as suicide thoughts, self-harm, or even suicide attempts.

The recent emergence of Covid-19 has caused abrupt changes in the daily lives of many. Covid-19 affects the livelihood of people in a lot of aspects e.g., financially (lost income and, confining people to their own houses and restricting people from certain recreational activities (such as hobbies or going on vacation). These significant changes exacerbate the dire mental health crisis that already plagues individuals globally[2]. Therefore, addressing mental health during this pandemic should be placed into the international and national public health priority in improving overall mental wellbeing.

Despite the severity of the problem and very many conversations / awareness events regarding mental health being shared in public, only a small proportion of depressed individuals actually received minimally adequate treatment. In a paper published by Psychiatric Service Checroud and colleagues[3], one of the top reasons for not getting treatment for depression is the lack of awareness of available treatment or remedy. A lack of motivation also contributes to a depressed individual not seeking the help they need.

Much like any other mental health illness, depression is a complex beast – While a person needs to be clinically diagnosed by a mental health specialist who performs a clinical assessment by considering a wide variety of factors, it usually requires the self-awareness and self-motivation of the person to consult a specialist in the first place. Often times, depression goes undiagnosed and untreated to a point where medical intervention becomes urgently required. Therefore, more pro-active approaches should be done to reach out to these individuals and to put forward the necessary information of help and treatment that they can access.

One of the growing interests today is in identifying depression through social media with machine learning. It is found from a study conducted in 2016 [4], that the amount of time of social media use is significantly positively correlated with increased depression. Furthermore, social media platforms such as Reddit, Twitter, or Facebook, amongst others, operate to create a safe space for individuals to anonymously share their feelings and opinions honestly. Based on these findings, we initiate this project to identify people who have depressive, self-harming or suicidal mental states through analysis of their comments in social media, and intervene when a certain threshold is met or when the severity progressively deteriorates past a certain threshold.

2. Scope & Objective

The aim of this project is to develop a classification model that accurately identifies depressed and suicidal users based on their textual postings and highlights intervention is required if severe deterioration in the mental states of individuals is detected through recorded activity. This would facilitate a more pro-active approach in reaching out to both depressed and suicidal individuals as it has been found that depression increases the risk of suicide [5].

When individuals have been suffering from major depressive episodes and are active on social media, the posting activity leaves an imprint of their current mental state. Therefore, through the posting activity, the developed classifier should be able to highlight posts indicating depression and suicidal intent. For this, the strongest indicator is likely to be obtained from certain keywords and phrases, or usage of certain adjectives. The model should also be able to identify changes in text content that identify a worsening of the individual's state from only feeling depressed to becoming actively suicidal.

This would require the ability to detect the degree of negative feelings and the text content to one that denotes active participation in events leading to self-harm, and this could be derived from changes in verb and adverb usage.

3. Dataset

3.1. Data Source

Our class labels include Normal, Depressed and Suicide comments. In this project, we use Reddit as our example social media platform to illustrate the concept of our intervention system. Reddit is chosen as the data source for this project given the mature pushshift.io API [6] and the large number of active users in "r/depression" and "r/SuicideWatch" subreddit groups. These form the backbone and source text of our Depression and Suicide classes, respectively. For the Normal class, we have pulled data from the "r/confession" subreddit as we feel it represents conversational language about daily life. We opted against using completely irrelevant and unrepresentative subreddits (for example: secondhand car sales advertisements) because we feel they do not reflect the true test distribution should the system be deployed in production, and also because it would make our classification task overly trivial.

The submissions are then collected and converted to csv file for future processing steps. Since pushshift.io only allows 100 entries per call, the timestamp (*created_utc*) of last fetched entry of each call is used as a part of search key for the next call for continuous data collection. The following features are fetched in each API call:

Table 1: Features table extracted from pushshift.io API

	Variable Name	Variable Description	Variable Type
Author Related Information	author	Author name	String
	author_fullname	Author ID	String
Submission Details	created_utc	Created time	Int (Unix time)
	id	Submission ID	String
	full_link	Link to the submission	String
	title	Title of submission	String
	selftext	Body text of the submission	String
Response Details	num_crossposts	No of cross-posts	Int
	num_comments	No. of comments	Int
	score	Score	Int
	total_awards_received	Total awards received	Int
	upvote_ratio	Upvote ratio	Int
Regulation Related Details	over_18	If the submission is SFW/NSFW	Boolean
	sticked	If the submission is stickied	Boolean
	locked	If the submission is locked	Boolean

3.2. Data Description

3.2.1. Text Analysis (Phase 1)

For each subreddit mentioned in Section 3.1, we fetch 10 thousand submissions posted during early COVID-19 outbreak (February 2020 – March 2020). To facilitate manual labeling, the following entries were removed from the raw dataset:

- Submissions without meaningful content (e.g. deleted posts and posts with less than 10 words)
- Submissions with more than 250 words

Data Labelling

Although we acknowledge clinical diagnosis to be the only method of accurately determining whether an individual is depressed or not, we believe there is much to be gained by the early identification and detection of depression and suicidal intentions from a user's posts and comments found on social media as a proxy for the user's current state of mind. A wide net is cast, and we aim to tune our models to tolerate a higher number of false positives whilst keeping our false negatives to a minimum. In effect, it is much more detrimental to have missed raising the alarm on high-risk individuals at the expense of inconveniencing some.

In preparing our dataset, we randomly shuffled all comments into a single document. Three human labellers then go through the text of each post and classify them into one of three labels: '0' - normal, '1' - depressed, '2' - suicidal, based on each labeller's understanding of what amounts to a depressed or suicidal post. Each labeller's labels were then hidden to the other labellers to prevent bias or influence. A comment was classified as the determined label if a simple majority, i.e., 2/3, was reached. Should a post be split down the middle, the post is discarded from the dataset. After a grueling 60 hours of manual labelling, we acquired a preliminary dataset with 1,687 observations which were broken down into 992 observations of the Normal class, 401 observations of the

Depression class and 294 observations of the Suicidal class.

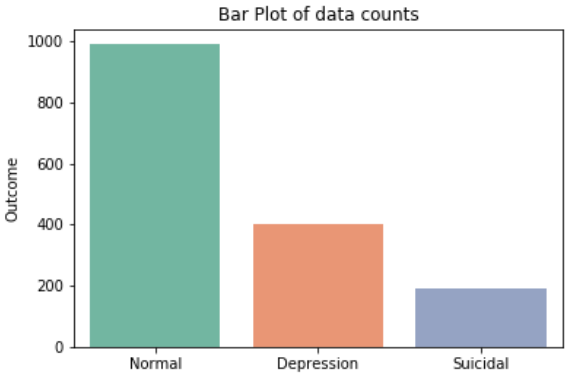


Figure 1. Bar plot of data counts for each class

3.2.2. User Behavior Analysis (Phase 2)

In preparation for Phase 2, a user list was generated from commenters of "r/SuicideWatch" and "r/depression" according to the results of Phase 1. Historical submissions and comments of each user in the user list were then fetched for user behavior analysis.

3.3. Explanatory Data Analysis

Explanatory data analysis was conducted to gain a better understanding on the nature of depressive expressions and suicidal expressions, as well as some characteristics and insights of corresponding post authors.

In terms of word counts, more uniform distribution was observed with depression and suicidal class, while normal class was observed more with less wordcounts (distribution peak at ~30 words).

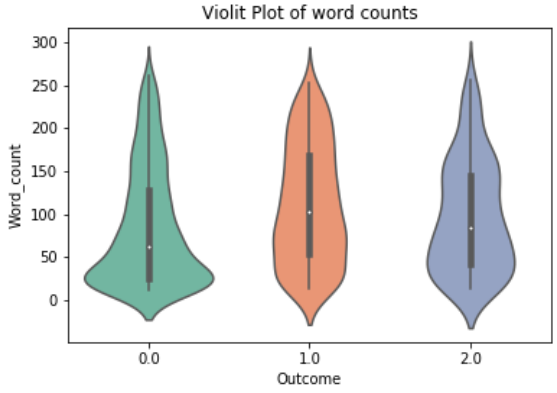


Figure 2. Violin plot of word counts from each class

To further understand the distribution of words used in each class, the text was transformed with Count Vectorizer and TFIDF Vectorizer, PCA (nclass=2) and was visualized through scatter plot as shown in figure 3. The distribution of words overlap significantly with one another both in Count Vectorizer and TFIDF Vectorizer transformer and is hardly distinguishable. This became one of our major challenge in training machine learning algorithm to distinguish between classes.

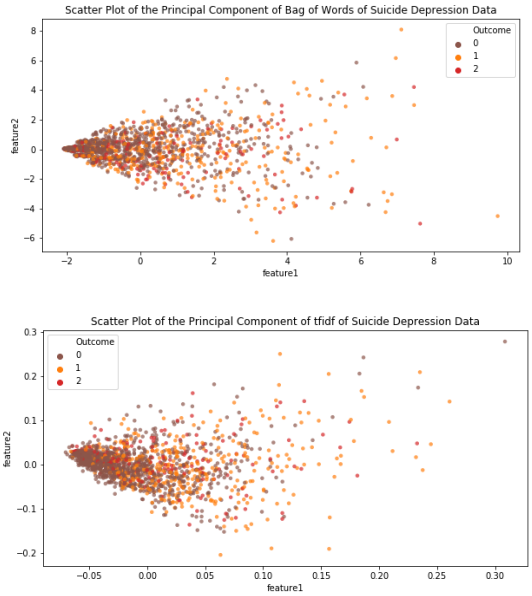


Figure 3. Scatter Plot with PCA (nclass=2). Figure (above) showed the scatterplot with bag of words, Figure (below) showed the scatterplot with TFIDF vectorizer

Word clouds were generated to visualize most common words that occur in each class. Similar most common words were observed through the text from different class, e.g., know, people, life, feel. This indicates the similarity of the text from each class.

To have an overview of words that may be important words to distinguish between class, we fit logistic regression to the bag of words (ngram_range = (1,3)) and take the top 10 most important words.

In Normal class, the important words that we can observe is we, he, her, you, in, him. This was an interesting observation, that normal class can be distinguished from the word that is pointed towards other. This is in line with our understanding that depressed / suicidal classes have tendencies of focusing on themselves in contrast with normal class that may focus on other people.

Furthermore, in depression class the important words are wish, depression, suicidal, everything, anything, empty, alone, pain. These words represent comments which

exhibit helplessness, a cry for help, depressive loneliness, lack of any motivation, or being empty and numb to their emotion.

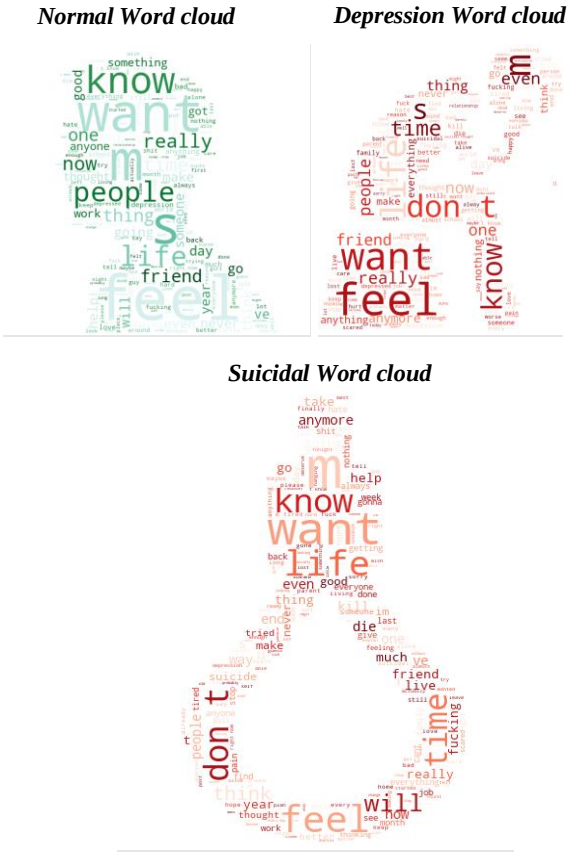


Figure 4. Word cloud of each class

Lastly, the suicidal class important words consist of pills, guns, goodbye, acetamol, jump in which the authors have contemplated ways in which they wish to end their lives. Other words such as goodbye, gonna, ready, bye, will express their readiness to act on the plan. Overall, the post in this class exhibited self-harming tendencies, suicide planning and execution. For people to act on their plan, it takes a lot of consideration and therefore only after struggling for a while that the person may be ready to follow through.

Table 2. Top 10 most important words

Class	Top 10 Most Important Words
Normal	We, he, her, you, in, him, something, or, at, what
Depression	Wish, depression, suicidal, everything, anything, empty, alone, pain, just want, my family
Suicidal	Pills, gun, goodbye, gonna, ready, ways, acetamol, bye, jump, will

4. Methodology

4.1. Preprocessing

To enable our models to learn better, the reddit posts were preprocessed with the following steps:

- Removing URL, punctuations, digits, '_', white spaces and other non-letter characters
- Lowercasing
- Removing stopwords
- Lemmatizing

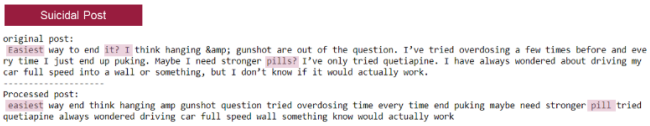


Figure 5. Pre-processing output of text

4.2. Feature Engineering

Different feature engineering techniques were conducted for different models. Four techniques were used for bag-of-words (BOW) models and two for deep learning (DL) models.

4.2.1 Name Entity Recognition

Named Entity Recognition (NER) was used to locate and extract named entities in the Reddit posts like date, time, cardinals, ordinals, name of people and more to provide more information about the class it is representing. This was engineered for BOW models.

SpaCy library was used to model the NER. The input to the model is the reddit post and the output is identified tags in the posts as shown in figure 6. There are a total of 18 types of tags in the train dataset. We then count the frequency of each tag in the post, and this is a feature engineered to the BOW model with 18 columns as dimension.

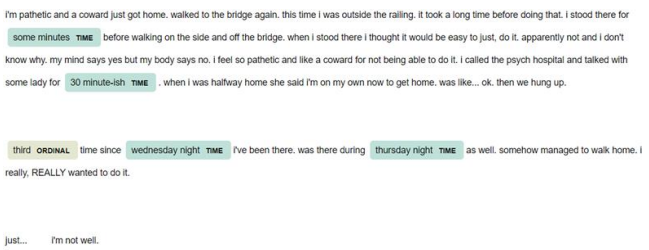


Figure 6. NER of reddit post

We observe that the Suicide class has more 'date' tags like 'tomorrow', 'few days', 'last year', etc. compared to other classes as shown in figure 7, figure 8 and figure 9. This is an indication that Suicide comments show some level of

urgency where users intend to attempt suicide / self-harm in the immediate future and this was not the case for Depressed or Normal comments.

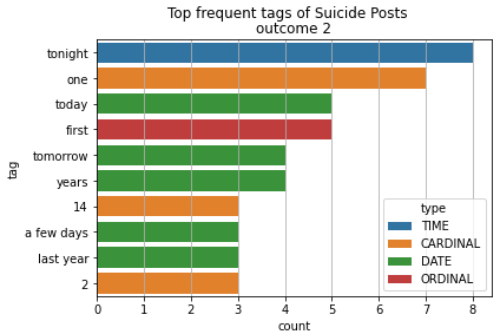


Figure 7. NER of suicide posts

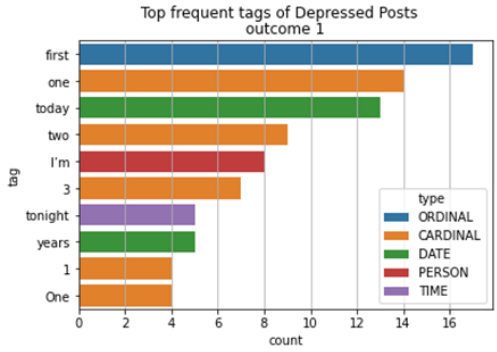


Figure 8. NER of depressed posts

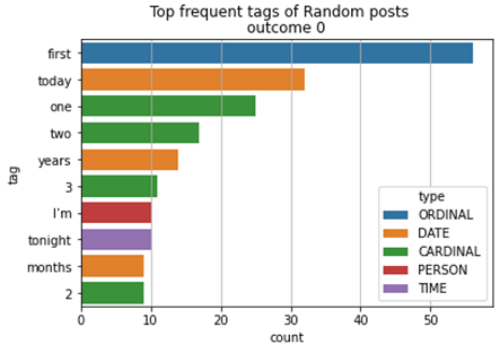


Figure 9. NER of normal posts

4.2.2 Sentiment Score

For each comment, we generated a sentiment score using the Vader and Textblob packages. This was engineered for BOW models. The sentiment score ranges between -1 and 1. The motivation behind this feature is that suicidal posts may have more negative sentiment scores than depressed and normal posts, this is shown in figure 10.

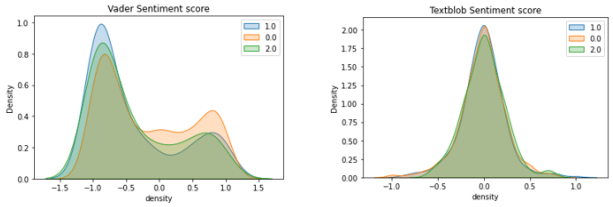


Figure 10. Vader and Textblob sentiment score

4.2.3 Topic Modelling

Topic modelling was performed to observe and find bunch of words in the posts to extract information as topic features. Latent Dirichlet Allocation (LDA) package was used in the project with the number of topics as the hyperparameter. We first tokenize each post and create a dictionary containing the number of times a word appears in the training set. This is then passed as the input to the LDA model, and the output is topics with words occurring in it as shown in figure 11.

We observed BOW models to perform the best with 15 topics as hyperparameter as compared to 10 or 20 topics. Hence, in the project we have selected 15 topics contributing to 15 features. The motivation behind this feature engineering is that each topic will provide more insights for each class when trained with BOW models.

In figure 11, we observe topic '0' is more related to 'depression' class since it contains words such as 'depression'. Topic '1' could be related to class 'suicide' or 'normal' based on the set of words in the topic.

```
Topic: 0
Words: 0.012*"like" + 0.011*"depression" + 0.010*"know" + 0.010*"year" + 0.009*"feel" + 0.008*"time" + 0.008*"depressed" + 0.008*"think" + 0.008*"day" + 0.007*"get"

Topic: 1
Words: 0.013*"like" + 0.011*"go" + 0.010*"feel" + 0.010*"want" + 0.008*"day" + 0.008*"know" + 0.008*"time" + 0.007*"nothing" + 0.007*"thought" + 0.007*"one"
```

Figure 11. Topic modelling as features

4.2.4 TF-IDF

'TF' refers to term frequency and 'IDF' refers to inverse document frequency. The objective of this feature engineering is to convert text to numbers when trained with BOW models. A score is assigned to each word as shown in equation 1. The technique is useful to extract keywords and information of each class. In the project, we use n_grams = (1,2) to extract single words and word pairs with a minimum frequency of 2 posts.

$$tf_{idf(t,d)} = tf(t,d) * (\log\left(\frac{1+N}{1+df(t)}\right) + 1) \qquad (1)$$, where N is the number of posts, $tf(t,d)$ is the number of times the

word *t* appears in the post *d*, *df(t)* is the number of posts containing word *t* [7].

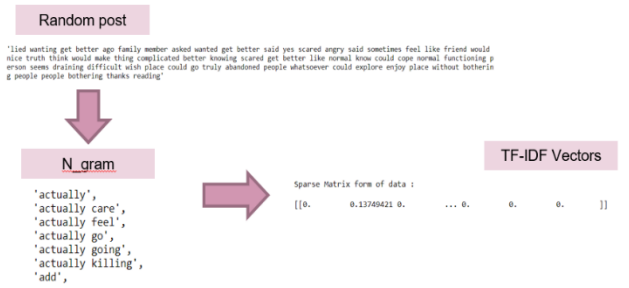


Figure 12. Tf-IDF vector of posts

4.2.5 Text Augmentation

Text augmentation allows deep learning models to generalize better on the test set by introducing noise. It also increases the dataset size, which greatly improves deep learning model performance. Three types of text augmentation strategies, namely Synonym Replacement, Contextual Word Replacement and Back Translation, were applied to the source text for deep learning models. Synonym Replacement generates a new post for each existing post in the dataset by randomly replacing word or phrases according to its synonyms. Contextual Word Replacement generates new posts from existing posts by replacing some random word or phrases according to BERT pretrained embeddings. Back Translation generates posts by translating existing posts to another language and then translating back to English. After augmentation steps, three new posts were generated for each original post and the total size of training dataset has grown to four times of the original dataset.

4.2.6 Word2vec

An embedding matrix was built according to the word vectors from Word2vec model which gave vector space according to their semantic similarity. Figure 13 shows the steps taken to build word embedding from Word2vec model.

According to the vocabulary from training set, a Word2vec model was built with python package *genism*. We empirically set the number of dimensions of word vector as 50 for the best model performance. Meanwhile, word index was created from the training data and all the input texts were converted into integer sequences. For each word, the row in the embedding matrix corresponding to the word's index was then filled with its word vector.

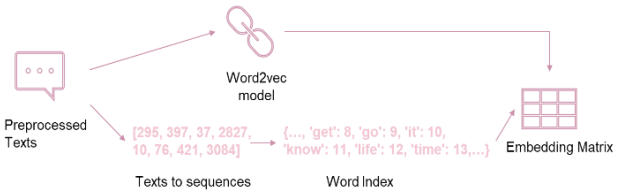


Figure 13. Word2Vec workflow

4.3. Models

In this project we will compare five model's performance:

1. Naïve-Bayes classifier
2. Logistic Regression (LR) classifier
3. Support Vector Machine (SVM) classifier
4. CNN Long Short-Term Memory (CNN-LSTM) model
5. Bidirectional Encoder Representations from Transformers (BERT) model

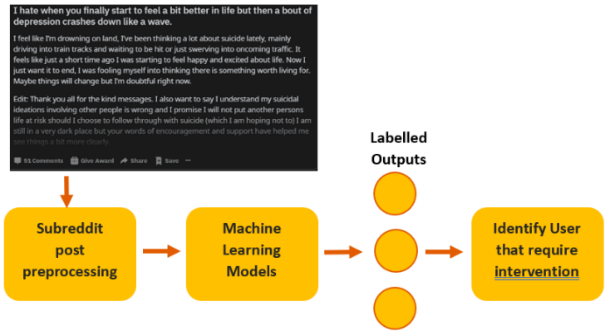


Figure 14. Model Framework

Bag of Words Models:

Naïve-Bayes classifier**:** This is a simple model that is known to work well on text data, and so was selected as one of our candidate models. In essence, the model classifies texts based on probability of events. The comment text is split into words which are independent events to generate the Term-Document Matrix (TDM) for each class. The model requires less training time compared to other models.

Logistic Regression: A simple and linear classifier for text classification that is easy to implement, interpret and efficient to train.

SVM classifier: SVM works well for text classification. It looks to find the optimal decision boundary between the class vectors. Hence, the comments' tokens are converted into vectors before applying the model. SVM works well when there is a clear margin of separation between classes

and is memory efficient. It might not work well when there is noise, and the target classes overlap.

Deep Learning Models:

CNN- LSTM[8]**:** A CNN-LSTM model generally contains two parts: Convolutional Neural Network (CNN) layers for text feature extraction, and LSTM layers for sequential dependencies exploration. The multiclass classification CNN-LSTM model built in this project consists of an embedding layer, a 1D convolution layer, Maxpooling layer, LSTM layer, and Dense layer. Figure 15 illustrates the structure of this CNN-LSTM model.

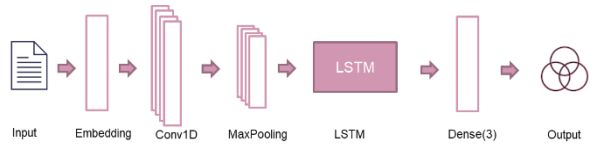


Figure 15. Structure of CNN-LSTM model [8]

Raw data is pre-processed and converted to sequence of integers (i.e. word index) with same length before feeding into the model. Padding is used to ensure all posts are of the same size. The first layer (i.e. embedding layer) matches the input with embedding matrix generated from Word2vec model and feeds the input to convolutional layer.

The parameters of CNN-LSTM model are listed in Table 3.

Table 3. Model Parameters of CNN-LSTM

Parameter Name	**Value**
Conv1D Kernel Size	3
Optimizer	Adam
Number of Iteration	5
Dense Activation	Softmax

The parameters of grid search are listed in Table 4, and the best parameters are in bold.

Table 4. Grid Search Parameters of CNN-LSTM

Parameter Name	**Value**
Learning Rate	0.1, 0.05, **0.01**, 0.005
Number of Conv1D Filter	**32**, 64
Dimensionality of the LSTM output space	**40**, 80
Number of Epochs	3, **5**, 10

BERT**:** The BERT model is a state-of-the-art language model that can learn the semantics of language better than a simple bag-of-words model. This is because it learns the sequence of words in relation to each other using bidirectional contextual learning through a mechanism known as self-attention. In using BERT, the text is first preprocessed into the shape and form expected by the model, that is, into 3 embedding vectors of tokens, segments and positions. These are passed into the model simultaneously and, as such, can benefit from the parallel computation speed-up of GPUs/TPUs (which result in much faster training compared to sequence models such as RNNs/LSTMs).

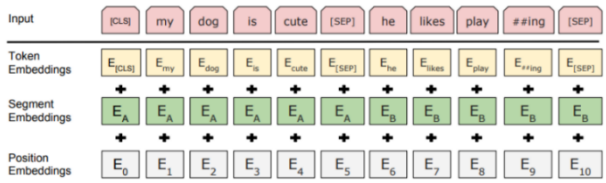


Figure 16. BERT Input Embeddings

Traditionally, BERT is pre-trained using two self-supervised tasks, masked language modelling (MLM) and next sentence prediction (NSP). In our implementation, however, we have decided to use a variant of BERT known as ELECTRA. ELECTRA is a recent development of BERT pre-trained as a discriminator using Replaced Token Detection (RTD) instead of MLM. In essence, instead of randomly masking certain words in a given sentence, these words are replaced with plausible "fakes". The model then tries to determine which words have been replaced or kept the same [9]. This is illustrated in the figure below.

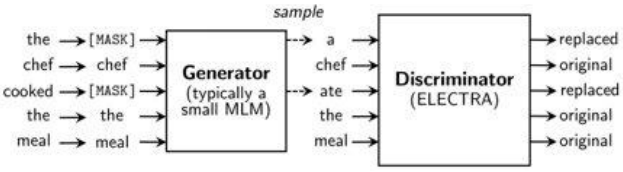


Figure 17. ELECTRA Model Illustration

Recent benchmarks have shown this to be very effective, and have produced very efficient models; outperforming other models of similar complexity.

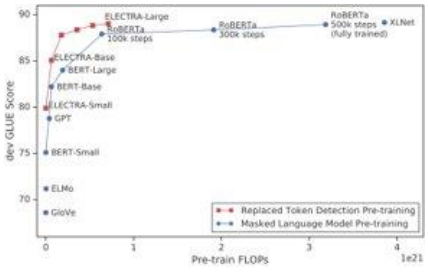


Figure 18. BERT Transformer GLUE Score Benchmark

We have selected ELETRA-Small as our representative BERT model due to its size-to-performance ratio and our resource constraints. The model architecture comprises 12 Layers, and a Hidden Size of 256, with a benchmark GLUE score that is nearly 5% higher than BERT-Small.

Transfer learning was performed by finetuning weights loaded from models trained on Wikipedia and BooksCorpus data. We have decided to keep all hyperparameters in line with literature recommendations but substituted the simple categorical cross-entropy loss for focal loss. As our PCA analysis above suggests, there is significant overlap in classes and the model should be adjusted to account for the fact that some classes should be penalized more for a wrong prediction in our imbalanced dataset and giving less weight to easily classified examples (i.e. the majority Normal class). Ultimately, this helps ensure a better F1-score for our downstream multiclass classification task.

4.4. Evaluation

In evaluating the performance across our five models, we have focused primarily on the F1-score, the recall rate, and the confusion matrix. F1-scores are a good singular metric that balances the precision and recall of a model. We also ensure that the recall rate for each class is high to ensure the model was accurately predicting true labels for actual true labels in the data. Finally, we scrutinize the confusion matrix and take into account the false negative and false positive classifications.

From the results, logistic regression performs the best in terms of F1-score and based on the confusion matrix (Appendix B). Deep learning methods, however, do not fare well. Zooming in, we see that the logistic regression model balances false negatives for the suicidal class and false positives for the depression class better compared to other models. We also note that our model should minimize misclassifying depressed or suicide comments as normal comments since this would have a huge impact on our intervention measures. To reiterate what was mentioned above, we would err on the side of caution to catch as many cases of suicide and depression comments as possible at the cost of intervening where a normal comment is posted. In this respect, the potential cost of self-harm or loss of life far outweighs the cost of inconvenience to users.

Users in the test dataset are identified and their posts are extracted from pushshift.io API. For all the user's posts collected, the model prediction is obtained. The predictions are analyzed in a chronological order of time. We will propose manual rule-based approach and obtain polarity score for sentiment analysis on how to identify if the user's depression has worsened or turned into a suicidal condition.

5. Applications: Active Monitoring Application

5.1. Business case: Intervention for users at risk

With the development of the classification model complete, it can now support the proactive approach mentioned earlier. An active monitoring system based on this which identifies depressed and suicidal users can be deployed on social media platforms such as Reddit or Twitter as an intervention measure.

As mental health concerns rise and more people succumb to their suicidal thoughts, social media platforms have a newfound responsibility to society to bridge the gap between those in dire need of help with the proper help channels. We envision a system packaged as a Corporate Social Responsibility (CSR) product of a social platform that will 'intervene' after a user posts his comments.

In cases such as these, the comment text is classified after its posting and predictions for each user's comments are logged chronologically. According to the severity of the classification and based on the user's history of comments, intervention may take the form of a text prompt, for example: "Help is available. Speak with someone today" for severe and urgent cases, or "Feeling down? Why not speak to someone today?" for less severe cases. This could then be followed up with the hotline number of a local mental health support organization such as Samaritans of Singapore or the ability to chat with them through a chat window on the platform anonymously. [10]

In any case, we do not think it is ethically appropriate to have a 'hard' intervention system by submitting users details to any authority as it would breach data privacy laws and undermine users' trust in an anonymous and open platform. Instead, we believe fostering partnerships with mental health organizations, support groups and crises intervention hotlines to be the key to success as we work towards a common goal.[11]

5.2. Monitoring System: Methodology

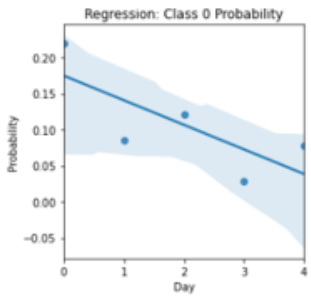
The monitoring system will take a heuristic rule-based approach based on three main rules which take into account the classification of comments, and the probability that a given comment is Normal, Depressed, or Suicidal. Comment history for each user in a certain period will be processed sequentially through these three rules and be flagged out as requiring attention and the urgency of

intervention depending on diagnosis of the user's mental state. The lookback period determines the relevant period prior to the current post where the comment history of the user will be reviewed, which is set to 90 days for the initial implementation of the system as while the results vary, existing research indicates the typical duration of depressive episodes to be at least 3 months [12]. For simplicity, a month is taken to be 30 days, leading to a lookback period of 90 days.

The **first rule** checks if a user has had any recent posts classified as suicidal posts. If so, the user will be identified as suicidal and requiring immediate attention and assigned priority 1.

The **second rule** first checks if a user has less than 5 posts as, if this is so, the posting history cannot be used to assess the trend in mental state and early intervention is important as it might be too late to intervene if more posts are required. It also checks if any posts have probability of being classified suicidal of at least 0.3 as this indicates high suicidal attributes in the post even if the post itself is classified as depressed or normal. If so, the user will be identified as having potential suicidal intent and requiring attention and assigned priority 2.

The **last rule** checks for 3 conditions: if a user has at least 5 posts, has any posts with probability of being classified as suicidal of at least 0.3, and the gradient of the trend of the probability of its posts being normal is at most 0.1 when the user's post history over time is fit under a linear regression. Illustrating the idea of the gradient of trend in figure 19, suppose over a period of 5 days the posts are recorded as per table in figure 19. Plotting the linear regression of the probabilities indicates a negative gradient for the trend of its Class 0 probability of the posts, which would be less than 0.1.



	pred_lr_0_prob
0	0.220072
1	0.084567
2	0.121692
3	0.028601
4	0.077912

Figure 19. 5-day post sample

In combination, **the logic for the last rule** checks for heavily depressed users which are not returning to normalcy and users meeting this rule will be identified as requiring monitoring and assigned priority 3.

The priority determined for each user will determine the message of intervention. With priority 1, a message will be sent to the user encouraging them to immediately seek help. This message would be sent again after 3 days. With priority 2, the message would encourage them to reach out for help if they feel overwhelmed. With priority 3, the message would simply let the user know that help is always available if needed. All users with priority would receive follow up on avenues of support as mentioned in section 5.1.

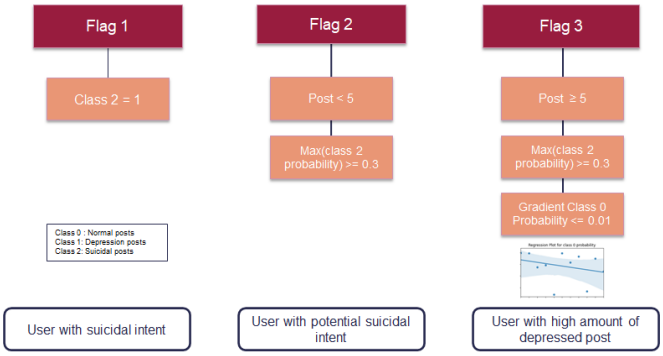


Figure 20. Summary of rules in monitoring system

Through the rule-based approach, the monitoring system determines if intervention is required for users and acts as an early warning system for users that are at risk of self-harming or becoming suicidal.

This proposed framework flagged 23.57% of users with suicidal intent, and 30.0% of users with potential suicidal intent and high amount of depressed posts out of 420 users which were observed with at least 1 depressed class post. Review of posts from a sample of users indicate that the framework is acceptable although more trials would be required.

6. Conclusion, Limitations & Further Improvements

In Phase 1, we determine our best classifier to be the Bag-of-Words Logistic Regression model with the inclusion of 5 feature engineering techniques, i.e., TF-IDF, Vader Sentiment Analysis, Textblob Sentiment Analysis, LDA Topic modelling, and Named-Entity Recognition (NER). In our experiments, the deep learning models did not perform well. This is likely due to the small dataset as deep learning models are known to require massive amounts of data to perform well.

In Phase 2, we tap into our domain knowledge of the Reddit platform and rely on mental health literature to inform our rule-based algorithm design choices. Our proposed monitoring and intervention system tracks mental health deterioration and severity, and flags users as being Depressed or Suicidal with three priority levels of intervention based on a lookback window of their comment history.

In this project, we have experimented and demonstrated a 2-phase mental health monitoring and intervention system for social media platforms. However, preliminary results are still far from ideal.

Dataset

To help our models perform better, we require many times more observations than what we currently have for training. The dataset has to be medically-curated or sourced from a trustworthy authority such as from research labs. As mental health is complex in nature, and as machine learning models can only be as good as the dataset it learns from, specialists or domain experts should rightly be the deciding authority on whether a comment amounts to a Depressed or Suicidal classification.

Forecasting

In our project, our system monitors users' comments in real-time and considers historical comments using a lookback window. However, it is likely possible to design a forecasting model to identify users who are predicted to have deteriorating mental states in the future and to prevent depressive episodes or attacks before they even take place. Predicting user trends would likely require more data points than is currently available through free and publicly accessible platform APIs, and would require the platforms themselves to make available internal data such as "total/average user time on depression subreddit", "platform browsing history", "user account statistics", etc.

Experiments and Evaluation

Finally, there is no true evaluation method that can be used to assess the quality of our Phase 2 results. Indeed, it would be preposterous to identify and label if a given user had actually self-harmed or committed suicide. However, we may be able to adjust our algorithms based on the click-through rates or engagement rates of users when they click-through the prompts generated by the website. Experiments such as these can be used to objectively assess if our methods are in the correct direction or a complete overhaul needs to be ordered.

Appendix A Github Link to the data and code

https://github.com/tommykangdra/Bt5153-Group-14-Active-Depression-Monitoring-and-Alert-System-in-Reddit-through-Machine-Learning

Appendix B. Confusion Matrix & Classification Report

SVM

Model: SVM		Predicted		
		Normal	Depression	Suicidal
Actual	Normal	161	35	3
	Depression	29	48	3
	Suicidal	10	21	8

	precision	recall	f1-score	support
0.0	0.81	0.81	0.81	199
1.0	0.46	0.60	0.52	80
2.0	0.57	0.21	0.30	39
accuracy			0.68	318
macro avg	0.61	0.54	0.54	318
weighted avg	0.69	0.68	0.67	318

NB

Model: NB		Predicted		
		Normal	Depression	Suicidal
Actual	Normal	84	83	32
	Depression	9	60	11
	Suicidal	1	20	18

	precision	recall	f1-score	support
0.0	0.89	0.42	0.57	199
1.0	0.37	0.75	0.49	80
2.0	0.30	0.46	0.36	39
accuracy			0.51	318
macro avg	0.52	0.54	0.48	318
weighted avg	0.69	0.51	0.53	318

LR

Model: LR		Predicted		
		Normal	Depression	Suicidal
Actual	Normal	152	38	9
	Depression	20	54	6
	Suicidal	4	20	15

	precision	recall	f1-score	support
0.0	0.86	0.76	0.81	199
1.0	0.48	0.68	0.56	80
2.0	0.50	0.38	0.43	39
accuracy			0.69	318
macro avg	0.62	0.61	0.60	318
weighted avg	0.72	0.69	0.70	318

CNN-LSTM

Model: CNN-LSTM		Predicted		
		Normal	Depression	Suicidal
Actual	Normal	166	24	9
	Depression	42	25	13
	Suicidal	20	5	14

	precision	recall	f1-score	support
0	0.73	0.83	0.78	199
1	0.46	0.31	0.37	80
2	0.39	0.36	0.37	39
accuracy			0.64	318
macro avg	0.53	0.50	0.51	318
weighted avg	0.62	0.64	0.63	318

BERT

Model: BERT		Predicted		
		Normal	Depression	Suicidal
Actual	Normal	152	38	9
	Depression	21	55	4
	Suicidal	12	14	13

	precision	recall	f1-score	support
0	0.82	0.76	0.79	199
1	0.51	0.69	0.59	80
2	0.50	0.33	0.40	39
accuracy			0.69	318
macro avg	0.61	0.59	0.59	318
weighted avg	0.70	0.69	0.69	318

Reference

[1] Suicide. (2019). https://www.who.int/news-room/fact-sheets/detail/suicide

[2] Notivol, J.B., Garcia, P.G., Olaya, B., et al. (2021). Prevalence of depression during the COVID-19 outbreak: A meta-analysis of community-based studies. International Journal of Clinical and Health Psychology, 21(1).

[3] Checkroud, A.M., Foster, D., Zeutlin, A.B., et all. (2018). Predicting Barriers to Treatment for Depression in a U.S. National Sample: A Cross-Sectional, Proof-of-Concept Study. Psychiatr Serv, 69(8).

[4] De Choudhury, M., Gamon, M., Counts, S., & Horvitz, E., et all. (2013). Predicting Depression via Social Media. Proceedings of the International AAAI Conference on Web and Social Media, 7(1).

[5] Does depression increase the risk for suicide? https://www.hhs.gov/answers/mental-health-and-substance-abuse/does-depression-increase-risk-of-suicide/index.html

[6] API Documentation. https://pushshift.io/api-parameters/

[7] Tfidf Transformer. https://scikitlearn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfTransformer.html

[8] Text Sentiments Classification with CNN and LSTM. https://medium.com/@mrunal68/text-sentiments-classification-with-cnn-and-lstm-f92652bc29fd

[9] More Efficient NLP Model Pre-training with ELECTRA. https://ai.googleblog.com/2020/03/more-efficient-nlp-model-pre-training.html

[10] Hotlines - SuicideWatch. https://www.reddit.com/r/SuicideWatch/wiki/hotlines

[11] Tan, R. (2021). How to support someone who may be suicidal. https://www.sos.org.sg/blog/how-to-support-someone-who-may-be-suicidal

[12] Spijker, J., De Graaf, R., Bijl, R., Beekman, A., Ormel, J., & Nolen, W. (2002). Duration of major depressive episodes in the general population: Results from the Netherlands Mental Health Survey and Incidence Study (NEMESIS). British Journal of Psychiatry, 181(3), 208-213. doi:10.1192/bjp.181.3.208