TALLINN UNIVERSITY OF TECHNOLOGY
School of Information Technologies
Department of Software Science

Rufat Valiyev 182461IVSM

Deep Learning for Sentiment Information Measurement based on Social Media Posts (Chatbot for Suicide Detection)

Master's thesis

Supervisor: Sadok Ben Yahia
PhD

Tallinn 2020

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond
Tarkvarateaduse instituut

Rufat Valiyev 182461IVSM

Sügav õppimine sentimentaalse teabe mõõtmiseks sotsiaalmeedia postituste põhjal (Chatbot enesetappude tuvastamiseks)

Magistritöö

Juhendaja: Sadok Ben Yahia
PhD

Tallinn 2020

Author’s declaration of originality

I hereby certify that I am the sole author of this thesis and this thesis has not been presented for examination or submitted for defence anywhere else. All used materials, references to the literature and work of others have been cited.

Author: Rufat Valiyev
14.05.2020

Abstract

Sentiment analysis (*aka* opinion mining) is the study field used for analyzing the opinions of people based on their written statements. It is used to identify the aspect of written texts (main subject of text) and to categorize them. This type of analysis is related to natural language processing.

In last years, popularity and importance of sentiment analysis increased, and it creates some new application areas of sentiment analysis. One of the most important application areas of the sentiment information measurement is to build an effective tool in order to detect harmful contents of people and prevent them. In this thesis, our main objective was developing chatbot for detecting suicide-related messages of the user and answering him or her with appropriate preventive messages by using sentiment analysis techniques.

The project contains two main parts. The first part is detecting if the user's messages are suicide-related or not. The second part is classifying these sentiment-related messages into categories and give appropriate preventive messages to the user.

This thesis paper is written in English is 30 pages long, including 7 chapters, 11 figures.

Annotatsioon

Sentimentide analüüs (ehk arvamuste kaevandamine) on uuringuväli, mida kasutatakse inimeste arvamuste analüüsimiseks nende kirjalike avalduste põhjal. Seda kasutatakse kirjutatud tekstide (teksti põhiaine) aspekti tuvastamiseks ja nende kategoriseerimiseks. Seda tüüpi analüüs on seotud loomuliku keele töötlemisega.

Viimastel aastatel on tundeanalüüsi populaarsus ja tähtsus kasvanud ning see loob tundeanalüüsi mõned uued rakendusalad. Üks olulisemaid tundeteabe mõõtmise rakendusalasid on tõhusate tööriistade loomine inimeste kahjuliku sisu tuvastamiseks ja ennetamiseks. Selles lõputöös oli meie põhieesmärk arendada vestlusprogrammi kasutaja enesetappudega seotud teadete tuvastamiseks ja talle vastavate ennetavate teadetega vastamiseks sentimentide analüüsimeetodite abil.

Projekt koosneb kahest põhiosast. Esimene osa tuvastab, kas kasutaja sõnumid on seotud enesetappudega või mitte. Teises osas liigitatakse need sentimentidega seotud sõnumid kategooriatesse ja antakse kasutajale asjakohaseid ennetavaid sõnumeid.

See ingliskeelne lõputöö on 30 lehekülje pikkune, sisaldades 7 peatükki ja 11 joonist.

List of abbreviations

WHO - world health organization

NLP - natural language processing

LSTM - long short-term memory

GloVe - global vectors

seq2seq - sequence to sequence

UI - user interface

CNN - convolutional neural network

POS - part-of-speech

cBLSTM - contextual bidirectional long short-term memory

LM - language model

NN - neural networks

RNN - recurrent neural networks

BRNN - bidirectional recurrent neural networks

APA - American Psychiatric Association

AAS - American Association of Suicidology

API - application programming interface

RT - retweet

MVP - minimum viable product

Table of Contents

List of figures

1 INTRODUCTION

Sentiment information measurement is one of the most important branches of natural language processing. Main work in sentiment analysis is to detect the subjects of the texts and classify them according to defined categories. In last years, thanks to many academic researches and studies in this field, the importance of sentiment analysis increased. As a result of this improvement, many application areas of sentiment information measurement appeared, mostly business and social areas. Sentiment analysis has a crucial role for businesses to get information about their customers' opinions on their products. By aspect extraction techniques of sentiment analysis, they can clarify which features of product and/or services satisfy their customers, and which features do not match with customers' needs. For these situations, customer reviews in both companies' own site and different sale websites, e.g., amazon, ebay, etc., can be used. Besides that, sentiment analysis helps us to classify the texts into "good", "bad", "happy" and some other classes. Such classification can give business companies accurate statistical data that they can analyze it to improve their services and/or products. Moreover, analyzing people's written texts can be used to make predictions about future events. The best example can be making prediction about results of elections based on people's social media posts (Facebook, Instagram, twitter, etc.). In addition to these, we can also predict the people's possible future actions in their social life based on their posts and messages in their social media accounts.

Suicide is one of the most important problems of the modern society and it needs to be solved. According to WHO, around 800,000 people dies because of the suicide each year. This type of deaths is popular among the young generations (15-29 years old) [1]. There can be many reasons for people to make a suicide, for example anxiety, depression, concerns about future, unemployment, etc. Most of time, people with suicide ideations are not willing to communicate with people and socialize, therefore, it was so difficult to detect them in the past. However, nowadays huge number of people with suicidal ideations expressed their ideas and plans about suicide commitment on their social media profiles (as a post or message to their friends). Therefore, by using social media datasets,

we can better understand their thoughts and behaviors. Consequently, we can make preventive plans to save them.

There are many application areas of NLP and it solved a lot of social problems till now. Considering this and huge dataset that we can get from social media platforms, it is clear that deep learning and automated NLP techniques are very useful to detect people with suicide ideations and behaviors beforehand.

1.1 Research Goal

The main research goal of this thesis to develop a chatbot that will get messages from the user, analyze them with NLP methods based on trained social media posts and detect if there is any suicidal content, then classify them and give appropriate answers to the user. The main focus of this research is to use the best NLP technique to suicidal content detection and preparing most suitable answers for this content.

1.2 Unit of Study

The main unit of study is to detect suicidal contents in messages which are sent from the user in the application. Currently, there are some techniques which can categorize the textual contents as suicidal or not. In this research, the author plan to analyze these techniques and use the best techniques to detect suicide messages in the project. The project will be in conversation format (by other words, in chatbot format).

In the research, the plan is to use two models. The first one is created using Keras to classify sentences as suicidal or not. Bidirectional Long Short-Term Memory (LSTM) with embedding layer is used to train the model. Pretrained GloVe embeddings used for the embedded layer. For the second model that learns to reply based on user input is on encoder-decoder based seq2seq model.

1.3 Research Questions

i. What are the current techniques in sentiment analysis?
ii. How can the selected technique (Bidirectional LSTM) be applied in suicide detection based on social media posts?
iii. How this model can be implemented in chatbot?

1.4 Organization of the Thesis

This thesis is divided into following chapters:

- Introduction
- Literature Review
- Background
- Project Overview
- Project Development
- Future Work
- Summary

2 LITERATURE REVIEW

Thanks to the growing importance of the sentiment information measurement, many researches were operated in order to increase the accuracy scores of the existing methods, provide better solutions that can overperform the existing methods and/or implement these methods in different application areas that eases the work of the business.

In one of the most recent researches, which was about extracting aspects with deep convolutional neural networks and conducted by Soujanya Poria et al., aspect extraction method of sentiment analysis was investigated. During the research, they used 7-layer deep convolutional neural network to get main aspects of opinions. Besides that, they used some linguistic patterns with neural network to minimize the errors. They thought that features of aspect term depend on its surroundings. Therefore, they created a window of 5 words around each word (5 words before and after each word) and considered this window as features of each word. Then, they applied convolutional neural network (CNN) on this feature vector. Their CNN contained the following layers:

- Input layer
- 2 convolutional layers (the first had 100 feature maps and filter size 2, the second had 50 feature maps and filter size 3)
- 2 max pool layers (each layer's pool size was 2)
- Fully connected layer with softmax output

In addition to word embeddings, they also used 6 basic parts of speech tags (POS tags): noun, verb, adjective, adverb, proposition, conjunction. Moreover, they defined 5 linguistic rules to mark words as an aspect correctly (one of them is: if a noun has an adverbial or adjective modifier, then mark this noun as an aspect). Their methods allowed them to mark both explicit and implicit aspects. As a result of their studies, combining CNN and linguistic patterns gives more accurate outputs (precision, recall and f-score) [2].

The other research, which was about analyzing and comparing the sentiments on the internet and conducted by Liu, Bing et al., proposed a system analyzing customers' comments on different products to extract aspects and categorize them as positive or negative. Their method used in the paper is related with supervised pattern discovery and

natural language processing. Similar to the first research, they also defined some rules to extract aspects of opinion and applied automated opinion analysis based on these rules. But there are 2 main dissimilarity between these two papers. Firstly, the second one also proposed user interface to show visual diagram about the numbers of positive and negative opinions on features of different products. Secondly, the second paper proposed "Semi-Automated Tagging of Reviews", which means the analysts can review the results of the automated tool and make changes on it if there exists any problem/error. Moreover, there can be any situation that two or more words can be candidate as aspect, or automatic tool can choose wrong word. In order to minimize such errors, they used frequent terms method which also checks how many times the word is marked as aspect and makes correction based on this checking. [3].

In the research conducted by Amr El-Desoky Mousa and Björn Schuller, a new generative method for sentiment information measurement have been introduced. In this research, they used *contextual Birectional Long Short-Term Memory Language Model* (cBLSTM LM) as sentiment classification model. They used the IMDB dataset, which originally was presented by Mass et al. (2011). The dataset contains 50.000 balanced and labeled movie reviews posted on IMDB (25.000 of them are positive and 25.000 are negative). Firstly, they used distinct LM probability distributions for each type of the sentiment in the training data and then they utilize these models for categorizing their test dataset as positive and negative. They compared their results with discriminative classification method. As a result of this comparison, they noticed that their generative model yields considerably better results than do the other competing models. Besides that, they also observed that using both of the discriminative and generative models gives better performance in sentiment analysis [4].

There are some important benefits of the chatbots. Some of them were listed below with more details and examples [5].

1. Chatbots have an unbeatable ability to communicate with the users on their own. When someone needs information, the chatbot can help them. Especially, well trained chatbots can be better than traditional workers on specific types of tasks;
2. Almost 50 percent of the people prefer to communicate with the messages rather phone calls or emails. Considering this, the number of end-user can be higher by using chatbots;

3. Of course, chatbots are not so effective like human workers for some tasks. But on the other hand, especially for small companies who do not want to have many customer representative chatbot is better solution in terms of availability. Since it is online platform, the users can ask their questions or talk to it whenever they want;
4. The chatbots can also reduce the redundant works and wastes. Especially for simple tasks (for example, just getting information about any subject), usage of chatbot is better so that both the users and the organization will not lose time;
5. Sometimes people want to get specially prepared contents. By the usage of the NLP techniques, the chatbots can mimic the human messages and the end-user can be satisfied. Especially for suicide preventive chatbots, it is so important to satisfy the people with suicidal ideation and persuade them not to commit a suicide.

Besides these general benefits of the chatbots, there can be many other advantages of them for different business areas. For example, the bot can answer the questions about the price and other details of the products for sales company. Also, bots in the banking systems can enable the users to do some operations, like applying for the new credit card, without calling the customer representative or visiting the bank [5].

In another research by Lennart Hofeditz et al. about uses and applications of social bot in disasters, they investigated the first stage of the application of gentle social bots for managing emergency situations during the disasters. Their paper contains following three important research areas:

1. Current applications of social bots to provide communication in natural emergency situations.
2. Investigations of requirements and the possible problems while using of social bots which are needed to tackle in Australia.
3. The applications of social bots in controlling and managing the emergency situations and natural disasters (What tasks can be performed by these bots?)

In the first research area of their paper, they found that there are limited number of social bots in Twitter in Australia. Generally, the complexities of these bots are nominal. However, most of the companies in the country has already started to use the bot

applications to send automatic messages to the users, but they are in very simple level and not fully useful for emergency management and organizations need more complex bots for other tasks, like translation, app guides, etc [6].

Considering their second research area, they documented many requirements and possible problems in the use of bots. The most important requirements of the organizations for emergency management were reliability of the social bots and their active interactions with the users. Besides these, they also mentioned that it would be better if the bots can access the location of the users and connect to the other platforms. Also, the IoT bots were the one of the requirements mentioned in the paper [6].

In the final research area of their paper, they investigated the tasks which bots can execute during emergency situations. For this purpose, they interviewed with experts and listed their recommendations about possible task for bots. These recommendations are shown in the Figure 1 below. Moreover, they offered to apply the bots for many areas other than emergency management, like healthcare chatbots, and detecting and preventing domestic violence [6].

Name	Type	Phase	Description
Message translation bot	Chatbot	All phases	Providing a real-time translation for outgoing and incoming messages via for social media channels and app
Customer service bot	Chatbot	All phases	Providing a chatbot guide for the app of the emergency organisation
Prevention activity bot	News bot	Prevention	With systems linked bot with informs about hazard reduction fires based on geolocations
Preparedness messenger bot	Chatbot	Preparedness	Facebook messenger bot, which answers frequently asked questions automatically
Preparedness social bot	Social bot	Preparedness	Scouting for issues in communities and chatting with members on social media platforms
Smart speaker bot application	Social bot	Preparedness / Response	Emergency warning application for speech assistances using meta data such as location and app data
Emergency report bot	Chatbot / Monitoring bot	Response	Providing additional automated channel for accepting emergency reports on social media
Response messenger bot	Chatbot	Response	Answering questions regarding ongoing disasters such as information about bushfires

Intelligence bot	Monitoring bot	Response	Collection and analysis of disaster relevant material from social media platforms by using keyword and hashtag search
Emergency warning bot	News bot	Response	Automated caution & advice messages regarding the disaster on different channels
Information dissemination bot	Public dissemination bot	Response/ Recovery	Providing geolocation-based information about recovery centres, save zones and save roads
Misinformation fighting bot	Social bot	Response / Recovery	Fighting misinformation in social media by using keyword search and highlighting of false messages
Recruitment bot	Social bot	Recovery/ Preparedness	Finding and recruiting of suitable volunteers on social media platforms based on profile descriptions
Blame, anger and dissatisfaction warning bot	Monitoring bot	Recovery	Gathering and informing about dissatisfied content based on keywords and sentiments
Problem solving bot	Social bot	Recovery	Going back to people with reported problems and ask, whether the issues are solved
Feedback bot	Chatbot	Recovery	Asking for feedback regarding the work of the emergency organisation
Volunteer managing bot	Social bot	Recovery	Managing spontaneous volunteers by gathering volunteer requests and location-based needs on social media platforms
Recovery social bot	Monitoring bot	Recovery	Gathering the need combined with locations of people after a disaster and visualising that on a map by using keyword search and meta data search

Figure 1. Possible bot tasks suggested by experts *[6]*.

Although chatbots have been implemented in many different application areas and they can be implemented to fight against the suicide problems of the society, there are the huge gap in suicide detection and prevention by using chatbots still. There are still limited number of researches in this field.

3 BACKGROUND

3.1 Relevant concepts

Sentiment analysis is one of the most complicated tasks in IT-related sciences. Generally, there are 5 steps in implementation of sentiment measurement techniques [7].

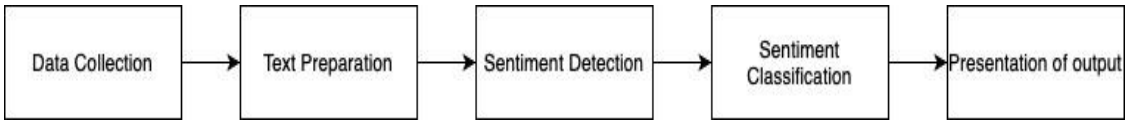

Figure 2. Steps of sentiment analysis implementation.

1. Data collection

Initial step of sentiment analysis is gathering user generated data. Main data sources are blogs, forums, social networks (Instagram, twitter, etc.). Most of time, these data are unbalanced, disorganized and can contain informal, daily life words, such as slangs and acronyms. Besides that, the size of such data is big enough. Considering these, it is obvious that making manual sentiment analysis on these data are practically impossible. Therefore, some extra techniques (like natural language processing) are required to extract and clean these data.

2. Text preparation

All data gathered from main sources (mentioned above) must be cleaned and irrelevant data must be eliminated before sentiment analysis. Furthermore, these data can be non-textual. In this case, these data must be converted to a standard form, so that sentiment analysis can be operated on them.

3. Sentiment detection

After cleaning data, many defined rules or methods are applied on them in order to obtain comparable results. For example, if extracted data are textual, the analysis must be based

on subjective sentences (author's opinions, belief and etc.), therefore factual information (objective sentences) should be eliminated. After that, specific machine/deep learning techniques are applied.

4. Sentiment classification

In this step, categories and rules for deciding which sentence is related to these categories are defined. Then, the results from sentiment detection are compared based on these rules and subjective sentences are classified according to defined categories.

5. Output presentation

After finishing analysis, it is important to extract meaningful and statistical information about data. Graphs and some other visual contents can be to show statistics about data. For example, if the selected data was classified in positive, negative and neutral, pie chart or bar chart can be used to visually show the result. Additionally, depending on the results of sentiment analysis, these graphs can contain the number of increase and decrease over a specified time. But the visual contents are not the only way for presentation. This step can be skipped and/or the results from previous steps can be used for further operations.

There are 4 main sentiment classification methodology: machine learning, lexicon-based, hybrid and n-gram modelling [7]:

1. In machine learning approach, there is train dataset which is used to fit in decided model. Then, the model is used to make predictions on test data. Accuracy of model in test data determines if the model is successful or not.
2. In lexicon-based analysis, train and test data are not required. A set of words and its sentiment value are defined, and analysis is operated based on words in a sentence and their sentiment values in this set.
3. Hybrid methodology uses the combination of previous methodologies. Since this approach uses the positive sides of others, it has great potential to improve the accuracy of existing models.
4. N-gram modelling is related with combinations of words. Since some words can have different meanings when they are used in combination of other words, only analyzing each word can possibly give results which are irrelevant or have less

accuracy. Depending on the number of words in a word combination, this approach can use uni-gram, bi-gram, tri-gram and etc.

Among these methodologies, first and second methodologies are the most commonly used. In machine learning methodology, supervised learning techniques are more useful than unsupervised learning. Also, there are 2 main types of lexicon-based approach of sentiment analysis: Dictionary-based and corpus-based approaches. Figure 3 shows the mostly used techniques of machine learning and lexicon-based sentiment analysis methodologies [7].

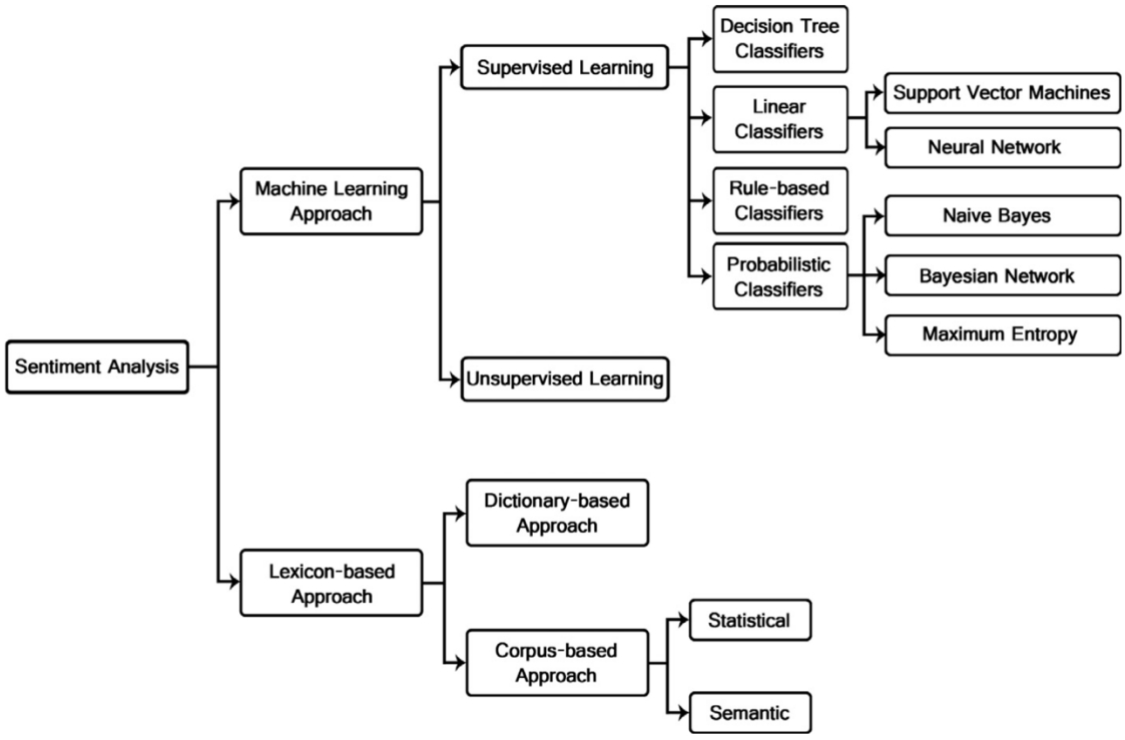

Figure 3. Sentiment analysis techniques.

Sentiment analysis can be operated in 3 different levels. Firstly, this analysis can be in document level. In this level, polarity of the whole document which contains opinions and thought about a topic or product is analyzed. It is obvious that document level analysis takes the whole document as input and gives one result (polarity value or class of the document). The next one is sentence level analysis. This level use almost identical techniques with the previous one, only difference between them is that sentence level

takes only one sentence and gives appropriate result. The last one is aspect level sentiment analysis. This type of analysis uses different techniques, because in this analysis, the results are given for each aspect and one sentence can contain more than one aspect. So, each aspect of sentence and the words related to it must be determined, sentiment analysis must be operated on these sections of sentence. Therefore, this level analysis requires more time and work than previous ones [7].

This research project is based on the machine learning approach. Also, the inputs that users type to the chatbot application of this research contain just a sentence in most cases Sometimes it can get a couple of sentences, but these can be thought as one sentence since they have common meaning)

3.2 Theoretical background

In this research, the author used *bidirectional Long-Short-Term-Memory, also known as bidirectional RNN*s, to train the first model in order to classify the user inputs as suicidal or not. Bidirectional LSTM is one of the kinds of Neural Networks. For the second model which learns to reply the user inputs to mimic the chatbot conversations, the model is based on encoder-decoder based sequence to sequence model. In this chapter, the author briefly explained the models used in the project.

1. What are Recurrent Neural Networks?

Recurrent Neural Networks (RNNs) are one of the branches of artificial neural networks. In simple Neural Networks (NNs), the inputs (and also the outputs) are not dependent of the others. But for more complex models, this idea does not work. For example, using traditional neural network for predicting the next word in the sentence will be bad idea, because usually the words in the sentences depend on the others and it would be better to add the previous words in the predictive model. For such kind of tasks in which the using of sequential information is needed, RNNs yields better results than traditional NNs. Since it makes the same calculations for each sequence element and outputs depend on the previous calculations, this type of the NNs is called recurrent. In the previous example, if it is needed to apply RNN to predict the next word, it can be thought that it

has a memory which holds the previous words and it predict the word concerning this "memory" rather than giving random predictions [8] [9]. Figure 4 shows how the RNNs look like:

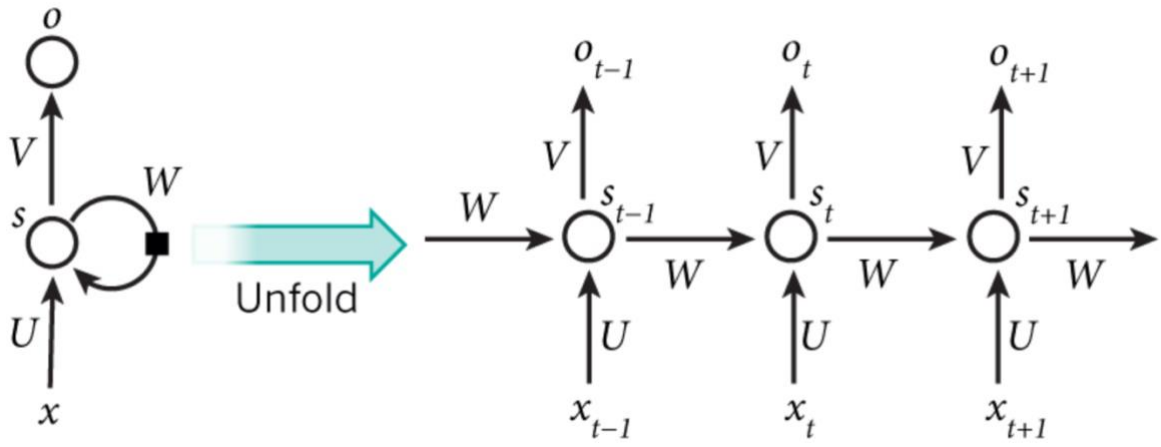

Figure 4. Recurrent neural network and the unfolding.

Figure 4 shows the unfolding of RNN to the full version. The folded version shows general idea of the RNN, but it can be hard to understand at the first glance. In unfolded version (also known as unrolled version), all sequences in the network are clearly written. For instance, to write the full recurrent neural network for the sentence which contains 5 words, the unfolded network will contain 5-layer. By other words, each word will be computed in the separate layer. Meanings of the symbols in the picture and the formula of RNN computation are following:

- x_t – it is the input of the sequence (layer) *t*. For example. x_0 and x_1 are the input vectors for the first and second words of the sentences (respectively)
- s_t – it is the hidden state of the layer *t*. It can be thought as the memory unit of the layer which allows us to predict each word considering the previous ones. By other words, calculation of s_t depends on the output of the preceding hidden state and the inputs of the current one. The mathematical representation of this calculation is as follows:

$$s_t = f\,(U\,x_t + W\,s_{t-1})$$

This *f* function is usually non-linear. Also, it is obvious that s_{-1} must be known in order to evaluate the first hidden state. In most cases, this value is initialized with zeroes.

- o_t – it is the output for the layer t. Considering the same example again, for predicting the next $(t+1)^{th}$ word of the sentence, the o_t value will be equal to the probability vector of the wordlist which defined before. The expression of output for the hidden layer is as follows:

$$o_t = softmax(V s_t)$$

2. What are Bidirectional RNNs (Bidirectional LSTM)?

Bidirectional recurrent networks (BRNNs) were firstly introduced in 1997. BRNNs are the extended version of simple RNNs in order to get more accurate outputs. In simple RNNs, all calculations in each hidden layer (state) depend on its inputs and the previous states. But it can be clearly understood from its name, each state gets information from its input, the previous states and also the next states for its calculations in BRNNs. Considering the example used before, all the words in the sentence are chosen based on the preceding and the succeeding words while predicting in RNNs [10] [11]. Figure 5 shows the structure of the BRNNs.

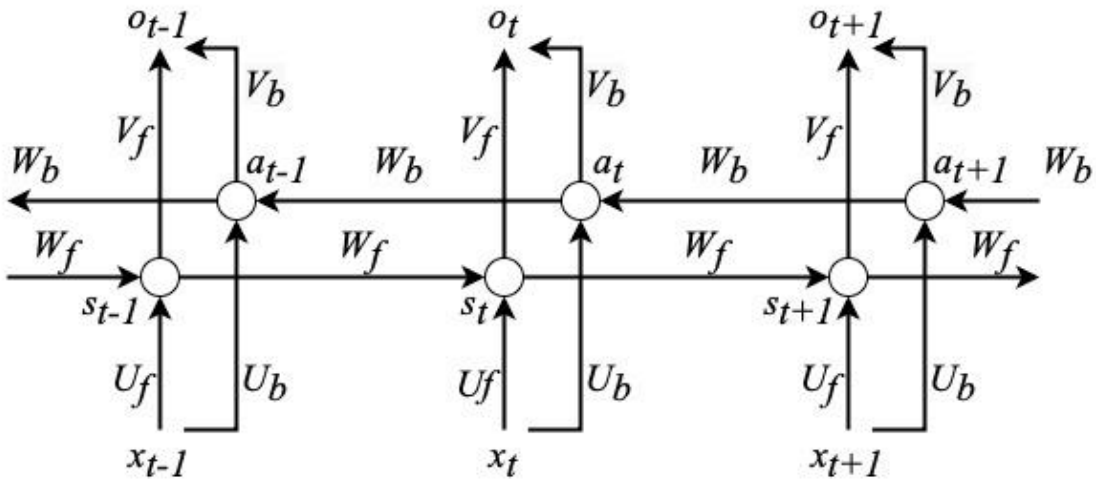

Figure 5. Structure of Bidirectional Recurrent Neural Network.

It shows the unfolded version of the BRNNs. In the figure, subscripted *f* means that the element (for example, input) belongs to the forward direction of the BRNNs, while subscripted *b* means backward direction. Of course, this version of the recurrent networks has some common features with the simple RNNs. For example, s_t is the hidden state of the forward direction of BRNNs, just like the simple version.

BRNNs differ from the simple RNNs with its hidden states in the backward direction (a_t). Unlikely to the simple version, each hidden state in this version takes its input and the information from the succeeding state, then makes calculation based on this information. Mathematical representation of this calculation is following:

$$a_t = f\,(U_b\, x_t + W_b\, s_{t-1})$$

The final output for layer t (o_t) is calculated based on the states in both forward and directions.

3. What is encoder-decoder based seq2seq model?

The sequence to sequence model (seq2seq) was firstly introduced by Google team in 2014. There are many application areas of the (seq2seq) models. Most use cases of this model are related with machine translation, speech recognition and online chatbots [12].

The main purpose of this model is to map the inputs and outputs with different lengths. Notwithstanding considering them separately, each of the inputs and the outputs have a fixed length. The most obvious example to this is translation applications. While translating a word from one language to another, the lengths of the sentence in the first language and the second language are different (for example, there can be 5 words in original, but its translation can be 10 words). It is obvious that regular models will not work for such cases. Therefore, sequence to sequence model is very useful for solving such downsides [12].

Figure 6 shows the general structure of the sequence to sequence model. Seq2seq models consist of 3 main parts: encoder, decoder and intermediate vector.

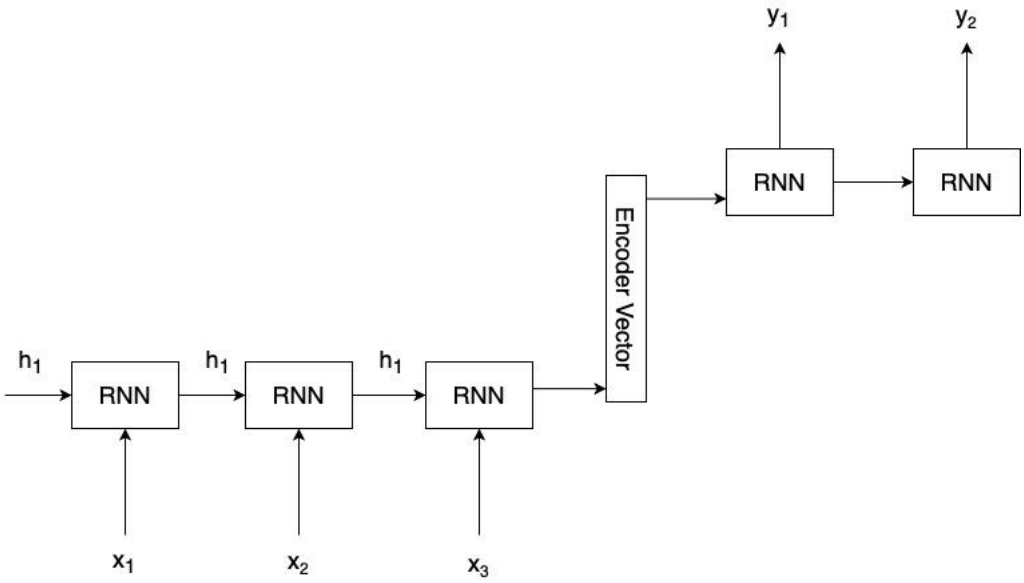

Figure 6. Structure of Sequence to Sequence models.

In the figure above, the part is the encoder part of the model and the part is the decoder part. Between them, there is encoder vector (also known as intermediate vector).

Encoder

Encoder part can be thought as the type of recurrent networks. Encoder consists of many recurrent components. Each component receives its own section of input sequence and gets appropriate information about previous, then sends it to the next one.

In chatbots or other applications that include conversation format, input sequence of the encoder is the all the words in user inputs. Each word of this inputs is shown as x_i and sent to the separate component.

The formula of each hidden state in encoder is below:

$$h_t = f(W_{(hh)}\, h_{t-1} + W_{(hx)}\, x_t)$$

As it is clearly seen from formula, for each hidden state, suitable weights are applied to the previous hidden state and input [12] [9].

Encoder Vector

Encoder vector is generated from the last state of the encoder of sequence to sequence model. Therefore, it uses exactly the same formula with other states of encoder. The main purpose of the encoder vector is to combine all the information for all inputs in order to enable the decoder to give predictions with higher accuracy. Besides these, the encoder vector can be thought as the first hidden state of decoder part of the sequence to sequence model [12].

Decoder

Decoder part of the sequence to sequence model consists of many recurrent components and each component gives prediction y_t. These components get the information from the preceding state and propagate it to the next.

Considering the example mentioned in the encoder part, the output sequence is all the words in the answer of the chatbot. Each word of this answer is the output of the recurrent components.

The formula of each hidden state in encoder is below:

$$h_t = f(W_{(hh)}\, h_{t-1})$$

As it is obvious, only the previous hidden state with its weight is used for each hidden state of the decoder. There is no input sequence in this part [12] [9].

4 PROJECT OVERVIEW

The source code of the project is available in the following link:

https://bitbucket.org/RufatVali/suicide-detect-thesis/

4.1 Scope of the project

In this research project, the author developed the chatbot project. The project is in conversation format which means the user can continuously enter new text message until he or she exit from the project. The chatbot automatically analyze the message of the user and decide whether it contains the suicide-related content or not. After that, it gives appropriate message to prevent the user from suicide.

The user interface and the typical conversation in the chatbot are shown in Figure 7.

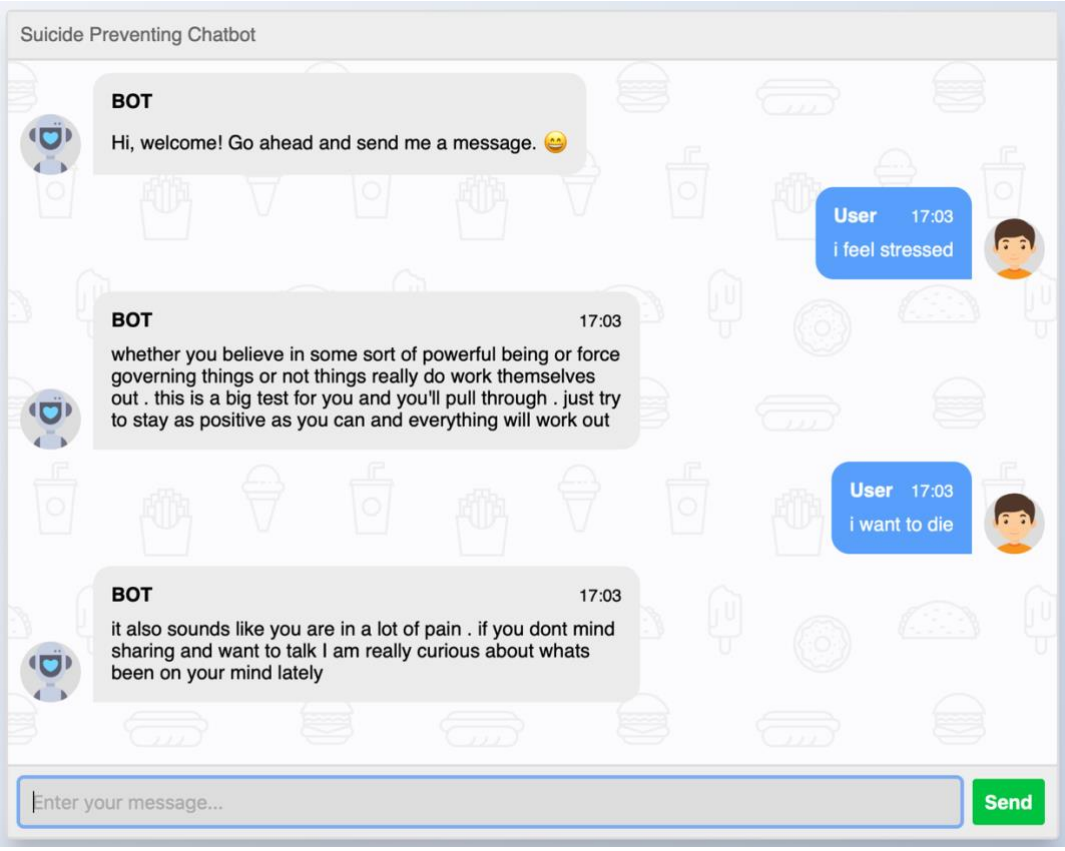

Figure 7. UI and typical conversation in chatbot.

4.2 Tools and overview

The project was fully developed in Python programming language. For applying the deep learning algorithms and techniques, Keras (Python library) was used.

Keras is an open-source library of the Python. The main objective of this library is to allow the developers to make quick experimentations on neural networks [13]. The main features of Keras is its user-friendliness and modularity. Keras allows us to easily implement the main blocks of the artificial neural networks, for example layers, optimizers and activation functions [14]. It has a lot of tools which help us to simplify the required code for creating deep neural network in order to work with text and image data. Besides these, Keras supports specific types of neural networks (convolutional and recurrent neural network). Since the author decided to use recurrent neural network, Keras was helpful for our project.

The project contains 5 main steps:

1. Data analysis
2. Data preprocessing
3. Model creation
4. Model training
5. Model deployment

Each step of the project and validation of results were analyzed separately in the next chapter.

5 PROJECT DEVELOPMENT

5.1 Data Analysis

In the past, people with mental health problems, such as depression, suicidal ideation and etc. tended to be more asocial and did not want to share their thoughts with others [15]. As a result of this trend, it was so difficult to get data about such kind of people and prevent them from self-harm. But, thanks to the modern advances in social media platforms and increase in their usage, they started sharing more status and/or comments about their problems and looking for the help from others, in most cases with anonymous profiles. So, the number of data about them are increasing and we can discuss about more preventive plans for them by using these data.

During the project, 2 different datasets were used. The first one was trained in order to decide whether the user message is suicidal intended or not. On the other hand, for suicidal intended text, the author trained the second dataset to enable chatbot gives appropriate responses. The datasets were analyzed below with details

As mentioned before, the first dataset was used to analyze the user's messages. In order to keep their privacy, personal data of the comment owners are not used in the dataset. This dataset was collected with Twitter API and firstly used in the research [16] by Vioulès et al in 2018. During the collection of the data, they took the advantage of the two main resources and prepared the main suicide-related keywords:

1. The words and phrases which were defined as the main reasons of suicide and/or risky by American Psychiatric Association (APA) [17]
2. The words which were defined as warning and critical by American Association of Suicidology (AAS) [18].

Then, they collected the dataset of the tweets by using Twitter API for these defined key words.

In general, the dataset contains the posts from two different types of the people: problematic and normal people. The first category of the users is the people who often

share the posts related to the depression, problems, ideas about suicide and other negative posts. The second type of the users is the ordinary users who have not the signs of the any types of self-harm attempt.

The author used the partition of this dataset for the project. This new dataset was firstly prepared by Safa Abdellatif and used for his doctoral thesis [1]. In the preparing process of the dataset, 260 random tweets were chosen from both user categories (100 of them were from problematic users and the rest of them were from ordinary users). Then, these posts were analyzed by experts and classified as suicidal or not. Then the tweets were labeled as “Yes” or “No”.

As a result of all these processes, the final dataset contained 96 posts with suicidal content and 164 non-suicidal posts (totally 260 posts). The samples from this dataset are shown in the Figure 8.

When you finally hear that one song!	non
Invest in yourself. Make yourself better	non
Triple over time win. Imagine that	non
RT @girlideas: marry someone u want to annoy for the rest of your life	non
Summer can't get here fast enough. (Emoji sad)	non
RT @girlideas: if u don't sing at the top of ur lungs while driving then what do u do??????	non
RT @girlideas: rt if mozzarella sticks have impacted ur life in a positive way	non
RT @cynthiaovbiojie: Want to go swimming, then eat crawfish afterward	non
I just love my best friend (Emoji loveeyes)	non
@babi_lexi lol if it's who I'm thinking you stupid !	non
July needs to hurry and get here .	non

i don't think i'm going to make it out of this year alive	oui
I can't take this any longer	oui
Tired of the pain I feel inside everyday	oui
I want to cut, I want to cut, I want to fucking cut the shit out of me !!!	oui
I can't do this anymore	oui
I just wanna give up	oui
I want to die	oui
I could end my life and nobody would care	oui
I should kill myself tonight	oui
Nothing can save me	oui
Trying to find away to leave this world	oui

Figure 8. Samples from the first dataset (oui = yes, non = no).

The second dataset was used for training the bot to answer the users' message and mimic as the chat communication (conversation format). This dataset was prepared by Manas Gaur et al. for their research [19]. The dataset was created based on Reddit users' conversations and discussions on five different levels of suicidal thoughts (indicator, supportive, ideation, behavior and attempt). This dataset contains 500 Reddit posts totally. Figure 9 shows the samples from this Reddit dataset

User	Post	Label
user-0	['Its not a viable option, and youll be leaving your wife behind. Youd Pain her beyond comprehension.It sucks worrying about money, I know that first hand. It can definitel	Supportive
user-1	['It can be hard to appreciate the notion that you could meet someone else who will make you happy when you are so deeply in love with your boyfriend. Your desires are	Ideation
user-2	['Hi, so last night i was sitting on the ledge of my window contemplating whether or not i should jump. My dad had just choked me and told me that i should get out of th	Behavior
user-3	['I tried to kill my self once and failed badly cause in the moment i wanted to do it i realized that i want to live! I still have Suicidal thoughts and i often question myself w	Attempt
user-4	['Hi NEM3030. What sorts of things do you enjoy doing?', 'Personally, I always welcome music suggestions with open arms. Nothing like losing yourself in music, escapin	Ideation
user-5	['Since I dont know what DBT is, would you mind explaining it to me a little bit more? I am really sorry to hear that someone who is supposed to be helping you is actuall	Supportive
user-6	['No matter what you ever think, there will be people who care. Whether they are people like me, who youve most definitely never talked to before or seen, or even heard	Supportive
user-7	['Dont see it as failing at killing yourself, theres a reason why you lived. Theres something for you here on Earth. If you need someone to talk to you can come to me and I	Ideation
user-8	['The reason I have faith in our species ability to spread and survive is that homo sapiens have surpassed themselves over and over again throughout their existence. The s	Supportive
user-9	['A book is usually what I do when Im getting down, but it doesnt work when I start getting panicky. Ill try the carbs, the caffeine doesnt work because Ive gotten it in a m	Ideation
user-10	['Dont do it man. Seriously this is making me sad. I dont know you but I feel like youve got something to offer. Everyone does. Maybe not now, maybe next year, but sor	Indicator

Figure 9. Samples from the Reddit dataset.

These datasets are in raw format and could not be directly used in further operations. Therefore, they need to be prepared.

5.2 Data Preprocessing.

As mentioned before, because the raw materials can contain some irrelevant and useless data, they cannot be used in sentiment analysis operations. So, the text preprocessing has the crucial role in sentiment analysis.

The first dataset which the author used in the project was collected from Twitter. Since Twitter is used by all kind of people and it has its own special symbols, it is also needed to clear them and make standard for all sentences. Following list contains the most common cases in Twitter posts and its examples in our dataset:

1. In Twitter, users can re-share the posts from other users. In these cases, the re-shared posts contain the word "RT" followed by the "@" sign and the username of the user which originally posted the text (tweet). For example:

RT @girlideas: marry someone u want to annoy for the rest of your life

2. Twitter uses hashtag symbols to categorize the posts and enable users to search the posts which they are interested in. But it is obvious that this symbol has no meaning in language and has no value for our search. For example:

Happiness in a bottle. Let the weekend begin! #100happydays

Since the Twitter users mostly use informal language in their posts, there can be some other kinds of problems in their posts, like incorrect use of the punctuations and lowercase/uppercase letters. At first step, in order to remove all these cases that can cause a problem to the model, the following steps was operated:

1. All punctuations were removed
2. All words were lowercased

3. All non-English words and characters were removed
4. Sentences were split into words

After this cleaning process, further operations must be done on the dataset. The textual datasets cannot be used to train deep learning models directly. For this purpose, all words were integer encoded with OneHotEncoder in order to be used as inputs and outputs for the selected models. These operations are also supported in Keras library.

After applying integer encoding, the words become the integer vectors which contains a lot of zeroes and have very high dimensionality. But still there are two main problem in these datasets. Firstly, especially for the very large datasets, this high dimensionality can seriously affect the performance of the application. In addition to that, integer encoding assigns the specific number to the words, but it does not consider the meaning of the words. As a result of this, many words which have exactly the same meanings are considered as a different word. For example, the words “airplane” and “aircraft” are considered as different words, but it is obvious that the words share the same meaning and can be used interchangeably. Considering the high dimensionality and semantic problems of the integer encoded, the final operation was done on the dataset to make it ready for using our models. Therefore, GloVe Embedding with 100 dimensions is used for getting contextual meaning of the dataset.

Considering the second dataset (Reddit dataset), this set contains 2 extra columns which would not be used in the project. Firstly, the user column (means user ID) did not make any sense. Also, the dataset contains the labels (5 categories), since the original research, which this dataset was prepared for, did used these labels for further operations. But since the author used this set of posts as a message to the user, these labels were not used. Besides these, four cleaning steps which mentioned before are also operated for this dataset

5.3 Model Creation.

First model was created using Keras to classify sentences as suicidal or not. Bidirectional LSTM with embedding layer is used to train the model. the following figure shows the

general overview of the model. In this model, pretrained GloVe embeddings used for the embedded layer.

```
model = Sequential()
model.add(Embedding(num_words,
                    embedding_dim,
                    embeddings_initializer=Constant(embedding_matrix),
                    input_length=maxlen,
                    trainable=True))
model.add(SpatialDropout1D(0.2))
model.add(Bidirectional(CuDNNLSTM(64, return_sequences=True)))
model.add(Bidirectional(CuDNNLSTM(32)))
model.add(Dropout(0.25))
model.add(Dense(units=2, activation='softmax'))

model.compile(loss = 'categorical_crossentropy', optimizer='adam',metrics = ['accuracy'])
```

The first layer (Embedding layer) helps us to initialize the word embeddings. Also, this layer has some weights and they are optimized according to the activation function on each epoch in order to minimize the errors [20].

The next layer of the model (Spatial dropout layer) is a type of dropout regularization and of performs almost the same role with simple Dropout in Keras. Dropout regularization is one of the easiest ways to standardize the deep neural network. Main purpose of this regularization is to minimalize the overfitting in the model [21]. In our model, the dropout rate was set to 0.2.

The following two layers of the model is Bidirectional LSTM layers. These layers are succeeded by simple Dropout layer with dropout rate of 0.25.

The final layer of the model (Dense layer) deals with the output of the model. Its units parameter means that the output has two categories. The activation function for this layer is softmax.

The following figure shows the general overview of the model.

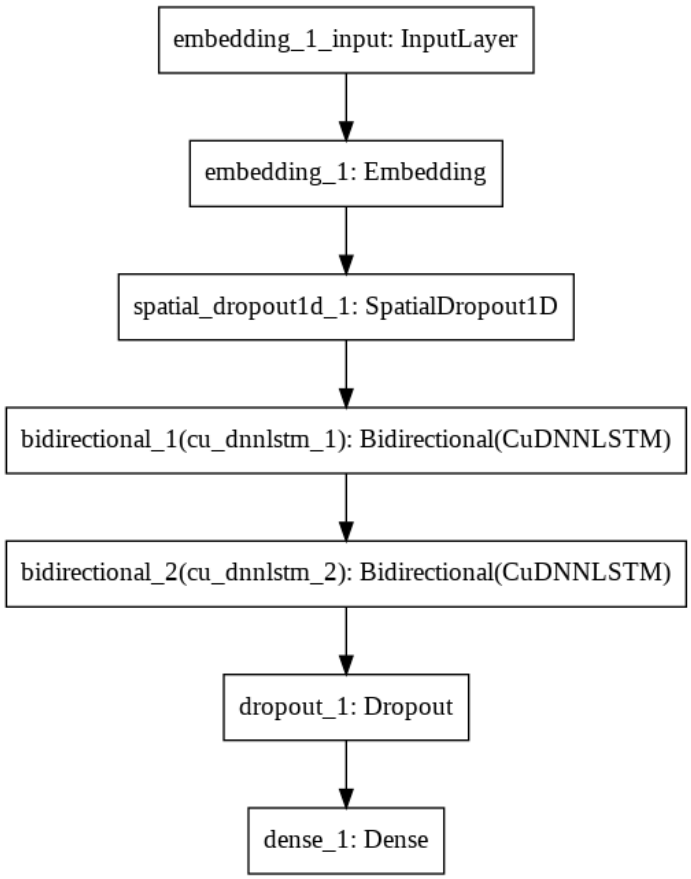

Figure 10. Structure of Bidirectional LSTM model for suicide classification

The second model of the project which deals with the chatbot's answers to the user's inputs was developed based on encoder-decoder based sequence to sequence model.

```
model = Sequential()

model.add(Embedding(num_tokens, HIDDEN_UNITS, input_length=max_seq_length,
mask_zero=True,
embeddings_initializer=Constant(embedding_matrix),trainable=False))

model.add(Bidirectional(LSTM(HIDDEN_UNITS, return_sequences=True)))

model.add(Bidirectional(LSTM(HIDDEN_UNITS)))
```

```
model.add(RepeatVector(max_seq_length))


model.add(LSTM(HIDDEN_UNITS, return_sequences=True))


model.add(TimeDistributed(Dense(num_tokens, activation='softmax')))
model.compile(optimizer='adam', loss='categorical_crossentropy',metrics=['accuracy'])
```

Like the previous one, this model also has the embedding layer for initializing the word embeddings. HIDDEN_UNITS value was defined as 256. The outputs of each hidden states are accessible by setting return_sequences as true.

The following table summarize the layers of the model and its outputs.

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_3 (Embedding)      (None, 50, 256)           1265920
_________________________________________________________________
bidirectional_3 (Bidirection (None, 50, 512)           1050624
_________________________________________________________________
bidirectional_4 (Bidirection (None, 512)               1574912
_________________________________________________________________
repeat_vector_2 (RepeatVecto (None, 50, 512)           0
_________________________________________________________________
lstm_6 (LSTM)                (None, 50, 256)           787456
_________________________________________________________________
time_distributed_2 (TimeDist (None, 50, 4945)          1270865
=================================================================
Total params: 5,949,777
Trainable params: 4,683,857
Non-trainable params: 1,265,920
_________________________________________________________________
```

Figure 11. Summary of seq2seq model

5.4 Model Training.

In the first model, the dataset was divided into the train set and test set. Totally, 90 percent of the dataset was assigned as train set, the remaining was assigned as test set. These sets were fitted to the model with epoch 50 and batch size 10.

On the other hand, the second dataset was divided into 80 percent trainset and 20 percent test dataset. These datasets were fitted to the model with epoch 20 and batch size 64.

Both models were saved for the future development

5.5 Model Deployment.

Flask is lightweight web framework written in Python. The main purpose of the Flask is to enable the developers to create simple and easy applications, but it also offers other extensions in order to upgrade the application to more complex structures. Since the project is one-page application and does not have any complex structure, Flask is used in creating the chatbot application of the project.

5.6 Validation of Results.

For validating the results of the models, the accuracy scores of the both models were used. As mentioned before, these models were divided into train and test datasets.

As mentioned above, the first model was trained 50 epochs. In the first epoch, the accuracy of the model was around 74 percent. Then, the accuracy score gradually increased and it got maximum value in 20th epoch. After completing all epochs, it had almost 99.6 percent accuracy on train dataset.

On the other hand, the accuracy score of the test dataset (validation dataset) is less than the train dataset. In the first epoch, its accuracy was around 65 percent and then it decreased. The final accuracy score was 50 percent on test dataset. But, since the size of the test dataset was less than usual, so this will increase if many new sentences is added to the dataset.

The second model has the dataset which includes more data than the previous. Therefore, this one gives more realistic accuracy rate. This model also had less accuracy scores for initial epochs and increased. Finally, the model has approximately 80 percent accuracy on the train dataset and 86 percent accuracy on the test dataset.

6 FUTURE WORK

Since this chatbot application is MVP, there can be some improvements in the future.

Firstly, the datasets contain relatively less data, so there may be such cases that the bot cannot detect suicide intention in the user messages. In order to solve this problem, the size of the datasets can be increased so that the bot has more train data about messages with suicide contents.

Secondly, the application can be extended to the more complex structure by adding user profiling. By this way, the history of the messages per user can also be kept and try to track their emotional feelings with more accuracy.

Furthermore, this project has separate web application and the users need to enter it to talk with the bot. But, the usage of the model that detect the suicidal contents can be extended by implementing it in the most famous social media platforms in which almost everyone uses in the daily life. By this way, it is possible to have the detailed statistics about the people who have suicidal ideations and provide better preventive plans against such kinds of ideations.

7 SUMMARY

This project is chatbot to prevent its users from the suicide. This conversation bot can classify the suicidal messages from user inputs and give appropriate messages. As training data, the anonymous posts from social media platforms were used to train the chatbot.

For the chatbot development, NLP and sentiment analysis techniques were used for classification of inputs and preparation of output.

There are many different techniques and methodologies that were used for sentiment information measurement. These methodologies were also investigated during the research.

Moreover, the research also contains the analysis of the application areas of the chatbot, its benefits and the tasks that the experts thinks the bots can accomplish.

The project contains 2 main parts: sentiment analysis for detecting the suicidal contents and the chatbot implementations. For the first model to detect suicide ideations, Bidirectional LSTM with embedding layer was used. As a train dataset, unbalanced Twitter dataset was used. Pretrained GloVe embeddings were used for the embedding layer.

For the second part of the project (predicting the bot's answers), the encoded-decoded based seq2seq model was used and 500 comments from suicide-related posts in Reddit were trained.

REFERENCES

[1] S. Abdellatif, Classification of Imbalanced Datasets: Application to Sentiment Analysis, University of Tunis El Manar, 2020.

[2] S. Poria, E. Cambria and A. Gelbukh, Aspect Extraction for Opinion Miningwith a Deep Convolutional Neural Network, 2016.

[3] B. Liu, M. Hu and J. Cheng, Opinion Observer: Analyzing and Comparing Opinions on the Web, University of Illinois at Chicago, 2005.

[4] A. E.-D. Mousa and B. Schuller, Contextual bidirectional long short-term memory recurrent neural network language models: a generative approach to sentiment analysis, 2017.

[5] M. D. Vivo, How chatbots can refine the customer acquisition process, 2017.

[6] L. Hofeditz, C. Ehnis, D. Bunker, F. Brachten and S. Stieglitz, Meaningful use of social bots? Possible applications in crisis communication during disasters, Stockholm & Uppsala: In Proceedings of the 27th European Conference on Information Systems (ECIS), 2019.

[7] A. D'Andrea, F. Ferri, P. Grifoni and T. Guzzo, Approaches, Tools and Applications for Sentiment Analysis Implementation, nternational Journal of Computer Applications (0975 – 8887) Volume 125 – No.3, 2015.

[8] D. Britz, "Recurrent Neural Networks Tutorial, Part 1 – Introduction to RNNs," 17 09 2015. [Online]. Available: http://www.wildml.com/2015/09/recurrent-neural-networks-tutorial-part-1-introduction-to-rnns/. [Accessed 20 03 2020].

[9] D. Persiyanov and L. Belokon, "Chatbots With Machine Learning: Building Neural Conversational Agents," 15 09 2017. [Online]. Available: https://dzone.com/articles/chatbots-with-machine-learning-building-neural-con. [Accessed 17 03 2020].

[10] M. Berglund, T. Raiko, M. Honkala, L. Kärkkäinen, A. Vetek and J. T. Karhunen, Bidirectional Recurrent Neural Networks as Generative Models, Advances in Neural Information Processing Systems 28, 2015.

[11] M. Schuster and K. K. Paliwal, Bidirectional recurrent neural networks, IEEE Transactions on Signal Processing 45, 1997.

[12] S. Kostadinov, "Understanding Encoder-Decoder Sequence to Sequence Model," 05 02 2019. [Online]. Available: https://towardsdatascience.com/understanding-

encoder-decoder-sequence-to-sequence-model-679e04af4346. [Accessed 23 03 2020].

[13] Keras, "Documentation. Why choose Keras?," [Online]. Available: https://keras.io/why_keras/. [Accessed 30 03 2002].

[14] Keras, "Core Documentation, Core layers," [Online]. Available: https://keras.io/api/layers/core_layers/. [Accessed 30 03 2020].

[15] L. Brådvik, Suicide Risk and Mental Disorders, Int J Environ Res Public Health, 2018.

[16] M. J. Vioulès, B. Moulahi, J. Azé and S. Bringay, Detection of Suicide-Related Posts in Twitter Data Streams, Ibm Journal of Research and Development 62(1), 2017.

[17] A. P. Association, Practice guideline for the assessment and treat- ment of patients with suicidal behaviors, Am J Psychiatry, 2003.

[18] M. D. Rudd, A. L. Berman, T. E. Joiner, M. K. Nock, M. M. Silverman, M. Mandrusiak and K. V. Orden, Warning Signs for Suicide: Theory, Research, and Clinical Applications, Suicide and Life-Threatening Behavior 36(3), 2006.

[19] M. Gaur, A. Alambo, U. Lokala, U. Kursuncu, K. Thirunarayan, A. Gyrard, A. Sheth, R. S. Welton and J. Pathak, Question Answering for Suicide Risk Assessment Using Reddit, IEEE 13th International Conference on Semantic Computing (ICSC), 2019.

[20] Kerad, "Documentation Embedding Layer," [Online]. Available: https://keras.io/api/layers/core_layers/embedding/. [Accessed 15 04 2020].

[21] J. Brownlee, "How to Reduce Overfitting With Dropout Regularization in Keras," 03 10 2019. [Online]. Available: https://machinelearningmastery.com/how-to-reduce-overfitting-with-dropout-regularization-in-keras/. [Accessed 20 04 2020].