SafeChat - Final Report

Word Count: 1951

Permissions

	Video	Final report	Source code
Junming Zhang	Yes	Yes	Yes
Jiakai Shi	Yes	Yes	Yes

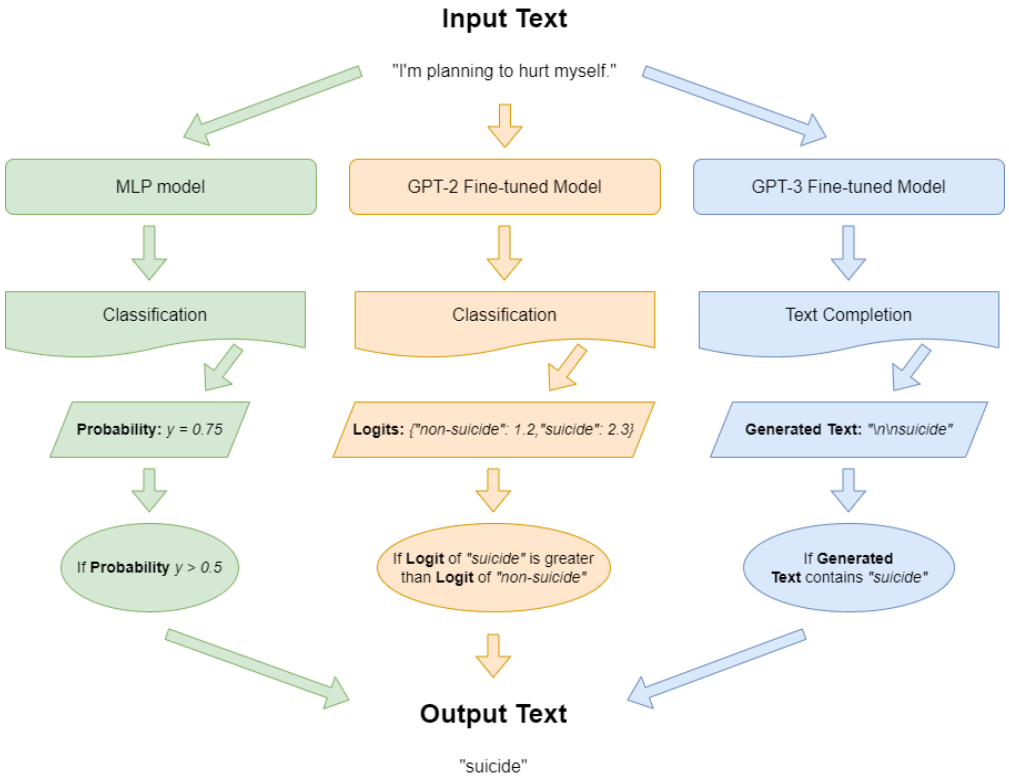

Figure 1 ***(Illustration/Figure)***: A diagram that illustrates the overall model working process.

Introduction

Large Language Models (LLMs) have exhibited remarkable results in various tasks such as understanding human language. Some studies suggest that LLMs can help psychological diagnosis. For example, a paper “Identifying Suicide Ideation and Suicidal Attempts in a Psychiatric Clinical Research Database using Natural Language Processing” shows that NLP approach can classify suicide ideation by extracting information from electronic health records. It is interesting to experiment on how NLP can be applied to maintain mental health, which is useful in discovering psychological disorders in time. Therefore we were inspired by these studies and then developed an application in psychological therapy.

Our goal is to classify whether a user/generated message has self-harm related contents with GPT-3/2, a cutting-edge transformer-based NLP model family capable of multiple tasks including text classification and generation. Our classification follows these guidelines:

- Messages are labelled as “suicide” if they 1) express suicidal thoughts, for example, “I want to die”; 2) include potential suicidal actions, for example, “recently, I have a lot of negative thoughts”.
- Messages are labelled as “non-suicide” if they 1) formally discuss suicide, for example, “The suicide ideation is a serious topic to our society”; 2) refer to other’s suicide, for

example, “It’s an unfortunate to hear someone’s suicide”; 3) are not relevant to suicide, for example, “how is your progress today”.

Background & Related work

- *Supervised Learning for Suicidal Ideation Detection in Online User Content*: This paper suggests the classification criteria of if a text implying a suicide ideation, the criteria has been listed in the introduction section. We labeled our dataset based on the criteria.
- *A Comparative Analysis on Suicidal Ideation Detection Using NLP, Machine, and Deep Learning*: This paper suggests a list of metrics to evaluate our models in suicidal ideation detection (by classifying a text to “suicide” or “non-suicide”), including both numeric scores like accuracy, precision, recall, and f1 score and illustrations like the word cloud and confusion matrix. We evaluate our models based on these metrics.

Data and Data Processing

We first extract messages from a large datastore of chat logs from an experiment of participants chatting with the OpenAI's GPT-3 model in 5 minutes to improve the mental health of participants. However, we only extract 30 “suicide” texts for limited topics of conversations about suicidal ideation in real-world experiments. Thus, the database is imbalanced compared to 163 “non-suicide” texts extracted from the conversation.

Then we simulate bot-to-bot conversations to collect more “suicide” texts. We use two GPT-3 models to simulate conversations. One GPT-3 model takes a designed prompt about a person who expresses suicidal thoughts and may have potential suicidal actions. We also try to add descriptions to create a persona, who is very impatient and hard to be coped with, thus we can collect more “suicide” texts within a single bot-to-bot conversation. We collected 104 “suicide” texts manually.

In addition, we find a public dataset on Kaggle with a huge amount of “suicide” texts and “non-suicide” texts from the #depression and #SuicideWatch posts on the Reddit Platform.

Next, we merge the datasets with the texts we extracted. During data cleaning, we remove texts with several tokens more than the maximum allowed by GPT models. Finally, we get 110K “suicide” and 110K “non-suicide” texts in total. Since GPT-3 has requirements on dataset format for both training and validation, we use the tool from OpenAI to prepare the dataset, i.e., convert the training and validation dataset in CSV format to JSONL format with the command “openai tools fine_tunes.prepare_data -f <LOCAL_FILE>”.
To train and validate the models, we split the dataset into the training and validation dataset in a ratio of 4:1. For fine-tuning the GPT-3 model, we only sample 1/10 of the training data without

replacement since the fine-tuning service of OpenAI fails to handle the whole training data. It is capable of fine-tuning the GPT-2 model with all data.

Model Architecture and Software

An overview of the model architecture is in Figure 1. We use three different models to perform the classification. Given a user input message, our baseline models (MLP and fine-tuned GPT-2) perform the classification task which computes the logits, whereas the fine-tuned GPT-3 model performs the text completion task which generates the most probable text.

We used the GPT-3 model provided by the OpenAI community, which provides APIs for fine-tuning (which can take the validation dataset for evaluation) and answer generation. GPT-3 is a transformer-based and auto-regressive language model with 175 billion parameters and capable of processing input and output context with a maximum of 2048 tokens in total. This model takes a prompt (and optionally examples) and generates the text based on the prompt. Since our task is classifying if an input text implies suicidal ideation, and the primary task of GPT-3 from OpenAI is text generation as a continuation of the given prompt, we fine-tuned GPT-3 model to specifically work for our classification task - the input of the model is one message, the output is the label “suicide” (positive) or “non-suicide” (negative). We choose “ada”, the fastest GPT3 version of OpenAI. To reproduce our result, just fine-tune models with our dataset.

Baseline

To compare with our formal model (GPT-3), we first construct a simple MLP model and then fine-tune GPT2ForSequenceClassification, GPT-2 a variation for text classification, provided by HuggingFace as two baselines. The rationale is that they are capable of text classification, where MLP is a naive text classifier, and GPT-2 is the last generation of GPT-3. We fine-tune these two models with the same dataset and then compare them with the fine-tuned GPT-3 with the same validation dataset. The MLP consists of one linear layer with input as tokens from GLoVe and output as the classification (logit) of the input text. GPT-2 is a transformer-based NLP model with 1.5 billion parameters. MLP outputs a list of logits, which will be converted to probability by the sigmoid function separately since logits are used to compute the loss. The model predicts “suicide” text if the converted probability is greater than 50%, otherwise, it predicts “non-suicide”. The output of the fine-tuned GPT-2 contains the logits of two labels (“non-suicide” and “suicide”). We derive the label with the maximum logit.

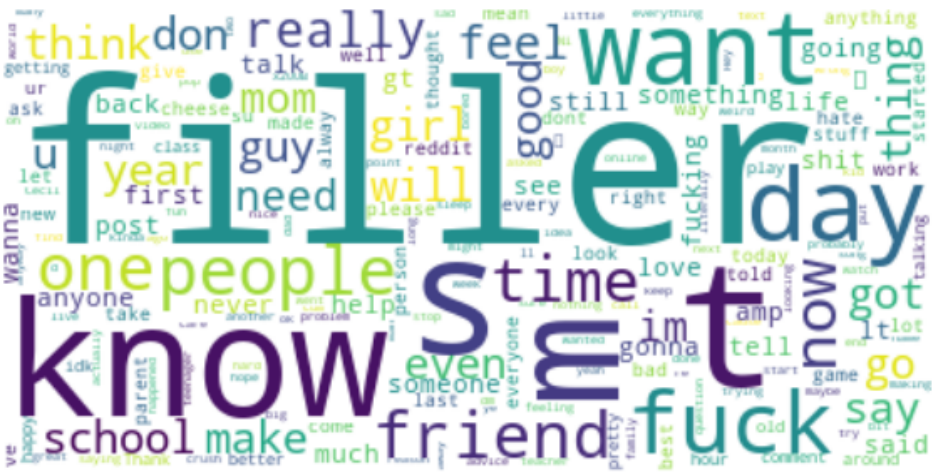

Figure 2: word-cloud of non-suicidal texts

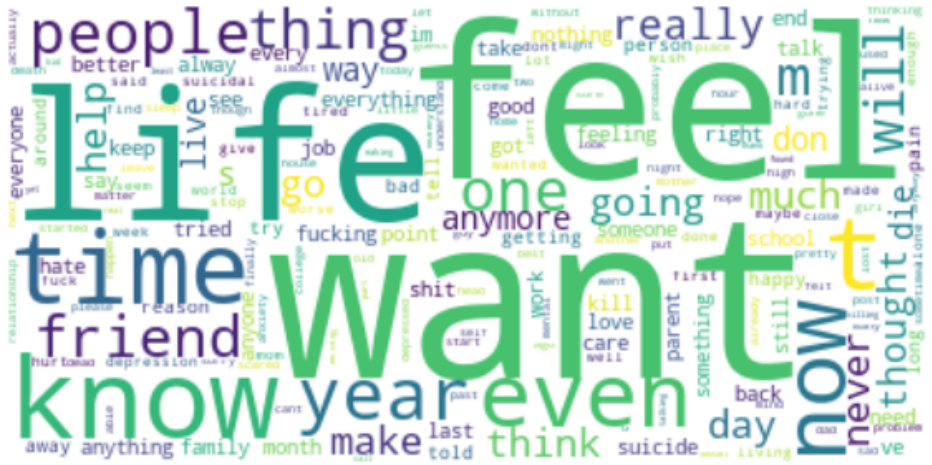

Figure 3: word-cloud of suicidal texts

Qualitative Results

We first review and analyze the whole dataset for training and validation. We compute the frequency of single tokens appearing in non-suicidal and suicidal texts so that we can evaluate the difference between the top 200 occurrences of tokens in the two vocabularies. Figures 2 and 3 illustrate the WordClouds of non-suicidal and suicidal texts respectively.

The WordCloud of suicidal texts includes terms like “die”, “pain”, “suicide”, and “kill”, which express suicidal thoughts and actions, and also negative sentiment words like “alone”, “hate”, and “death”. On the other hand, the WordCloud of non-suicidal texts includes relatively random terms, such as “filler”, “know”, and “day”, and also positive sentiment words such as “good”, “love”, “best”, however, it contains some similar words in suicidal WordCloud results as well, like

“fuck”, “shit”, “fucking”. Therefore, this could be one potential reason that causes false positive/negative predictions.

Quantitative Results

We choose the metrics accuracy, precision, recall, f1-score, and confusion matrix for evaluating binary-classification models in machine learning. These metrics help measure the models in making true/false positive/negative predictions. For simplicity, the higher the numeric scores, the better the model performs. Definitions of four scores are listed as follows.

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

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

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

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

From table 1, we see that the fine-tuned GPT-3 beats MLP and zero-shot GPT2/3 models far and performs slightly better than fine-tuned GPT-3 for all metrics. Although the zero-shot GPT-3 does not perform as well as the two fine-tuned models, it still beats zero-shot GPT-2 and MLP. It is interesting to see that the MLP beats the zero-shot GPT-2.

Table 1: evaluation metrics on classification for different models

	Trained MLP	GPT-2 (fine-tuned)	GPT-2 (zero-shot)	**GPT-3 (fine-tuned)**	GPT-3 (zero-shot)
Accuracy	0.709	0.990	0.548	**0.993**	0.814
Precision	0.833	0.995	0.780	**0.995**	0.804
Recall	0.522	0.984	0.113	**0.991**	0.830
F1	0.642	0.989	0.197	**0.993**	0.817

Figure 4 to 8 are confusion matrices of each model. **The fine-tuned GPT-3 achieves the highest true positive/negative**, and the fine-tuned GPT-2 achieves slightly lower true positive. Both of these two models achieve 1.0 in true negative and nearly 1.0 in true positive. From all matrices, we see that all models perform relatively well in true negative, but MLP and zero-shot

GPT-2 perform poorly in true positives (zero-shot GPT-2 achieves the lowest true positive rate), this could imply that our models are inclined to make negative predictions (i.e. “non-suicide”).

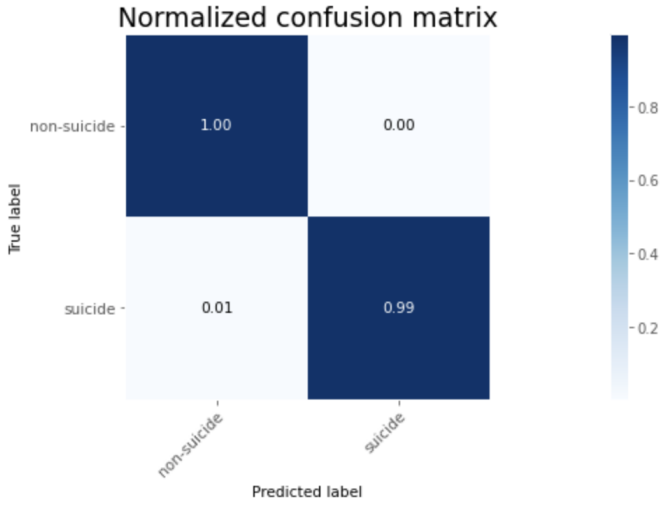

Figure 4: confusion matrix of fine-tuned GPT-3

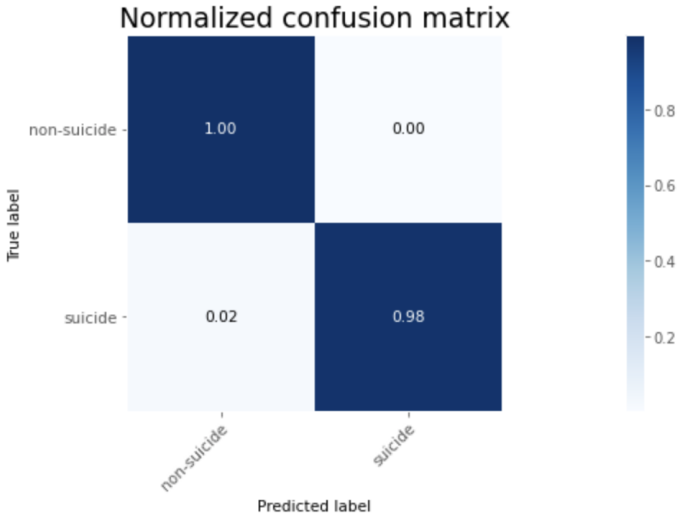

Figure 5: confusion matrix of fine-tuned GPT-2

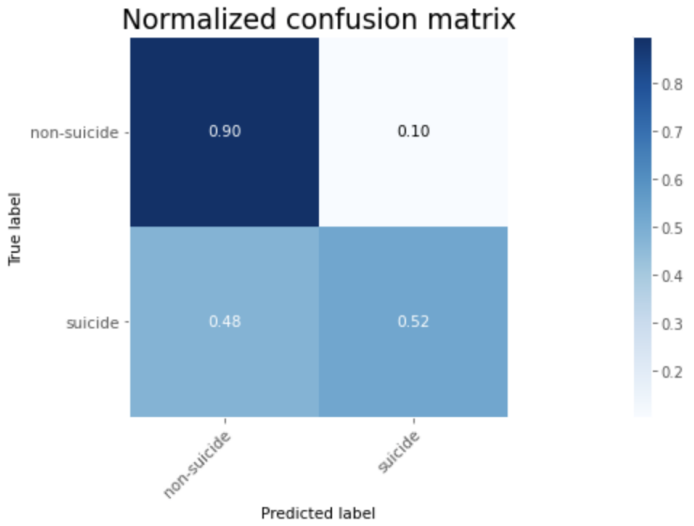

Figure 6: confusion matrix of trained MLP

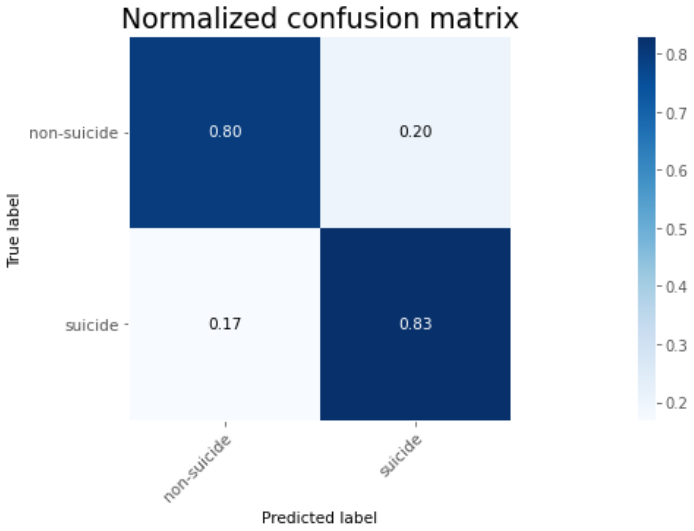

Figure 7: confusion matrix of zero-shot GPT-3

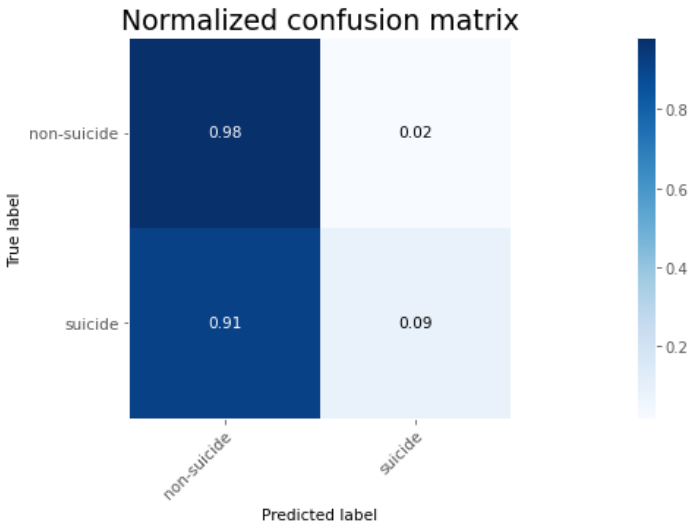

Figure 8: confusion matrix of zero-shot GPT-2

Discussion and Limitation

This work shows that the fine-tuned GPT models are powerful in binary classification tasks, even the zero-shot GPT-3 produces acceptable predictions. This work is an exploratory step to prove that the GPT model can be strong in classifying potential self-harm risks in mental health chatbot conversations if we have sufficient data to fine-tune. The result also implies that a simple trained MLP model is inadequate in such tasks, since the performance is unacceptable in comparison with the GPT-3 model. It is also interesting to observe both high true and false negative rates in the zero-shot GPT-2 model prediction.

However, there are a few limitations of this work. Firstly, Kaggle's public dataset occupies a large portion for fine-tuning and validations, where the models are trained and evaluated under the scope of specific texts so we cannot conclude that the models can achieve similar performance in other datasets. Secondly, the whole classifying process is completed without a real human in the loop so that there is no comparison between human and the model. Lastly, even though we define the “non-suicide” and “suicide” texts with different categories before preparing the data, we never collect the information about how many texts are falling into each categories. It is unclear for qualitative analysis. For example, whether the text is referring to other’s suicide or discussing suicide formally or irrelevant to suicide when the fine-tuned GPT models classify the text as “suicide”.

Thus, we can crowdsource labels from real participants by providing the same text to the models and then compare the differences. Additionally, we need to consider specifying the labels into more detailed categories. Finally, it is always necessary to have a real person (e.g. an expert in mental health) in the loop of the model's classifications since the model cannot achieve 100% accuracy.

Team Contribution

Jiakai Shi

- Modified original project proposal.
- hand labeled 193 data samples from chat log database.
- Designed prompt and Simulated bot-to-bot conversations and hand labeled 104 data samples.
- Collected dataset from Kaggle.
- Implemented user-facing side of the Flask application, including designing the navigation view and classification view.
- Implemented server side of the Flask application, including connecting OpenAI APIs for zero-shot, one/few-shot, and fine-tuned GPT-3/2.

Junming Zhang

- Split and cleaned the collected dataset for training and validation, including further dataset processing and split to fine-tune GPT-3
- Implemented baseline (GPT-2 & MLP) interfaces, including the prediction method of (zero-shot & fine-tuned) GPT-2 and fine-tuned GPT-3 in our application.
- Wrote scripts to fine-tune GPT2/3 and train MLP, and saved the models for evaluation & inference
- Searched for metrics to evaluate the models (quantitative analysis) and implemented the script to evaluate the performance of models before & after fine-tuning