Your Companion : Chatbot

Group 14

Abstract—A chatbot that helps people suffering from depression. It also provides movie and song suggestions to cheer up your mood.

Index Terms—Depression, Artificial Intelligence, Natural Language Processing, Knowledge Graph.

1 Introduction

In this project we have build a chatbot that helps people with depression. The major problem of people with depression is that they don't have anyone to share their feelings. They feel alone, because of this some people take drastic steps. We want to tackle this problem by using a chatbot. Nowadays chatbots are every where. They act as your personal assitant like siri, cortana etc. They can also control your home appliances with devices like Google Home. So we want to make a chatbot that can help people suffering from depression. Our chatbot, Your Companion, talks to you about your problems, listens to you and motivates you. All conversations with our chatbot are private, anonymous so that you can freely chat with our bot without any concern.

Also, movies and songs can be of great help in depression. So, we suggest motivational, comedy, or feel-good movies and songs that can help people cheer up their mood.

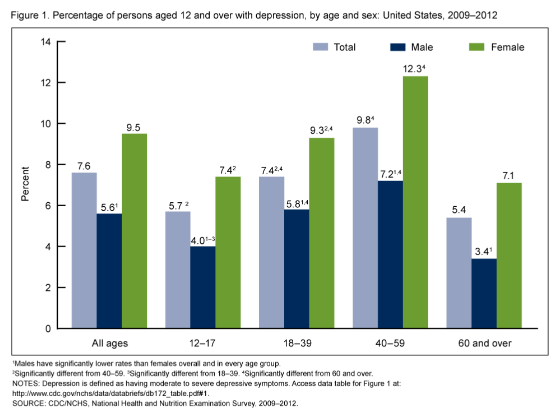

Fig. 1. Percentage of people depressed in different age groups.

2 Motivation

According to WHO [6], depression is one of the most common mental disorder. Globally, more than 300,000,000 people from all the age groups suffer from depression. It is also the leading cause of disability, and a major contributor to the overall global burden of disease. Around 15% of people suffering from depression commit suicide. Figure contains the statistics of percentage of people depressed in different age groups.

One of the major reasons of depression is absence of communication. Communication tend to help people suffering from depression.

Due to recent developments in Artificial Intelligence, Natural Language Processing, and Machine Learning, we feel computers can definitely help people who are suffering from depression. Hence, we provide a neat interface where people can go and anonymously chat with our bot. We ensure privacy so that people concerned about their conversations remaining private can chat freely.

3 Database

For training a chat bot we need conversational database. For our specific purpose we needed conversations where Person 1 is sad or depressed and Person 2 is trying to help him. Obtaining dataset for training our chat bot was very hard as there was no existing dataset for helping depressed people. These type of conversations generally happen on helpline or with doctors and both of these sources don't want to reveal the information because of the confidentiality.

So we conducted a survey on IIT Guwahati Current Campus Junta to obtain the conversations

• *Mohit(130101048), Shivam(130101071), Rohan(130101066), Prabodh(130101087)*

that will help depressed people. Aside from the responses we got from CCJ we also manually created healthy conversations. This acted as our training dataset.

These conversations were then fed to the training process of the chat bot which were used to build the Knowledge Graph.

Following are some examples of the training data:

Person : "I want to improve my life."

Chatbot: "For me meditation and breathing helps...but real change comes by changing our thoughts"

Person : " "Why are people so selfish?"

Chatbot : "Nobody is selfless in the world. Being selfless means doing good to others which makes 'you' feel good, and hence a selfish deed. It's just that reasons of being satisfied are different for different people."

4 SUB-MODULES

4.1 Sentiment Analysis

Sentiment analysis or opinion mining is the use of Natural Language Processing to obtain the overall Attitude (positive or negative) of a sentence. We used [3] to implement Opinion Mining. The sentiment of a sentence helps to find a appropriate reply. If the sentiment is very negative and if there are tags such as "kill", "die", "suicide" etc, we suggest the person to talk to a help-line because we feel that they may take some drastic actions.

Following are some of the examples of sentiment analysis:
Sentence : "I am sad!" Sentiment: -0.5255.
Sentence : "I am very happy!" Sentiment : 0.6468
Sentence : "I am ecstatic" Sentiment : 0.5562

4.2 Tagging

Tagging or Part-of-Speech(POS) Tagging is the process of assigning each token a Part-of-Speech tag. For English, there are 9 parts of speech: conjunction, adverb, pronoun, preposition, adjective, article, verb, noun. Certainly, there are more subcategories(or fine-grained categories) of each POS such as, a noun can be categorised based on gender in some languages or it can also be categorised on the basis of plurality. Most of the POS taggers do have fine-grained categorization.

The challenge with POS tagging is, most of the time single word belong to different parts of speech depending on context and neighboring words. Following statements illustrate the problem:
Sentence 1: "They wandered *about* the town."
Sentence 2: "They wanted to talk *about* him."
The word *about* in sentence 1 is used as an *adverb* while in sentence 2 it is used as *preposition*.

We used [7] to implement POS tagging. POS tagging was mainly used to pre-process the raw data which would be used by different algorithms.

4.3 Lemmatization

Sometimes some words may mean same (to some extent). e.g drink, drinking, drank. These words can be derived from the word *drink*. This makes the task of semantic evaluation a bit easier as these have almost same meaning if they are present in similar context. The process of converting these words into a single form (from which it's meaning can be derived easily) is called as *lemmatization*. We used WordNet [8] lemmatizer for lemmatization. Lemmatization was mainly used at preprocessing stage of raw data to simplify the data.

4.4 Profanity Detection

With the increase in content of conversations, the explicit textual content in them has also grown. If we try to pick up a dataset with a large number of conversations in it, we need to ensure that our chatbot would not in-turn learn to swear or reply with any objectionable remark/content. Profanity detection and its removal has become one of the major issues while creating a chatbot.

Some of the major challenges faced in doing so will include:

- The system has to detect profanity based on abbreviations, misspellings (both intentional and unintentional).
- Moreover some of the terms which might seem as offensive in one community may be acceptable in another.
- New slangs are developed and get popular/viral very quickly in this era of Internet, so the detection system has to keep itself updated otherwise it would become stale.

The profanity based detection systems are mostly list based i.e. they are based on a list containing explicit words. Profanity detection is done by using regex and edit distance from the common explicit word list.

4.5 Semantic Networks

Usage of graphs (well directed graphs) to represent the semantic memory (knowledge) is kind of new paradigm. Semantic networks or more commonly knows as knowledge graphs helps to represent the semantic learning of ontological aspects of human understanding. It tries to mimic the way cognitive information is store in human brain. [2].

Today, some of the most famous products are based use semantic networks to answer queries or generate responses, one of them includes **Google**. 2 represents a sample knowledge graph and

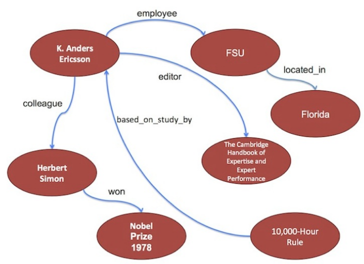

Fig. 2. A sample knowledge graph. [4]

Fig. 3. A sample query result by google using knowledge graph. [5]

We have used semantic networks/knowledge graphs to generate response to query posted by the user.

5 Workflow

The end to end workflow i.e. from the user input to the response generated can be broken down into smaller steps which are explained below:

1) We first try to find the sentiment of the user input. (The details are explained in 4.1)
2) On the basis of sentiment we try to find out if the case of the user is very critical. If it is, we suggest a helpline number for such very critical cases otherwise we try to generate our own response.
3) For generating the response we first lemmatize (explained in 4.3) the input.
4) After lemmatization, we tag the different components of lemmatized sentence.
5) And finally, by using the tagged results and lemmatized sentences the semantic network is used to generate response/output.
6) The generated response passed through a profanity checker. If it passes that test, we output the response to the user.

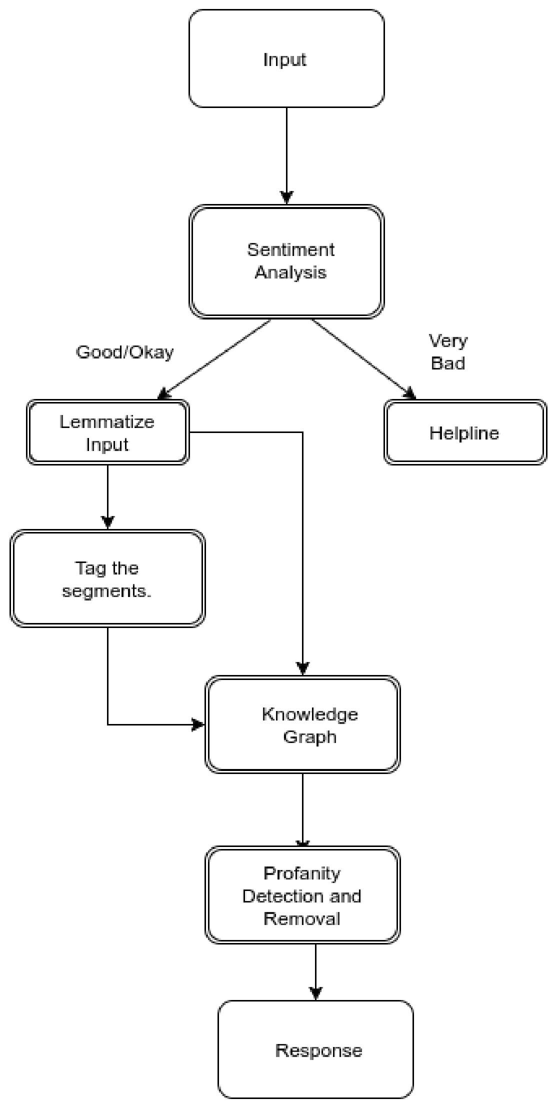

Fig. 4. Showing the end to end work flow of our bot.

6 Experiments

The following figures 5, 6, 7 show demo conversation between us and the chatbot (**Your Companion**).

In conversation 6, when the user says "Will I die alone?" the bot gives an answer which may give some hope to the user. This conversation was fed to the bot in our training dataset. Clearly, it has stored it in its Knowledge Graph.

As shown in 7 when the user talked about dying (which is an extremely negative sentiment) the bot directly refers the user to an external helpline whereas for less critical situations the chatbot tries to handle the situation on its own.

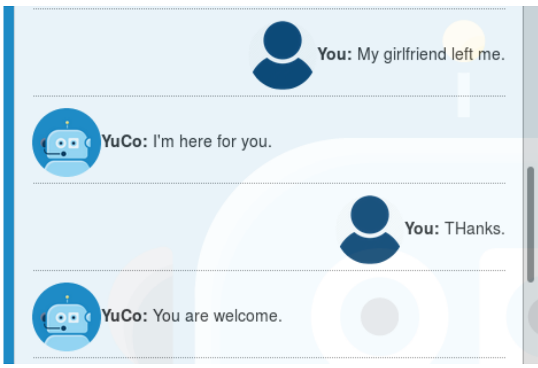

Fig. 5. Conversation:1

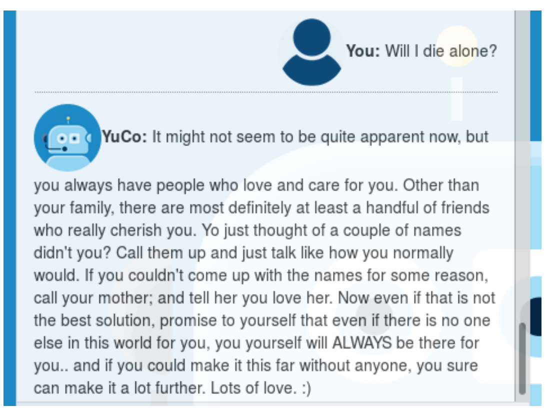

Fig. 6. Conversation: 2

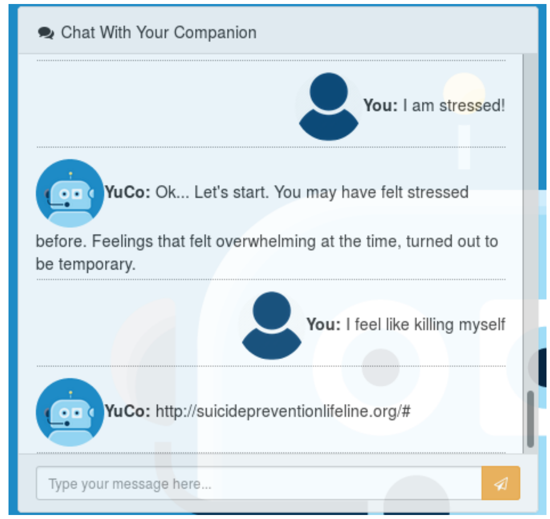

Fig. 7. Conversation: 3

7 Conclusion & Future Directions

We are getting decent results as can be seen in the experiments section given the small dataset. Since there is no formal way to test the chat bot, we plan on getting feedback from users to see if this bot works fine. Knowledge-Graph generated is very sparse which can be solved by feeding a larger dataset.

We will further work on including gifs and emoticons as they tend to make the conversation more informal and lively. We also plan on working on remembering the context of the conversation so that the conversation doesn't feel like just question-answers.

Acknowledgments

We would like to thank Prof. R. D. Baruah and all teaching assistants because of whom we got an opportunity to work on a project like this. We would also like to thank all those who participated in the survey which we took to create the database.

References

[1] H. Kopka and P. W. Daly, *A Guide to LaTeX*, 3rd ed. Harlow, England: Addison-Wesley, 1999.
[2] Loftus, Elizabeth F., and Geoffrey R. Loftus. "On the permanence of stored information in the human brain." American Psychologist 35.5 (1980): 409.
[3] Socher, Richard, et al. "Recursive deep models for semantic compositionality over a sentiment treebank." Proceedings of the conference on empirical methods in natural language processing (EMNLP). Vol. 1631. 2013.
[4] Sample Knowledge graph, http://www.keywordperformance.com/wp-content/uploads/2014/09/knowledge-graph.jpg, accessed: 19 April 2017
[5] Sample google response using knowledge graph, http://www.keywordperformance.com/wp-content/uploads/2014/09/knowledge-graph.jpg, accessed: 19 April 2017
[6] WHO on depression http://www.who.int/mediacentre/factsheets/fs369/en/, accessed: 19 April, 2017
[7] Toutanova, Kristina, et al. "Feature-rich part-of-speech tagging with a cyclic dependency network." Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1. Association for Computational Linguistics, 2003.

[8] Miller, George A. ”WordNet: a lexical database for English.” Communications of the ACM 38.11 (1995): 39-41.