TalkToMe

An Emotional Detection and Psychological Support Chatbot

Practical Language Processing

Practice Module

Semester Two 2021/2022

PREPARED BY:

Student Name	Student ID
Liu Ding	A0231429J
Mediana	A0231458E
Sarah Elita Shi Yuan Wong	A0231507N
Zeng Hanyu	A0231558A

Contents

1 Executive Summary 3
2 Market Research 4
3 System Architecture and Design 6
4 Pre-Deployment 8
4.1 Data Gathering and Pre-processing 8
4.1.1 ESConv Dataset for Conversation Model and Problem Classification Model 8
4.1.2 Kaggle Suicide Dataset for User Condition Classification Model 9
4.2 Training and Evaluation 9
4.2.1 Conversation Model 9
4.2.2 User Condition Classifier 10
4.2.3 User Problem Classification 11
5 System Implementation 14
5.1 TalkToMe Flow Chart 14
5.2 Conversation Model, Users' Problem and Users' Condition Classification Model 15
6 Conclusion and Future Work 17
References 18
Appendix A: Support Strategies Description 19
Appendix B: Project Proposal and User Guide Document 20

1 Executive Summary

In recent years, there has been an increasing trend of people being affected by feelings of depression and anxiety. This has been worsened by the various stresses due to the COVID-19 pandemic. According to the World Health Organization (WHO), the global prevalence of anxiety and depression had increased by 25% in the first year of the pandemic (World Health Organization, 2022). Prolonged periods of negative emotion might lead to more serious mental health conditions, such as depression, which can be fatal if left untreated.

Our team proposed TalkToMe, An Emotional Detection and Psychological Support Chatbot that can provide support to people who are feeling down and need comfort. TalkToMe is built using Natural Language Processing (NLP) techniques including BERT, Bi-LSTM and BlenderBot for modelling the conversations over multiple sessions, text generation and text classification. It uses the Emotional Support Conversation (ESC) framework (by Liu et al.) to support the general conversation with users.

TalkToMe interacts with user via Telegram, a widely used Messaging Platform. It has the following 3 main functions:

1. Chat with and comfort users who may be experiencing negative feelings (e.g., stress, anxiety, etc.) 24/7
2. Detect user's condition (risk score) and provide hotlines to seek professional help
3. Detect the category of the user's problem (e.g., emotional, work, partner relationship, friendship, school, family, and others)

To date, TalkToMe chatbot provides general emotional support to users and retrieval of hotlines or resources regarding mental well-being. In the future, we are looking at providing the following function:

- Specific conversation support and hotlines according to users' problem and their condition, including emotion type and intensity detected on-the-fly
- Text summarization of users' problem and aspect extraction of users' feedback for further analysis and insights to improve TalkToMe chatbot
- Rich responses support according to users' emotion and condition, including audio, video, picture, jokes, etc that can comfort them

2 Market Research

According to a survey published by Ministry of Health (MOH) regarding the willingness to seek help from a mental health hotline, it was reported that 50.1% of respondents would not consider seeking help from a mental health hotline, and 32.8% would consider seeking help from a mental health hotline but were unaware of any mental health hotlines (Ministry of Health and Institute of Mental Health, 2020). In another survey conducted by Institute of Mental Health, it was noted that people would not seek professional help due to several reason like they could cope by themselves, seek help from friends or family first, too costly and or too busy (Goh, 2021).

In this context, we launched a lightweight chatting bot, TalkToMe that can promote and empower people to improve their emotional wellbeing via chatting service to users, providing resources and hotlines to seek out professional help when needed in a single click. It allows users to improve their mental health whenever and wherever they are, and without any cost considerations.

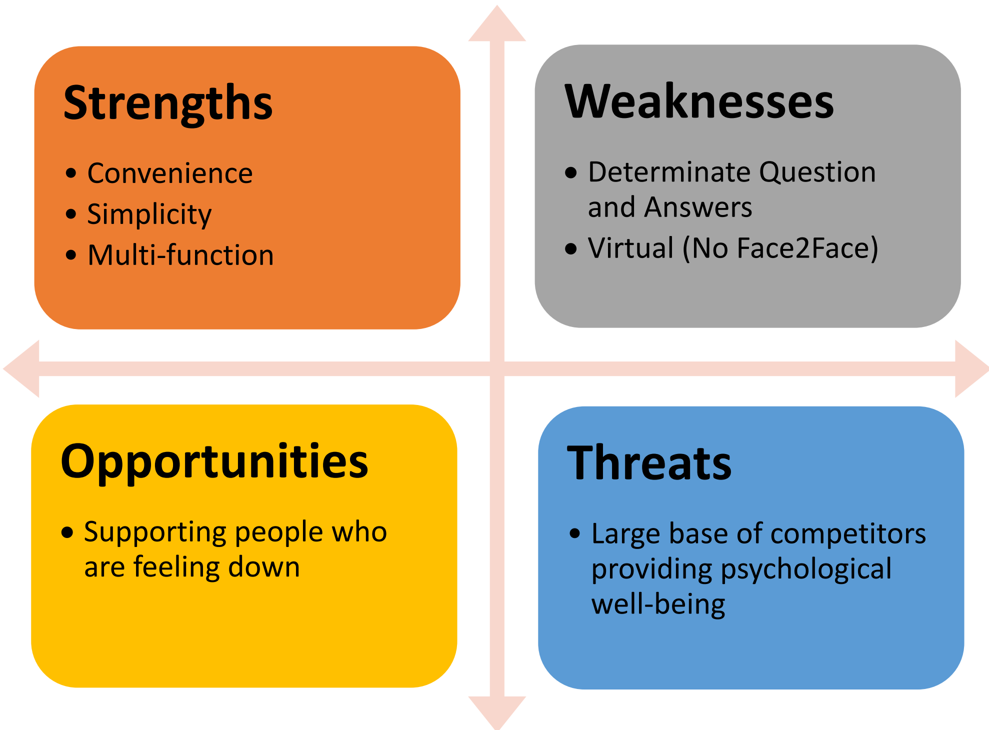


Figure 1 SWOT Analysis

Strength:

- **Convenience**. TalkToMe chatting bot built on top of Telegram Messaging Platform, makes it easily accessible to users 24/7
- **Simplicity.** Single click to retrieve the resources and hotlines
- **Privacy.** Do not collect any personal information from user. Just chat and comfort.

- **Multi-function.** The Talk2Me chatbot not only can chat with users and ease their pressure and burden, but also evaluate the risk of the users to hurt themselves.

Weakness:

- **Virtual.** Users are interacting with TalkToMe behind the keyboard, instead of face-to-face. This engagement may create a lack of trust and tough for follow-up.
- **Limited Generalization.** The internal problem classification model is built on a limited and specialized dataset. It might perform not well with trendy emotional words like emo. Timely updates on dataset are needed.

Opportunity:

- **Provides emotional support to people who are feeling down.** Even though TalkToMe cannot replace the position of professional practitioners, it is able to provide general emotional counselling.

Threat:

- **Large base of competitors.** There are several therapy chatbots on the market which aims to provide virtual counselling.

3 System Architecture and Design

The architecture of the system is shown below.

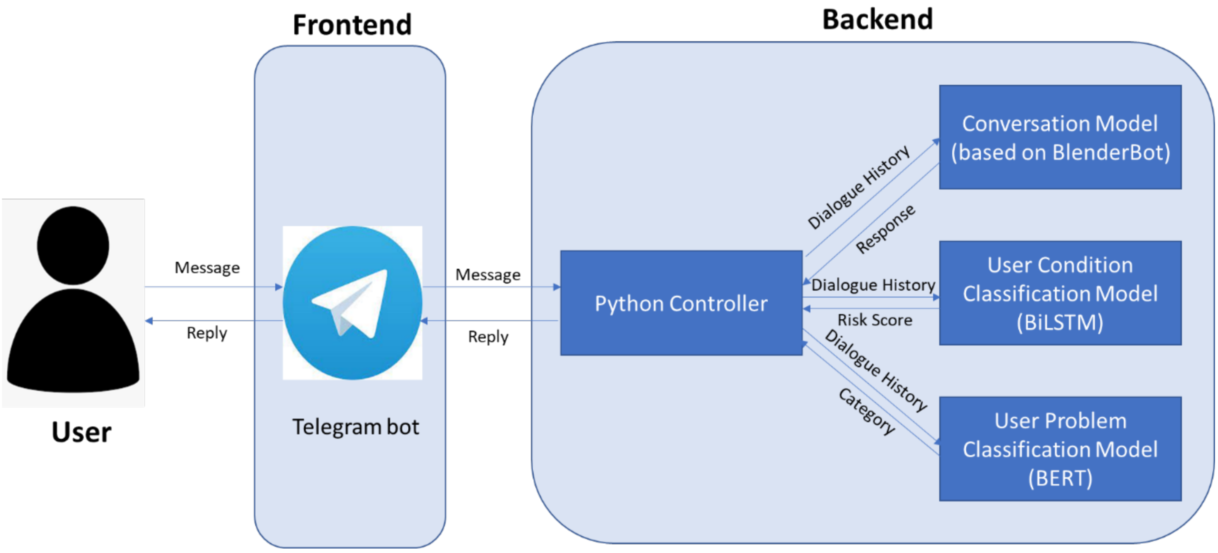


Figure 2 System Architecture Diagram

The following are the general functions of the components:

<u>Frontend</u>

Telegram Bot

- User-friendly UI for interaction.
- Send user messages to the Backend and return received responses to the user.

<u>Backend (Python)</u>

Dialogue Controller

- Handles input and response generation
- Handles end of conversation and saves user rating, risk and problem category

Conversation Model

- Generates a text response based on a user utterance and the session dialogue history

User Condition Classification Model

- Based on the session dialogue, generates a ‘risk score’ (0 to 1) which predicts the user’s probability of having self-harm tendencies
- User is then classified as high risk/low risk based on 0.5 threshold value

User Problem Classification Model

- Based on the session dialogue, predicts the category of the user’s problem out of a given set of common problems (listed in detail in Table 1)

4 Pre-Deployment

4.1 Data Gathering and Pre-processing

4.1.1 ESConv Dataset for Conversation Model and Problem Classification Model

The Emotional Support Conversation (ESConv) dataset (Liu, Siyang et al, 2021) consists of 1300 sample conversations between ‘help-seekers’ and 'supporters’. Each sample conversation has a metadata of the help-seeker’s problem category, and every response (within each conversation) from the supporter is tagged with a corresponding conversation support strategy. This conversation support strategy information is effective in providing emotional support and used to finetune the Conversation Model, while the problem category information is used to train the Problem Classification Model.


```
"dialog": [
  {
    "speaker": "seeker",
    "annotation": {},
    "content": "Hello\n"
  },
  {
    "speaker": "supporter",
    "annotation": {
      "strategy": "Question"
    },
    "content": "Hello, what would you like to talk about?"
  },
  {
    "speaker": "seeker",
    "annotation": {},
    "content": "I am having a lot of anxiety about quitting my current job. It is too stressful but pays well\n"
  },
  {
    "speaker": "supporter",
    "annotation": {
      "strategy": "Question"
    },
    "content": "What makes your job stressful for you?"
  },
  {
    "speaker": "seeker",
```


Figure 3 ESConv Dialog Sample

There are 8 support strategies as listed below, refer to Appendix – Strategies for details:

1. Question
2. Restatement or paraphrasing
3. Reflection of feelings
4. Self-disclosure
5. Affirmation and reassurance
6. Information
7. Providing suggestions
8. Others

To facilitate users' problem classification, we utilized the Emotional Support Conversation Dataset, and make slight modification on the grouping of categorization. Following are the distribution of users' problem category after modification for training. In total there are 7 problem category with total of 1300 samples.

Table 1 Users' Problem Category (after modification)

No	Original Problem Category	Problem Category	Count
1.	Ongoing Depression, Appearance Anxiety	Emotional	363
2.	Job Crisis	Work	280
3.	Breakup with Partner	Partner Relationship	239
4.	Problem with Friends	Friendship	179
5.	School Bullying, Academic Pressure	School	158
6.	Conflict with Parents, Issues with Children, Issues with Parents	Family	28
7.	Sleep problems, Procrastination, Alcohol Abuse	Others	53
		Total	1300

4.1.2 Kaggle Suicide Dataset for User Condition Classification Model

To train the user condition classifier, we used a suicide detection dataset available on Kaggle (by Komati et al.). The dataset consists of posts from the Reddit platform under the "SuicideWatch" and "depression" topics and the posts are labelled as "suicide" or "non-suicide". Since the original dataset was too large, we took a subset of this data by limiting the length of the text to 150 characters in our dataset. Our final condition classification dataset consisted of 64368 samples.

4.2 Training and Evaluation

4.2.1 Conversation Model

The Conversation Model is based on a pretrained conversation model, BlenderBot (Roller et al, 2020) and finetuned with the Emotional Support Conversation (ESConv) dataset and conversation strategy method proposed in ESConv Framework (Liu et al., 2021). The ESConv Framework proposed that emotional support generally occurs in 3 stages: (i) exploration, (ii) comforting, (iii) action, and different conversation strategies should be used for each stage (refer to 5.2 below). The BlenderBot model is finetuned on the ESConv dataset for 2 epochs using the Adam optimizer, batch size of 16 and a learning rate of 3e-5 (refer to args.json[1] file for the full set of parameters). File readme_run_train.txt[2] contains the step for training the conversation model.

[1] Training Parameters

[2] Step to Train Model

```
cls_strat_id: confusion_matrix
 [[566   0   0   0   0   0   0   0]
 [  0 152   0   0   0   0   0   0]
 [  0   0 235   0   0   0   0   0]
 [  0   0   0 264   0   0   0   0]
 [  0   0   0   0 434   0   0   0]
 [  0   0   0   0   0 507   0   0]
 [  0   0   0   0   0   0 186   0]
 [  0   0   0   0   0   0   0 457]]
```


```
{
  "Bleu_1": 0.19939608413892207,
  "Bleu_2": 0.07987918727008815,
  "Bleu_3": 0.040015948987298704,
  "Bleu_4": 0.023355670011495085,
  "CIDEr": 0.18094933043649508,
  "EmbeddingAverageCosineSimilarity": 0.898285,
  "GreedyMatchingScore": 0.710656,
  "METEOR": 0.07816004005371568,
  "ROUGE_L": 0.18022301137317454,
  "VectorExtremaCosineSimilarity": 0.513074
}
```


Figure 4 Confusion Matrix for Strategy Classification (Left) and NLGEval() Metrics Result (Right)

4.2.2 User Condition Classifier

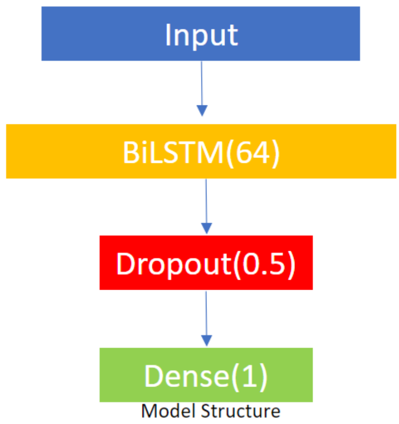


Figure 5 Bi-LSTM Model for User Condition Classification

The user condition classifier performs a binary classification for a user based on the session dialogue (predict high risk or low risk of suicide or depression). It consists of an embedding layer, a bidirectional LSTM layer and a dense output layer. The embedding layer turns the input (indexed word tokens) into dense vectors of fixed size. The Bidirectional LSTM layer has 64 nodes and a recurrent dropout of 0.5. The dense output layer has only 1 node and uses the sigmoid activation function (its output corresponds to the probability of being high-risk). It was trained using the Adam optimizer and binary cross-entropy loss function. It was trained for 25 epochs and the accuracy and loss curves are shown below. The final model was trained for 10 epochs to avoid overfitting and underfitting, and the final test accuracy was at 90.1%.

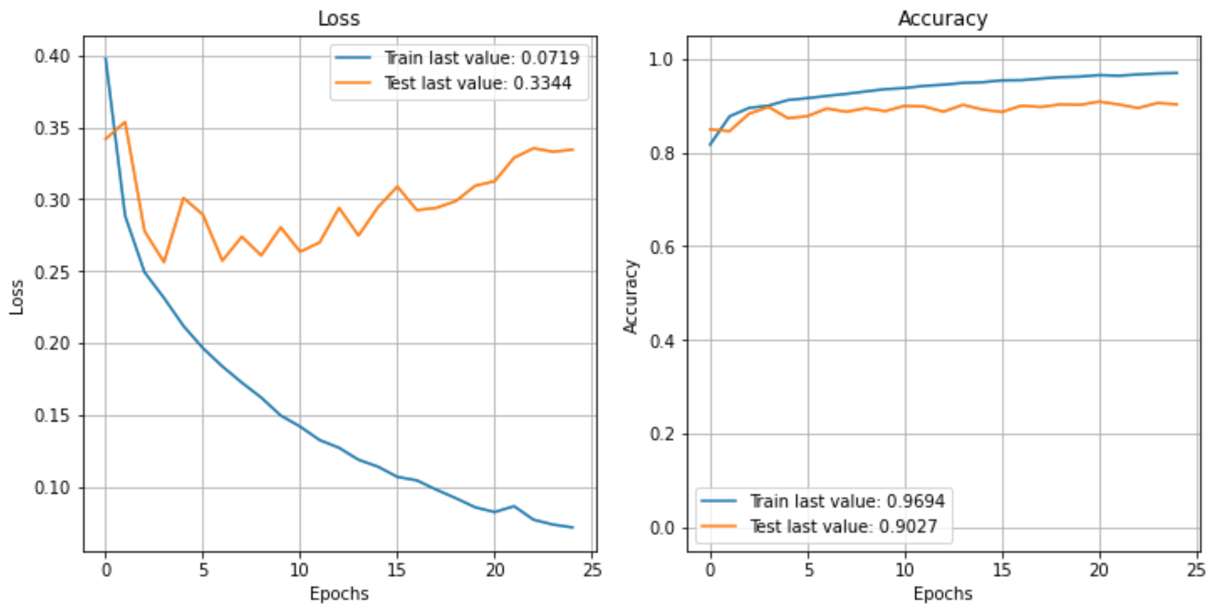
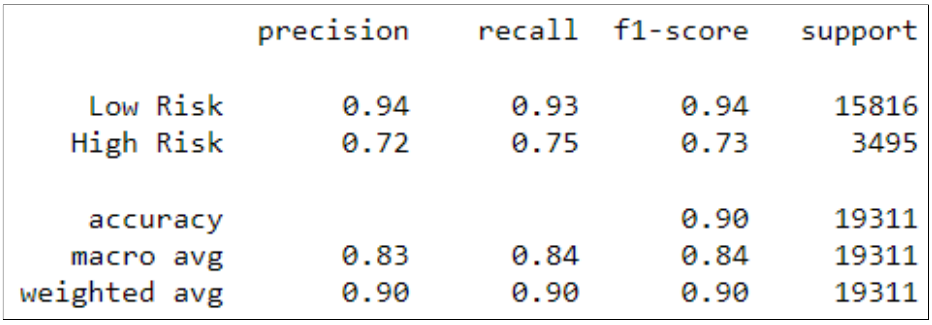


Figure 6 User Condition Classifier Performance (Loss and Accuracy)



	precision	recall	f1-score	support
Low Risk	0.94	0.93	0.94	15816
High Risk	0.72	0.75	0.73	3495
accuracy			0.90	19311
macro avg	0.83	0.84	0.84	19311
weighted avg	0.90	0.90	0.90	19311

Figure 7 User Condition Classifier (Classification Report)

4.2.3 User Problem Classification

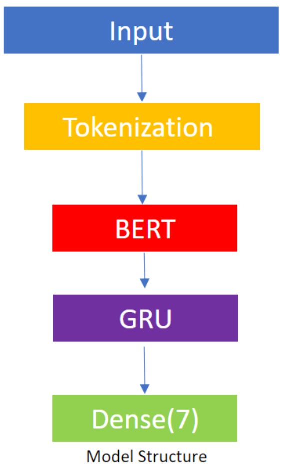


Figure 8 BERT Model for Users' Problem Classification

The user problem classifier performs a multi-classification prediction. It predicts the source of emotion or problem including emotional, work, relationship, friendship, school, family, or others based on the user session dialogue. It consists of tokenization and encoding layer and BERT layer. The tokenization and encoding layer turns the natural input into tokens first and then ids to BERT. The BERT layer loads the pretrained model checkpoint provided by Transformers and do finetuning on final dense output. The BERT layer has 256 hidden dimensions, 2 hidden layers, a dropout of 0.25, cross entropy loss and Adam optimizer. The dense output layer has 7 dimensions output corresponds to the probability of 7 sources. It was trained for 20 epochs, to avoid overfitting and underfitting, as supported by our experiment results. The train & validation accuracy and loss curves during training are shown below. Here we choose the checkpoint with best validation accuracy (79.67%) and the final test accuracy was at 91.69%.

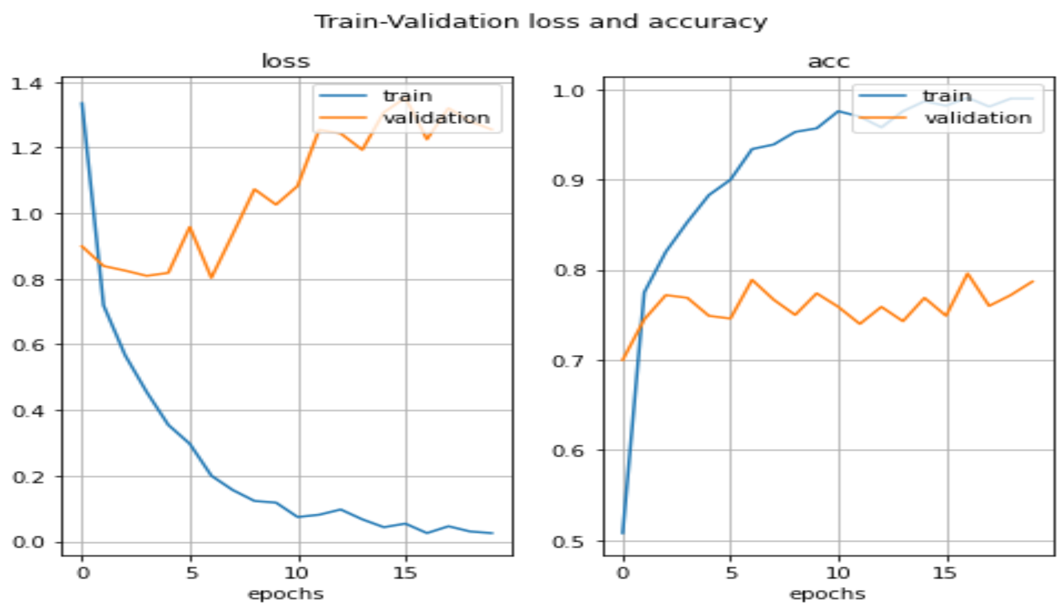


Figure 9 User Problem Classifier BERT Performance (Loss and Accuracy)

Table 2 shows the other model that we have trained to predict users' problem category and its test accuracy, including Naïve Bayes, Decision Tree, SVM, Random Forest, CNN and BiLSTM. For classic machine learning, we use CountVectorizer and TfIdfTransformer to build embeddings. And in LSTM and CNN, we build word2id from ESConv dataset and use GLOVE embedding matrix.

Table 2 Users' Problem Classification Model Test Accuracy

No	Model	Test Accuracy
1.	BERT with GRU	89.41%
2.	Naïve Bayes	66%
3.	Decision Tree	68%

4.	SVC	74%
5.	Shallow CNN	59%
6.	Shallow CNN with Dropout	65%
7.	Deep CNN	65%
8.	BiLSTM – Recurrent Dropout	71%
9.	Ridge Classifier	76.4%
10.	Random Forest	72.8%
11.	Voting Classifier (RF, RC, SVM)	76.6%

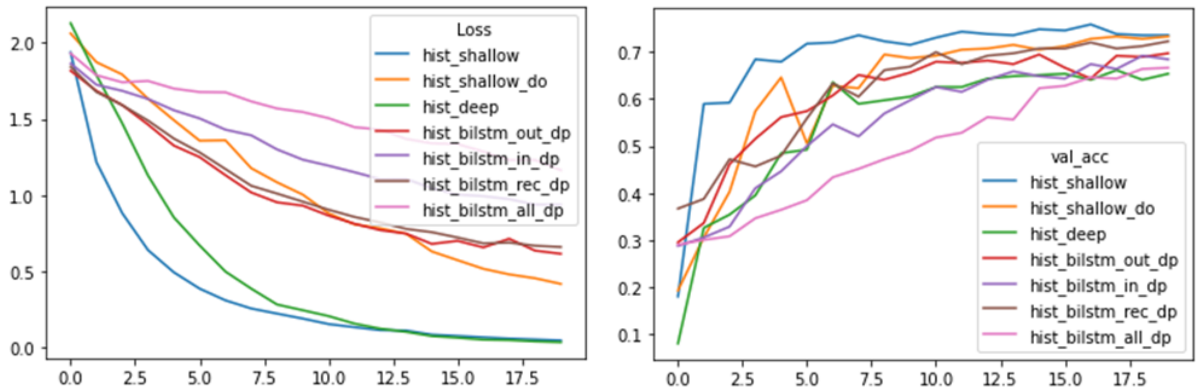


Figure 10 Model Comparison (Shallow CNN, Deep CNN, Bi-LSTM) for Users' Problem Classification

5 System Implementation

5.1 TalkToMe Flow Chart

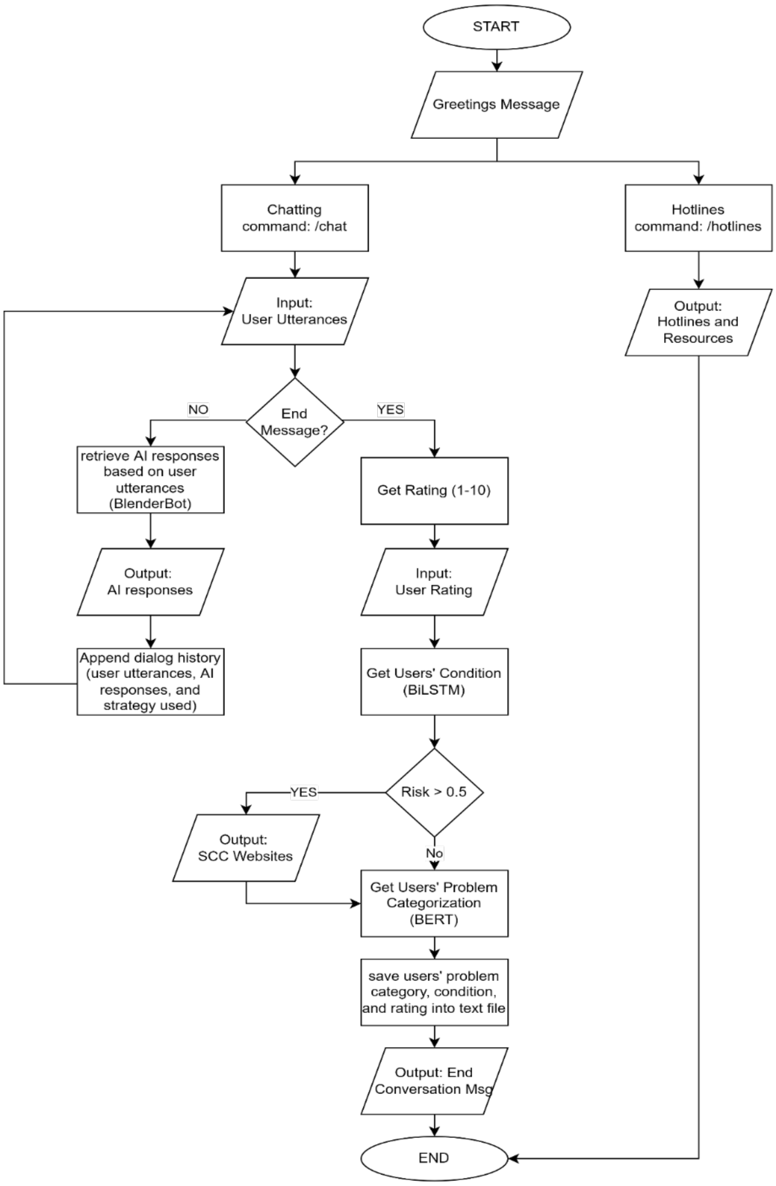


Figure 11 TalkToMe Flow Chart

Our chatbot is deployed on Telegram, a web-based messaging service widely used in Singapore. Users can search for the handle '@Talk2me_plp_bot' to talk to the chatbot. Telegram serves as a user-friendly UI, allowing user utterances and system generated responses to be exchanged via the Telegram chat window.

Figure 11 shows the TalkToMe conversation flow. It starts with a greeting message and inform the two services offered by the bot, including chatting services and hotline/resources information retrieval services. In chatting services, if the bot detected ending message like "bye", "quit" or "end", then it will end the conversation and request users to provide an overall rating about the bot. Subsequently, it will perform users' problem and users' condition classification (refer to section 5.2). All the information regarding username, user's problem, user's condition (risk score) and rating will be saved into text file in the system for further analysis and insights in improving the bot. If the user utterance is not ending message, then the utterances will be forward to conversation model for generating responses (refer to section 5.2).

5.2 Conversation Model, Users' Problem and Users' Condition Classification Model

When the user sends a message to the Telegram bot via the chat window, it gets forwarded to the python controller in the system backend and a response is generated accordingly. The generated response is then sent back to the same Telegram chat.

There are 3 models used in the chatbot backend. First, the conversation model (based on the Emotional Support Conversation Framework from Liu et al.) is the model responsible for generating the conversation responses given the user utterances. It uses a BlenderBot model (seq2seq) trained on conversations between 'help-seekers' and 'supporters. The model selects different strategies to generate responses and the strategies are selected based on the current stage of the conversation (i.e., number of turns).

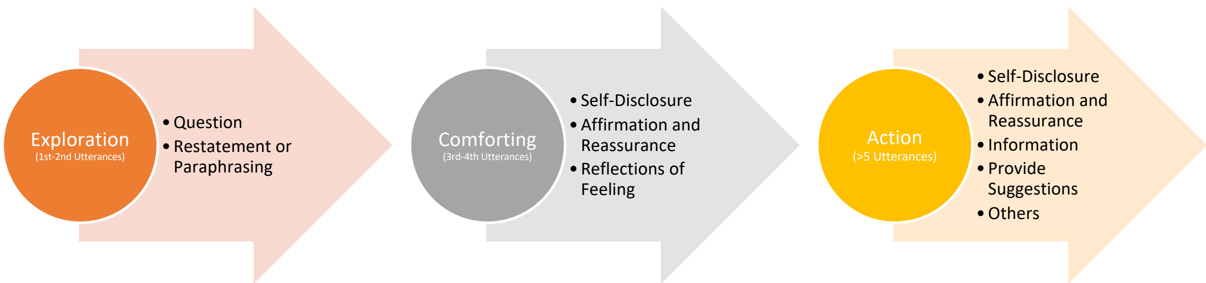


Figure 12 Procedure of Emotional Support in order: Exploration -> Comforting -> Action

For example, in the initial stage of conversation (1^{st} and 2^{nd} User Utterances), the model will use Exploration procedures. In this stage, two support strategies are choose randomly in generating the responses to users, ie "Question" or "Restatement or Paraphrasing". Then the model will move to the next stage, Comforting procedures, and subsequently to the Action procedures. At each of the stage, there are support strategies to be chosen randomly in generating responses to users.

The detailed strategy selection policy is shown in the table below.

Table 3 Three Stages and Supported Strategies Selection based on the User Utterances Turn

<table>
<tr><th>1st-2nd User Utterance
(Stage: Exploration)</th><th>3rd-4th User Utterance
(Stage: Comforting)</th><th>>5th User Utterance
(Stage: Action)</th></tr>
<tr><td>Question</td><td colspan="2">Self-disclosure</td></tr>
<tr><td>Restatement or Paraphrasing</td><td colspan="2">Affirmation and Reassurance</td></tr>
<tr><td>-</td><td>Reflection of feelings</td><td>Information</td></tr>
<tr><td>-</td><td>-</td><td>Providing Suggestions</td></tr>
<tr><td>-</td><td>-</td><td>Others</td></tr>
</table>

The second model is the User Condition Classification Model (based on BiLSTM). This model is used to infer the condition (i.e., high risk/low risk of self-harm) of the user at the end of each conversation. It takes in the session dialogue as input and outputs a value between 0-1 which represents the risk score of the user. We assume a threshold of 0.5 and classify values >0.5 as the user being at high risk of self-harm. When risk score is >0.5, the chatbot will automatically send some helpline resources and encourage the user to seek professional help if needed.

The third model is the User Problem Classification Model (based on BERT). This model is used to infer the user's problem category (e.g., school, work, family, etc.) at the end of each conversation. It takes in the session dialogue as input and outputs the predicted category of the user's problem. The category can be then used for further analysis and action downstream. For example, we can find out which are the areas people are having the most troubles in and try to increase the support in those areas.

6 Conclusion and Future Work

In this project, our team has implemented Talk2me, an Emotional Detection and Psychological Support Chatbot. The chatbot's main function is to provide emotional support to users who might be experiencing negative emotions, and this is achieved through strategic conversation. Besides being able to chat with users, it has two supporting and hidden functions: (1) detecting risk of suicide and (2) predicting the user's problem category. These additional functions help the chatbot to work more effectively as further action and targeted support can be implemented based on the user's condition. For example, we have implemented automatic sending of helpline resources to users who have been detected to have a high risk of suicide. Other methods of targeted support can also be implemented in future.

Currently, Talk2me is still in the minimum viable product stage and there are some potential improvements that we can make to increase the effectiveness of the chatbot. Firstly, we could implement on-the-fly emotion type and intensity detection to provide targeted conversation support and hotlines according to the user's problem and condition. Secondly, we could also do a text summarization and analysis of the users' problems and perform aspect extraction on consolidated users' open-ended text feedback to improve the chatbot. Thirdly, we could expand the training dataset with chat records between users and the chatbot (after doing manual annotation and checking quality of the dialogues). Lastly, we also consider deploying Talk2me on other popular social media (besides Telegram) in the future. We hope that our efforts can help more people and provide them with effective emotional support.

References

Goh, T. (2021). *IMH study points to likely increase in mental health issues in S'pore amid Covid-19*. Retrieved from https://www.straitstimes.com/singapore/health/imh-study-points-to-likely-increase-in-mental-health-issues-in-spore-amid-covid-19

Liu, Siyang et al. (2021). Towards Emotional Support Dialog Systems. *Proceedings of the 59th annual meeting of the Association for Computational Linguistics.*

Ministry of Health and Institute of Mental Health. (2020). *COVID-19 Mental Wellness Taskforce Report*. Retrieved from www.moh.gov.sg/docs/librariesprovider5/covid-19-report/comwt-report.pdf

Roller et al. (2020). Recipes for building an open-domain chatbot. *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics* (pp. 300–325). Association for Computational Linguistics.

World Health Organization. (2022). *COVID-19 pandemic triggers 25% increase in prevalence of anxiety and depression worldwide*. Retrieved from https://www.who.int/news/item/02-03-2022-covid-19-pandemic-triggers-25-increase-in-prevalence-of-anxiety-and-depression-worldwide

Appendix A: Support Strategies Description

No	Strategies	Description
1	Question	Asking for information related to the problem to help the help-seeker articulate the issues that they face. Open-ended questions are best, and closed questions can be used to get specific information.
2	Restatement or Paraphrasing	A simple, more concise rephrasing of the help-seeker's statements that could help them see their situation more clearly.
3	Reflection of Feelings	Articulate and describe the help-seeker's feelings.
4	Self-disclosure	Divulge similar experiences that you have had or emotions that you share with the help-seeker to express your empathy.
5	Affirmation and Reassurance	Affirm the help seeker's strengths, motivation, and capabilities and provide reassurance and encouragement.
6	Providing Suggestions	Provide suggestions about how to change but not overstep and tell them what to do.
7	Information	Provide useful information to the help-seeker, for example with data, facts, opinions, resources, or by answering questions.
8	Others	Exchange pleasantries and use other support strategies that do not fall into the above categories.

Appendix B: Project Proposal and User Guide Document

Group3_ProjectProposal.pdf

Group3_TalkToMe_UserGuide.pdf

Github Repository Link: PLP-GRP03-FT-2021-2022-Talk2Me