Suicide and Depression Detection Using an LLM Reinforced BERT Model

Christopher Cheung, Jorge Vigil, and Srikar Viswanatha
Georgia Institute of Technology CS 4650

Abstract

Our project focuses on developing a robust model for detecting suicide and depression-related sentiments from an extensive dataset, leveraging both traditional and transformer-based NLP approaches. By incorporating reinforcement learning with a large language model (LLM) like Gemini, we enhanced our BERT-based model's ability to handle context-lacking and ambiguous sentences, achieving improved accuracy and showcasing its potential for real-world mental health applications.

1. Introduction

Suicide and depression are significant public health challenges, with suicide ranking among the leading causes of death worldwide. According to recent statistics, over 700,000 individuals die by suicide annually, and millions more experience severe depressive episodes (Malhortra & Jindal, 2022). Social media has become an increasingly prominent space where individuals express their emotions, making it a critical avenue for identifying signs of mental distress. In this paper, we aim to develop a model that can accurately identify suicidal thoughts and depressive text from social media posts, enabling timely intervention to help individuals in crisis.

One of the key challenges in this endeavor lies in the subtlety of language used by individuals experiencing suicidal ideation. Texts can be cryptic or veiled in dark humor, which poses the risk of misclassification if not handled carefully. It is essential to distinguish between genuine distress and expressions of dark humor to avoid false alarms while maintaining sensitivity to signs of danger. Additionally, there is a nuanced difference between expressions of depression and suicidal intent; capturing these distinctions is vital to ensure appropriate responses.

Throughout this project we aim to conduct a comparative analysis of transformer-based models, such as BERT, and traditional machine learning models, including Logistic Regression and Naive Bayes, for detecting suicidal ideation and depressive expressions from user input. By levering both traditional models (as a baseline) and other transformer-based approaches, the research evaluates each approaches effectiveness in capturing the nuanced and often subtle language of mental distress.

1.1 Related Work

Traditional machine learning models have been extensively used for detecting suicidal ideation and depression. Jain et al. applied models like Logistic Regression and Random Forest to classify social media posts, relying on manually engineered linguistic features such as tokenization and stemming (Jain et al., 2022). Similarly, Li et al. used Random Forest to analyze clinical interviews, leveraging linguistic markers like past tense and anger words (Li et al., 2023). Both studies illustrate the utility of traditional machine learning approaches for detecting mental health signals, but they also highlight significant limitations. These models often fail to account for the complexities of language, such as sarcasm, cryptic expressions, or shifting context. Additionally, the dependency on feature engineering and domain-specific expertise makes these models less scalable and harder to adapt to diverse datasets or languages.

These limitations highlight the need for transformer-based models like BERT and LLMs, which excel at understanding word relationships and contextual nuances without extensive feature engineering. Pretrained on large datasets, transformers offer scalability, adaptability, and improved accuracy, making them well-suited for diverse and complex mental health applications. Transitioning to transformer-based approaches could enhance the detection of subtle indicators of suicidal ideation and depression, supporting more effective intervention strategies.

1.2 Data Collection

Our data comes from 2 different Kaggle datasets, which are linked in the references. The main dataset used is the Suicide Sentiment Analysis Dataset, which has around 16,000 data points, labeled as "Depression", "Neutral", or "Suicidal". We mainly utilized this dataset due to the extra label of "Depressed", with the

intent that we could train our model to distinguish between suicidal sentiment.

The other dataset is the Suicide-Watch dataset. There are over 230,000 datapoints, however they are only categorized into "Suicidal" and "Non-Suicidal".

For our preprocessing, we noticed that some data in the CSV's were unlabeled, and since the labels were separated into their own tables, we wrote a script which would append, label, and shuffle the data in preparation for model training and testing.

2. Models

2.1 Non-Transformer Models

We have decided to implement a few Non-Transformer Models in order to understand their strengths and weaknesses compared to transformer-based architectures.

For word tokenization, we utilize Bag of Words mainly for its simple implementation, as well as for the fact that this is a baseline model implementation. Our research shows that a majority of research in this area utilizes neural embeddings for word tokenization, due to its ability to learn and reweigh word vectors based on their frequencies and surrounding contexts.

2.1.1 Naive Bayes

For our baseline non-transformer model, we implemented Naive Bayes, not only because of its simple yet accurate architecture, but the fact that it is a probabilistic model. We can reason that certain texts that are classified as suicidal or depression would have a higher probability of negative words such as "sad", "lonely". Naive Bayes utilizes the probabilities of these words to classify it into the 3 sentiments. Our testing accuracy came to around 84.3% with this application. Below is a confusion matrix showing the correct and incorrect classifications the model made:

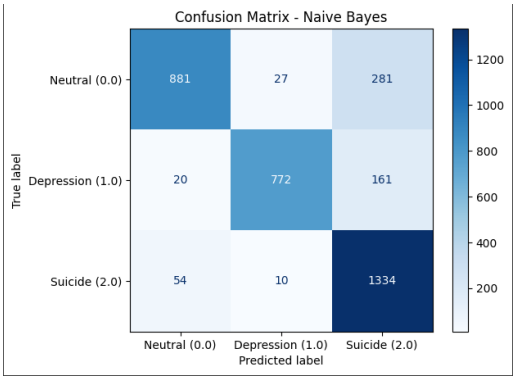

Figure 1: Confusion Matrix for Naive Bayes Model

We see that a majority of the mistakes with this model come with it predicating "Suicidal" sentiment, when it is actually Neutral. This is most likely caused by context issues and dark humor / sarcastic tones that the model cannot understand strictly from a probabilistic standpoint. Though an accuracy of 84% is satisfactory, we can achieve a much higher accuracy through transformer architecture, as well as neural word embeddings.

```
True Label: Neutral
Predicted Label: Suicidal
Input Text:  THANKYOU!  <3 iloveyoutwoooo
--------------------------------------------------
True Label: Neutral
Predicted Label: Suicidal
Input Text: Circling Uranus looking for Klingons.
--------------------------------------------------
True Label: Neutral
Predicted Label: Suicidal
Input Text: I would have won if it wasn't for a Fusilli mistakes!
--------------------------------------------------
True Label: Depression
Predicted Label: Suicidal
Input Text: I do not know what to do.
--------------------------------------------------
True Label: Depression
Predicted Label: Suicidal
Input Text: ! I read a lot here. I decided to write myself.
--------------------------------------------------
```

Figure 2: Naive Bayes Model Mismatches

2.1.2 LSTM

Another non-transformer model we decided to implement was a Long Short Term Memory (LSTM) network. By maintaining a memory of prior inputs through their cell state, LSTMs can process information across longer sequences, making them particularly valuable for understanding the nuanced and context-dependent language often associated with mental health indicators. They can discern patterns in phrasing, such as repetitive negative expressions or subtle shifts in tone, which may signal depressive or suicidal tendencies. Their gating mechanisms further enable LSTMs to focus on critical words or phrases while filtering out irrelevant information, adeptly handling the emotional and linguistic complexity of such texts. When combined with pretrained word embeddings, LSTMs enhance their ability to identify implicit cues and semantic associations, which could make it strong model for solving the problem we are addressing. As LSTM models are neural network based, they re-evalulate their weights based on sequential context. This makes them better for sentiment analysis than Naive Bayes, as it uses some form of context in its classification instead of pure probabilities. From the confusion matrix generated within the notebook we observe that the LSTM model performs well in detecting "Neutral" and "Suicide" sentiments, with high accuracy for both categories, though it struggles with misclassifying "Suicide" as "Neutral." Detection of "Depression" is moderately accurate but shows confusion with "Suicide," likely due to overlapping emotional tones.

2.2 Transformer Models

To address the limitations of context detection which include dark humor/sarcastic tones we have decided to implement a transformer based NLP model. Typ-

ically, transformer models perform well in NLP due to their self-attention mechanisms, capturing relationships between words regardless of distance, which enables a deep understanding of context. Transformer models are pretrained on extensive datasets allowing them to provide a robust foundation that can be fine-tuned to perform specific tasks with high accuracy and efficiency.

2.2.1 BERT

For our transformer model, we decided to implement a Bidirectional Encoder Representations from Transformers model. We chose this model specifically because of its advanced ability to understand complex language patterns, addressing key limitations of our baseline Naive Bayes model. BERT's bidirectional processing enables it to understand context by analyzing surrounding words, effectively handling polysemy and ambiguous language. Its pretraining on a vast dataset equips it to capture deep semantic relationships and subtle linguistic cues, crucial for detecting depressive or suicidal sentiments. By leveraging neural embeddings and BERT's transformer architecture, we address traditional models' limitations in handling subtle sentiments and misclassifications. This transition from Naive Bayes to BERT enhances the accuracy and reliability of our detection system, providing a more robust understanding of complex language dynamics. Below is the confusion matrix showing the correct and incorrect classifications the model made:

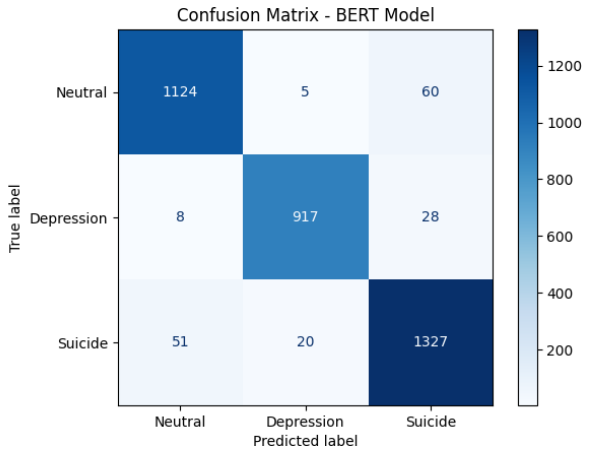

Figure 3: BERT Confusion Matrix

It can be seen that the BERT model demonstrates strong performance, with high accuracy across "Neutral," "Depression," and "Suicide" sentiments. It correctly classifies the majority of instances, with minimal misclassifications, such as 5 "Neutral" texts as "Depression" and 28 "Depression" texts as "Suicide." Compared to prior models, BERT excels in handling nuanced language and context, significantly reducing errors, particularly between closely related sentiments like "Depression" and "Suicide."

3. Further Analysis

3.1 User Input

After implementing the BERT model and observing its high accuracy, we wanted to see if our model would perform well on user-generated sentences. This user-input section was intended to provide challenging examples for our model to see if it would provide the correct classification. A challenging input would be something that either has sarcasm or little given context, both of which would confuse the model. With little context, there is not enough for the model to classify with, and with sarcasm, sentences which may seem suicidal from a simple word analysis, actually convey a different meaning. For example, the paper provides a quote: "If I want to commit suicide, I will climb to the height of your ego and jump". We can see with the first couple of words how a misclassification can happen. In fact, this is a major area of study in NLP - training models to learn context and sarcasm.

3.2 LLM-based Reinforcement Learning for BERT

Our idea to train our model to be able to properly classify challenging model inputs includes simultaneously prompting better trained models such as a therapist-based LLM, and utilizing their classification to train our model. Some of our inputs were user-generated, and some were obtained from the Suicide-Watch dataset, which we thought would be tricky for the model to classify. Whenever our model classifies something as "Depression" or "Suicidal", we ask the Gemini LLM to analyze the sentiment of the prompt. If there is a difference in sentiment analysis between the BERT model and the Gemini LLM, then the "correct" LLM label as well as the user input are stored in a CSV, and periodically our model would be additionally trained with the new examples. We define the "correct" label to be the one classified by the Gemini LLM. Our intuition behind this is that by utilizing an LLM with a more powerful sarcasm and context detection capability than our model, we can better reinforce our model on those specific sentiments.

3.2.1 User-Input Model Fine-Tuning

We chose to utilize Gemini Pro 1.5 as the API for our LLM assistance. With this, we made a prompt that we would feed into the model that asked if a given user input was expressing either suicidal, depressive, or neutral sentiment. We found that from multiple user inputs, that Gemini is much better at analyzing sentences with ambiguous meaning, such as "I'm not sure what I want to do next in life, just figuring things out". As said before, a problem with NLP classification is context and sarcasm detection ability - two things which our BERT model struggled on. With the LLM enhancement, we were able to get correct sentiment analysis.

One problem from our initial approach is that after training the model on the CSV, the model would still output the initial sentiment pre-training. To fix this, we

did 2 things: Firstly, we increased the frequency of entries that each sentence-label data had in the CSV. Instead of having 1 entry for the sentence-label, we added it 50 times to the CSV. This was to better tune the weights on the user data. Secondly, we fine-tuned the data with twice as many epochs as the original training (6 versus 3), which allows the model to go over the dataset multiple times, ensuring that the word vectorizations and weights to our model are tuned well enough to correctly classify the sentence. Doing this allowed us to obtain the expected sentiment from a given user input after fine-tuning the model on the Gemini classification.

3.2.2 Fine-tuned Model Comparison

A problem we thought that would arise from this method was that the model would be overfit on the user input data. We ran the model on the test data before and after fine-tuning it on a CSV file of over 100 user-input model misclassifcations. Before training, we achieved a 95.1% accuracy, and after the training we achieved a 95.5% accuracy. Though this might seem minuscule, the progress was substantial as our model is now better trained on sentences that do not have a lot of context (under 5 words), as well as sentences that have ambiguous sentiment. Our data set, though it includes sentences like these, include a lot of either very generic or very strongly depressive or suicidal sentences. By fine tuning our model on user-based input, we not only allow our model to be more accurate on real-world sentence examples, but for it to also be more conclusive on sentences lacking context and exhibit ambiguity.

4. Conclusion

Overall, our exploration into utilizing sentiment analysis in order to classify suicidal and depressive sentences was extremely successful. We observed the models that worked best as per the collective study, and utilized the best-working elements in terms of model and word tokenizations. We compared these with a baseline model, and understood the shortcomings and strengths of both models. With Naive Bayes, though a probabilistic approach to sentiment analysis will work, it cannot capture true context and the relations that arise between words, and struggles with context-lacking and confounding sentences. Understanding that the most accurate and used models and tokenizers were attention models and neural word embeddings respectively, we used these in our Transformer-based approach. We decided to utilize BERT, and obtained an accuracy of 95.1%. However, when testing our model on custom user input, we realized that it did not do well in general. For example, sentences with little to no context, such as "I am happy", was classified as suicidal, and that sentences such as "I am not depressed", were also being misclassified. After utilizing LLM reinforcement learning with Gemini Pro, we were able to get the accuracy of the model up to 95.5%, and able to train our model to be better on obtaining context and sentiment from context-lacking and ambiguous sentences. Through our LLM reinforcement learning, we were able to show positive results for a major problem in Natural Language Processing.

4.1 Further Work

Further work would be to improve our model to obtain an accuracy of around 96-97%, which would make its performance better than the current State of the Art Models, which are around 96%, utilizing more powerful models than BERT. We believe that this is achievable utilizing our current method of reinforcement learning. Methods that could help us to reach this goal would be mainly to increase our dataset. We only worked with around 16,000 data entries, and generated another 5000 through the LLM reinforcement learning. If we could work with a larger dataset, we would obtain a higher accuracy. Additionally, the quality of the dataset could be in question. When reading through some of the mismatches, it seems that some of the labels were misclassifed. With a higher quality dataset, this would not be a problem and could lead to a higher accuracy.

4.2 Results

Overall, we believe that LLM-reinforced fine tuning can help to address the problem of context and sarcasm detection in NLP. Additionally, we believe that our work can have a meaningful impact in helping to detect suicidal and depressive thoughts early in individuals, and providing them with the help they needed, especially with the advent of social media and its rising popularity. Through this work, we have demonstrated that leveraging advanced NLP techniques and reinforcement learning can lead to tangible improvements in sentiment analysis, providing a promising foundation for developing systems that can better understand and respond to critical mental health challenges. Below is a summary of the models we showcased and their accuracies.

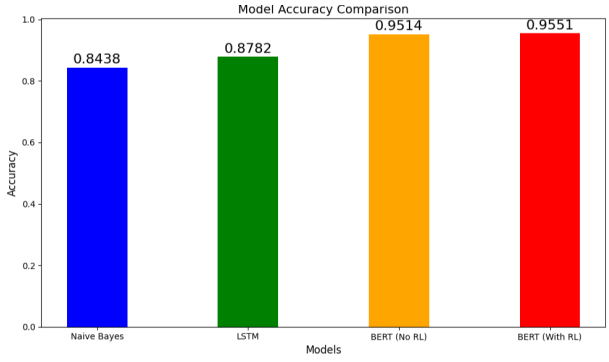

Figure 4: Model Accuracy Comparison

5. Dataset

https:/www.kaggle.com/datasets/umar1103/suicide-sentiment-analysis-dataset

References

Jain, P., Srinivas, K. R., & Vichare, A. (2022). Depression and suicide analysis using machine learning and NLP. In Journal of Physics: Conference Series (Vol. 2161, No. 1, p. 012034). IOP Publishing.

Li, T. M., Chen, J., Law, F. O., Li, C. T., Chan, N. Y., Chan, J. W., ... & Wing, Y. K. (2023). Detection of suicidal ideation in clinical interviews for depression using natural language processing and machine learning: cross-sectional study. JMIR medical informatics, 11(1), e50221.

Malhotra, A., & Jindal, R. (2022). Deep learning techniques for suicide and depression detection from online social media: A scoping review. Applied Soft Computing, 113, 109713. https://doi.org/10.1016/j.asoc.2022.109713