Medical Chatbot

Project report submitted in partial fulfillment of the requirement of the degree of

Bachelor of Technology

Prepared by:

Yogita Bala Singh (1505626), Gaurav Singh (1505627), Zeeshan Ali Naqvi (1505632), Siba Prasad Tripathy (1505638) and Sayan Chakraborty (1521039)

KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY (KIIT)

Deemed to be University U/S 3 of UGC Act, 1956

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY

Deemed to be University, BHUBANESWAR

(**March 2018**)

Table of Contents

1. Introduction

1.1 Purpose

Everyone knows what Chatbots are. They are “computer programs which conduct conversation through auditory or textual methods”. Google’s assistant, Microsoft’s Cortana, Amazon’s Alexa and Apple’s Siri are probably the most popular assistants which one knows of as of now. They try to understand what the user needs and accordingly come up with a proper answer to meet your requirements. Obviously, they are not 100% correct. But with proper training and time, it is thought that a moderately advanced level can be achieved which will be able to help people in their general day to day life.

These programs can help you do basic stuff like checking the weather, movie ratings, game scores to advanced stuff like translating a language for you.

In general, most of these programs have an auditory interface which helps the user for a more hands free experience. Unlike such assistants, chatbots can be thought to be more on the textual side. Where the conversation is mostly operated on textual fronts.

1.2 Document Conventions

The standards or typographical conventions that were followed are mostly independent of any other documentations as of now. The higher priorities are mostly described here in and is independent of a secondary read.

1.3 Intended Audience and Reading Suggestions

The archive is composed as a fundamental execution model of the minor venture proposed for the engineers and scholarly personals who have a general idea of the Natural Language Processing and general information arranging and alteration. Whatever is left of this SRS contains the different organizations in which the task was outlined and executed. In spite of the fact that there is further extent of changes which is all the more intricately determined later on in the archive. It is proposed that individuals experience the nuts and bolts of Natural Language Processing like stemming, lemmatizing and evacuating stop words.

1.4 Product Scope

The product is mainly intended for the people of rural areas where they are mostly deprived of facilities like the internet connection. The aim is to create an easy to use disease and treatment

(mostly medicine) suggestion system that would help them understand what kind of a disease they may have. Other than that, as a major scope the system is thought upon to be integrated in with SMS, so that it remains independent of internet connection.

1.5 References

- For Natural Language Processing and Data Preprocessing:
  - www.pythonprogramming.net
  - https://www.stackoverflow.com
  - http://www.nltk.org/book/
  - https://pypi.python.org/pypi/stemming/1.0
  - https://pythonprogramming.net/stemming-nltk-tutorial/
  - http://www.nltk.org/howto/stem.html
  - https://pythonprogramming.net/lemmatizing-nltk-tutorial/
  - https://pandas.pydata.org/pandas-docs/stable/
  - http://stackexchange.com

- For Dataset (Secondary):
  - https://www.kaggle.com/plarmuseau/symptom-disease-recommender/data

- For SMS Encryption:
  - https://security.stackexchange.com/questions/11493/how-hard-is-it-to-intercept-sms-two-factor-authentication

- For SRS document:
  - https://web.cs.dal.ca/~hawkey/3130/srs_template-ieee.doc

2. Overall Description

2.1 Product Perspective

This project is mostly a standalone implementation, without any previous follow-on. The SRS defines the major theory on how the model can be implemented in general and the possibilities of any future modifications to help in the overall development of the project on a major business scale .

2.2 Product Functions

The major functions of the product can be summarized as follows:
The product is mainly aimed at suggesting a series of diseases and their treatments which can be the result of a particular series of symptoms (which would be user specified) along with suggesting whether they can be life threatening or not. The program is mainly based on a simple string matching algorithm from a prebuilt and modified dataset after proper processing of the input string through NLP.

2.3 User Classes and Characteristics

Currently, at this stage the program is only meant to be used by any personal, who has the simple knowledge of python code compilation. After further improvements the project can be integrated with a SMS service, wherein the program will be easy to use by any people who is able to send a text through SMS.

2.4 Operating Environment

The program will run on any operating system having python versions 3.x installed. The program will be currently running only on the Terminal or Command Prompt without any UI implementation.

Current System where the program is tested and analyzed:

Operating System Overview:

- ✓ Windows 10 with
  - Intel® Core(TM) i5-5200 CPU @ 2.20GHz 2.20GHz
  - Memory: 16 GB
  - Architecture: 64 bit

- ✓ Linux Ubuntu 16.04 LTS with
  - Intel® Core(TM) i5-5200 CPU @ 2.20GHz 2.20GHz
  - Memory: 16 GB
  - Architecture: 64 bit

2.5 Design and Implementation Constraints

The program currently does not have any graphic interface (Dashboard, UI etc.) which would help the general people interact with it. The program is currently only restricted to the system terminal.

2.6 Assumptions and Dependencies

The project has a dataset from Kaggle which a set of attributes specified for a “disease recommendation system”. The dataset has been preprocessed and dealt with for implementation in our project. As of another assumption has been made with the weightage of a diagnosis specified. Here the higher weightage specifies a higher number of diagnosis for a particular symptom. The series of diagnosis is suggested according

3. Implementation

3.1 Proposed Model

As of now, we will be applying NLP. As a large scale, we will try to use a Recurrent Neural Network, where the inputs will go through a seq2seq model, and generate the proper output. The main advantage of a Neural Network is that it will be able to predict diseases based on particular symptoms, on its own, once it is properly trained.

3.2 Model Description

As said, the present model is based on a simple Natural Language Processing scheme, where we have done the following

:

1. Preprocessed the data:

   Our data is a secondary dataset, based on a disease recommendation system. (dataset)

   After preprocessing we had 4 final datasets

   a. The first one containing the list of **symptoms** along with unique symptom ids (sid) and their symptoms
   b. The second one containing diseases along with unique **diagnosis** ids (did).
   c. The third one containing a list of appropriate **medicines** or treatments for the specified diagnosis.
   d. As a measure of reference to the proper diagnoses from the symptoms, we had a separate dataset denoting the references to the proper diseases from their symptoms through the unique ids

2. The user input string was extracted:
    - Removed the unnecessary words from the input string.
    - Porter Stemmer Algorithm was applied to stem the input string and lemmatized to a proper matching format.

3. The symptoms dataset was modified for proper string matching:
    - The symptoms dataset was stemmed and lemmatized to have proper results on string matching.

4. Thereby, an intersection was performed between the input string and the symptoms dataset to find the most appropriate match.

5. The matching was further refined by inducing a secondary query, where the user was prompted to enter a more appropriate symptom based on the matching symptoms from the data.

6. Thereby the proper diagnosis of the disease and the medicine/treatment was suggested based on the symptoms chosen by the user

.

7. The suggestion was based on the weights on the diagnoses, which indicated the number of times a particular diagnosis was made for a corresponding symptom.

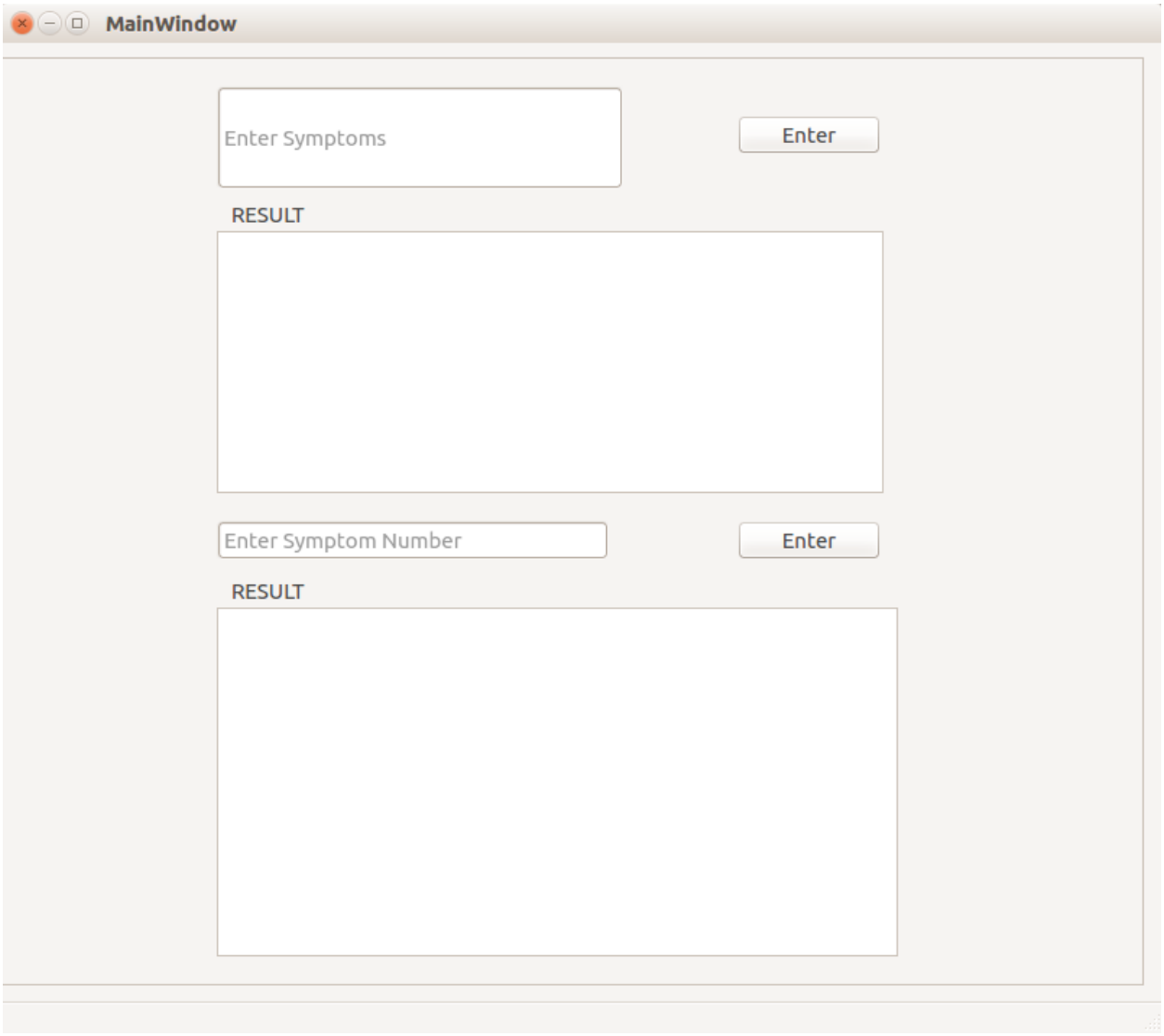
Python Native Packages installed on the machine during time of implementation:

- Nltk
- Pandas
- Numpy
- Matplotlib
- PyQt4

4. Interface Requirements (Future Perspective)

4.1 User Interfaces

As of now the basic mode of interaction is a basic GUI, which would take input the symptoms and spit out the proper diagnosis and medicines. The GUI is designed with Qt Creator (pyQt4)



4.2 Communications Interfaces

Presently, the main mode of interaction for the user a simple GUI. Later on a major scale the communication interface is thought upon, to be through any SMS Gateway servers which would allow us to send and receive messages through generic SMS.

Message Format:

The pertinent message format is of a maximum of 160, 7-bit characters (140 octets).
As of characters, 7-bit characters from the 3GPP 23.038 GSM character set are the default although other character sets may be supported by specific applications. The use of another character set may impact the character limit
e.g., UCS-2 16-bit characters, results in 70-character messages.

If other character sets are used, applications handling SMS messages are required to map the character sets to and from the character set used for SMS messages. Implementations may choose to discard (or convert) characters in the message body that are not supported by the SMS character set they are using to send the SMS message. If they do discard or convert characters, applications must notify the user.

Encryption and Security:

GSM incorporates some security through cryptography. The cell phone and the supplier (i.e. the base station which is a piece of the supplier's system) validate each other generally to a common mystery, which is known to the supplier and put away in the client's SIM card. A few calculations known under the code names "A3" and "A8" are engaged with the validation. At that point the information (as sent through the radio connection) is scrambled with a calculation called "A5" and a key got from A3/A8 and the common mystery.

There are a few genuine calculations which cover up under the name "A5". Which calculation is utilized relies upon the supplier, who, thus, is obliged by nearby directions and what it could permit from the GSM consortium. Additionally, a dynamic aggressor (with a phony base station) can possibly compel a cell phone to utilize another variation, unmistakable from what it would have utilized something else, and there are very few telephones which would caution the client about it (and even less clients who might think about it). **A5/0** means "no encryption". Data is sent unencrypted. In some countries, this is the only allowed mode (I think India is such a country).

- **A5/1** is the old "strong" algorithm, used in Europe and North America.
- **A5/2** is the old "weak" algorithm, nominally meant for "those countries who are good friends but that we do not totally trust nonetheless" (it is not spelled out that way in the GSM specifications, but that's the idea).

- **A5/3** is the newer algorithm for GPRS/UMTS.

A5/3 is a piece figure otherwise called KASUMI. It offers better than average security. It has a couple of inadequacies which would make it "scholastically broken", yet none extremely relevant practically speaking.

A5/2 is in fact frail. The assault requires a small amount of a moment, subject to a precomputation which takes not as much as a hour on a PC and requires a couple of gigabytes of capacity (very little). There are specialized points of interest, generally on the grounds that the GSM convention itself is mind boggling, yet one can accept that the A5/2 layer is weak.

A5/1 is stronger, but not very strong. It uses a 64-bit key, but the algorithm structure is weaker and allows for an attack with complexity about $2^{42.7}$ elementary operations. There have been a few productions which pivot this many-sided quality, for the most part by doing precomputations and sitting tight for the calculation inside state to achieve a particular structure; albeit such distributions publicize somewhat bring down multifaceted nature figures (around 240), they have disadvantages which make them hard to apply, for example, requiring a great many known plaintext bits. With only 64 known plaintext bits, the raw complexity is $2^{42.7}$.

The extent of the inward province of A5/1, and the way A5/1 is connected to encode information, additionally make it helpless against time-memory exchange offs, for example, rainbow tables (Barkan-Biham-Keller). This accepts the assailant ran once a genuinely gigantic calculation and put away terabytes of information; a short time later, the online period of the assault can be very quick. Points of interest calm a bit, contingent upon how much storage room, CPU control is accessible for the online stage, and to what extent would one be able to be prepared to sit tight for the outcome. The underlying calculation stage is tremendous yet innovatively possible (a thousand PC should be sufficient); there was an open dispersed task for that yet I don't know how far they went.

GSM encryption is just for the radio connection. In the greater part of the above, we focused on an assailant who listens stealthily on information as sent between the cell phone and the base station. The required radio hardware seems, by all accounts, to be accessible off-the-rack, and it is effortlessly considered that this situation is pertinent practically speaking. In any case, the SMS does not travel just from the base station to the cell phone. Its total voyage starts at the server offices, at that point experiences the Internet, and after that the supplier's system, until the point when it achieves the base station - and just by then does it get encoded with whatever A5 variation is utilized**.**

How information is secured inside the supplier's system, and between the supplier and the server which needs the SMS to be sent, is out of the extent of the GSM determinations. So anything goes. Anyway, if the aggressor is the supplier, you lose. Law implementation organizations, when they need to listen stealthily on individuals, ordinarily do as such by asking pleasantly to the suppliers, who constantly go along!

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The program machine specifications of the program where the model was implemented and tested is already described above. There are no such system requirements as of now. Except, obviously the machine should have a proper python (preferably 3.x) environment up and running;

5.2 Security and Safety Requirements

As of now, there are no such safety norms that needs to be followed. The entire program will be running on a local machine. For major scale implementation, the safety norms and certifications will be specified with time.

6. Result and Analysis

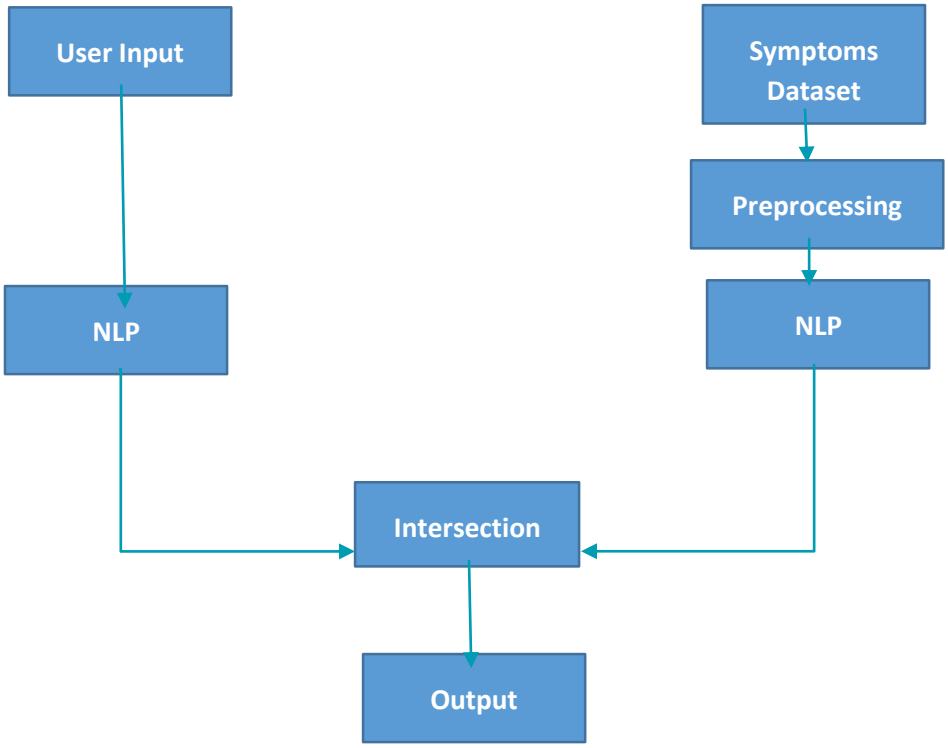
From the above usage, it is watched that the model is working moderately well if there should arise an occurrence of a wide range of manifestations identified with any illness. As said the model can be additionally enhanced with various application plans like neural net joined with the connected NLP usefulness.

Appendix A: Glossary

NLP: Natural Language Processing
GSM: Global System for Mobile
NN: Neural Network
RNN: Recurrent Neural Network
LSTM: Long Short Term Memory

Appendix B: Analysis Models

Figurative Representation of the implementation technique used:



Appendix C: To Be Determined List

1. Scope of RNN to increase the predictive nature of the model.
2. Scope of integration with SMS providing services or a general Phone UI for interaction.

CERTIFICATION

It is certified that the work contained in the project report titles “Medical Chatbot” by Yogita Bala Singh, Gaurav Sigh, Zeeshan Ali Naqvi, Siba Prasad Tripathy and Sayan Chakraborty has been carried out under my/our supervision and that this work has not been submitted elsewhere for a degree.

Signature of Supervisor(s)

_____

Name:
Bhabani P Mishra

Department:
Department of Computer Science and Engineering

KIIT Bhubaneswar

April, 2018

DECLARATION

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all the principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Signatures:

1. _____
2. _____
3. _____
4. _____
5. _____

Dated: 30/04/2018