International Journal of Electrical and Computer Engineering (IJECE)
Vol. 9, No. 4, August 2019, pp. 2585~2592
ISSN: 2088-8708, DOI: 10.11591/ijece.v9i4.pp2585-2592

Heart rate encapsulation and response tool using sentiment analysis

Nandyala Srujana Reddy[1], Prabadevi B.[2], Deepa B[3]
[1,2]School of Information Technology and Engineering, Vellore Institute of Technology, India
[3]Department of Information Technology, Sri Sairam Institute of Technology, India

Article Info

Article history:

Received Jun 27, 2017
Revised Mar 4, 2019
Accepted Mar 6, 2019

Keywords:

Comments
Feedback
Healthcare data
NLP
NLTK
Sentiment analysis

ABSTRACT

Users of every system expect it to get better. Providing feedback to the owners of the system was difficult but with the advent of technology, it has become handy. Users can now post their comments through online blogs, android apps and websites. Due to the enormous data piling up every second, it has become a problem in analyzing it. In this paper, sentiment analysis is used for analyzing comments and reviews posted by users. The experiments are done with dynamic and real data. The tools, algorithms and methodology that could fetch accurate results are described. Experimental results indicate 90% of accuracy in proposed system. The review report generated would help the hospital management to identify the positive and negative feedback which further assists them in improving their facilities that could not only create customer satisfaction but also enhanced business processes.

Corresponding Author:

Prabadevi B.,
School of Information Technology and Engineering,
VIT University,
Vellore, Tamil Nadu, India.
Email: prabadevi.b@vit.ac.in

1. INTRODUCTION

Healthcare industries like hospitals, pharmacies, laboratories, software solutions are growing tremendously which is leading to exponential growth of data. Continuous advancement of all these facilities is necessary as they deal with health of the human. The zones of enhancement are identified through observation, experience and feedback of the users. The word feedback refers to the reaction to a product that would act as a main ingredient in improvement. The technical boom has let the users deliver their feedback at any point in time. Hospitals consider this as an important parameter in providing care [1].

Upsurge in the patients directed to diverse views and insights with respect to clinical amenities. These are carried to the infirmary through android app submissions, mails and websites [2]. Survey reveals that 85% of individuals use websites and blogs to post their comments [3]. There is a wide angle to analyze but it has become difficult as the data is unstructured. Owing to this, an instant action cannot be applied to address the issue and correct the condition. This would result in loss of trust among the users. Manual scrutiny might draw precise results but would require profuse manpower and time. Since health information is sensitive, misusing it could cause drastic effects. Hence the associated data is to be collected in the utmost efficient way that would else result in improper data. The feedback submitted is an expressive statement of the user which aids as a grade sheet for the hospital. The usage of the words is diverse in numerous cases for which the algorithms are intended with many restraints like tense, context, substitutes, adjacent words. Some feedback is sensitive and hence sent through emails. The data is encapsulated so it is not exploited. The response tool proposed in this paper is built to expand the healthcare commercially [4], aesthetically and to increase user satisfaction [5].

The feedback passed via internet is available and can be accessed in snapshot. A survey suggested that 25% of people read the feedback posted and make their decisions [6] which states the importance of performing sentiment analysis on the huge data available. It is done for free of cost at any time by using the technologies accessible [7]. It involves two stages where the data is initially collected and then analyzed using the algorithms.

- Data collection: Since the data emanates from different sources and various forms, it is to be captured capably. Retrieving feedback from android applications, websites and mails is demonstrated.
- Sentiment Analysis: The information collected is directed to sentiment module which provides positive and negative feedback statement. The algorithms are explained and results obtained after implementation are demonstrated.

The following machine learning based algorithms pave a way for predictive analysis. This would aid the organization in making precise choices at right time by estimating risks and assessing them.

2. PROPOSED METHODOLOGY AND APPLICATION OF SENTIMENT ANALYSIS

2.1. Data collection from Android apps

Stage 1 identifying the data that is required for analysis: Consider an android application used by the users to provide feedback. This app might contain the fields for every feature like hospitality, ambience, treatment, cost. There are few apps where the data is posted in a single free text comment field which requires special attention. The data that could add value to the analysis is discussed with the management and considered. The data collection flowchart is depicted in Figure 1.

Stage 2 connecting the android application to the MySQL database: The data is to be collected from the android applications and stored in database. Asynctask is imported in the Android application to pass the data through internet. Http libraries are imported to connect the android app to the database. The data/comments are sent to the server using the AsyncTask package. Any data posted through the app would be updated dynamically.

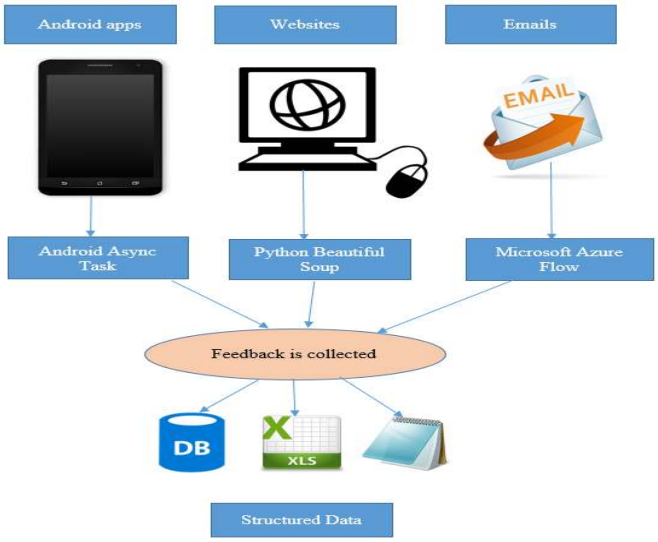

Figure 1. Data collection Flow chart

2.2. Web scraping to retrieve data

Stage 1 identify the websites to be scraped: There are several websites available where the reviews are posted. The ones that are often used and viewed by the users are identified using the count of views running in the background. It could also be done using the Google API's.

Stage 2 scrape the identified webpage: Webpages are coded with numerous tags where a specific field holds the comment section. The admin of the webpage can retrieve the comments from the database. As an analyst, one could scrape the data by inspecting the page source. The comment field can be identified by inspecting the element. BeautifulSoup in Python is used to parse the tags and scrape the data and is then stored into a text file.

2.3. Collect feedback sent through mails

Many organizations have specific mails where feedback can be posted. Most of the users do this as it is official. Analyzing every mail is a tedious process. Hence the data is collected and stored in excel where it can be converted to text file. This process is done using the Microsoft Azure cloud and Office 365 Flow technology.

2.4. Sentiment Analysis

The steps involved in sentiment analysis is depicted in Figure 2.

Stage 1 convert words to tokens: Each word is a token when a sentence is "tokenized" into words [8]. Each sentence can also be a token, if the paragraph is tokenized into sentences. Sent tokenize and word_tokenize from nltk.tokenize are to be imported. The text is passed as a parameter into one of these functions where words are returned as tokens. The split data is further processed using disparate algorithms.

Stage 2 stemming words: Sarcastically used words are stored in a list where the real data can be compared against. If the data has such words, they are removed as they don't carry any meaning and are not useful for sentiment analysis. Stemming [9] is a kind of normalization where the words that hold the same meaning but differed by tense are identified and segregated. This would remove unnecessary and redundant data.

Stage 3 parts of speech tagging: Tagging the words with parts of speech helps in capturing the emotions of users and its' intensity [10]. The words present in corpora are utilized to compare and segregate. The nouns and adjectives are assigned high score.

Stage 4 chunking and chinking data: The main goal of chunking is to group tokens into "noun phrases." It is done by combining the part of speech tags with regular expressions [11]. After a lot of chunking, there would still be some words in chunk that are ambiguous. Chinking helps in removing unchunked data from chunks.

Stage 5 lemmatizing the data: Lemmatizing is similar to stemming, but stemming can often create non-existent words, whereas lemmas are actual words. It takes part of speech as parameter and pulls the result. WordNetLemmatizer is imported from nltk.stem. The text is passed as a parameter into WordNetLemmatizer.lemmatize (). Resultant words are carried over for creating sentiment module.

Stage 6 corpora matching: The NLTK corpus is a huge heap of all kinds of natural language data sets. The corpus might be text files or XML data. It helps in hitting the synonyms and providing relevancy in the analysis. Gutenberg from nltk-corpus is imported. Gutenberg-Bible is a text file within Python library. This corpus helps in identifying the relevant words in the real world for analyzing the data. The data is passed and compared with corpus which helps in identification of useful words.

Stage 7 naïve Bayes Classification: It is a supervised rule mining where we feed the machine stating which is positive and negative. The data is featured into training data and run on naïve bayes classification. This provides the percentage of positive and negative data. The data for which you want to train and test the classifier is firstly split. Training is done through classifiers by passing the training set into nltk.NaiveBayesClassifier.train ().

Stage 8 Scikit-Learning: Words are processed to tokens and converted to features. Scikit-learn [12] is a package of various classifiers like Support Vector Machin (SVM) and Gaussian. The algorithm is applied to retrieve the fuzzy data. The data folder is passed as a parameter with pos and neg sub-folders where the positive and negative referenced data are present.

Stage 9 Creation of Sentiment Module: Since the data to be analyzed is collected and pre-processed, sentiment module is created which it is collectively loaded for analysis. The data fed is split into words/tokens and Parts of speech tagging is done. Positive and negative word libraries are opened and assigned to a variable that is further referred. Classification of data is done using the pickled algorithms. featuresets.pickle () and Bernoulli_classifier.pickle () are imported where the algorithms are fed. Open the pickled algorithms saved in documents.pickle. NLTK provides a functionality of classifying the data as per the pickle. The dataset is shuffled and split to training data and testing data that are passed to the classifiers for analysis. Nltkclassify.accuracy (classifier,testingset*100) gives the accuracy percent.

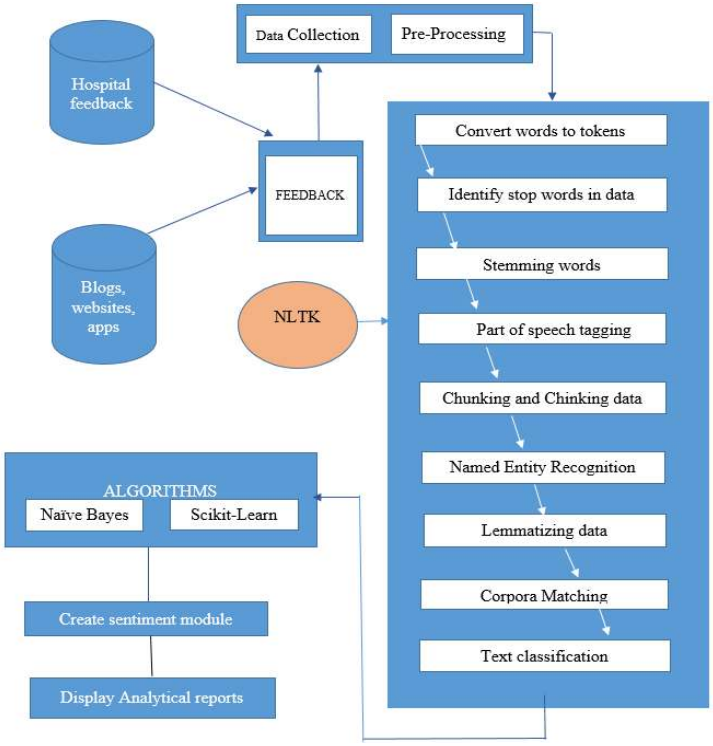

Figure 2. Sentiment analysis flowchart

2.5. Algorithm

Input: Data from Android applications, blogs and websites related to healthcare industries. Output: Sentiment Index of the reviews which denotes the positive and negative impression of the users/customers.

1. Compile the data from MySQL DB and test files into a single free text file.
2. Pre-Processing:
   a. Convert the words to tokens using NLTK library
   b. Identify stop words and clear them.
3. Tag the words with Parts of Speech (Noun and Adjective)
4. Match the words with the corpora available in the NLTK package.
5. For each word in the data set(words(i))

```
{
    If(if(words(i)) == positive and words(i-1)! = not/no)
        {
            PositiveWordCount++
        }
    Else
        {
            NegativeWordCount++
}
```

6. Store the PositiveWordCount and NegativeWordCount of the words to train the sentiment Module to determin SentimentIndex

3. EXPERIMENT

a. Post data through Android Application

An android application is created with the fields required and hosted locally. A survey has been conducted on the reviews and comments submitted by the users. The data is fetched into the MySQL Database dynamically. The snapshot of submitting comments through android app, storing data in database, web scraping comments from internet contents, output of comments after web scraping, identifying words and features of data and sentimental index obtained by applying sentimental analysis are depicted in Figure 3 through Figure 7.

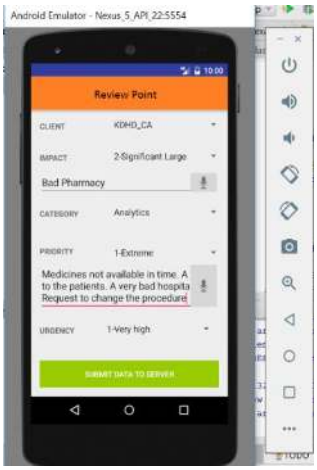

Figure 3. Submitting comments through android application

b. Store comments to database

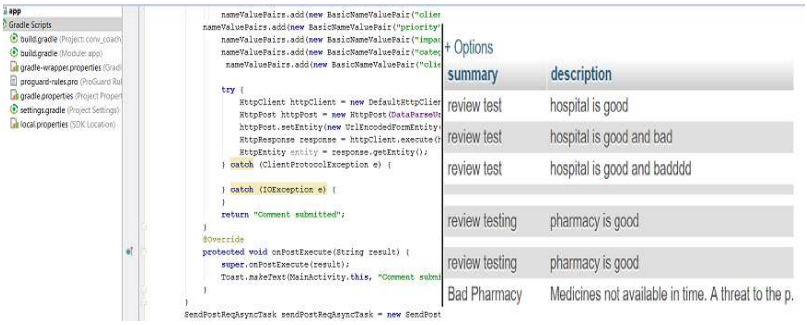

Figure 4. The submitted comments stored in database

c. Web scraping comments from websites and blogs

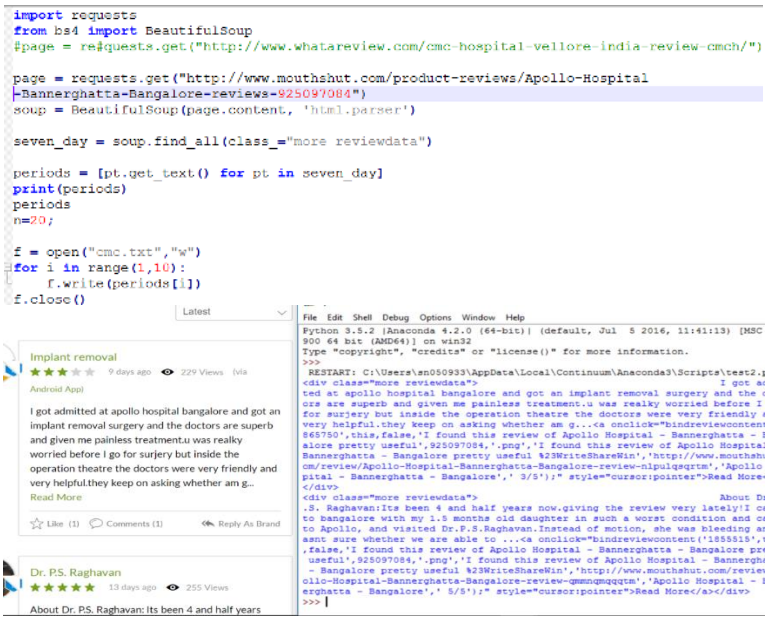

Figure 5. Output of comments after web-scraping

d. Finding words and features

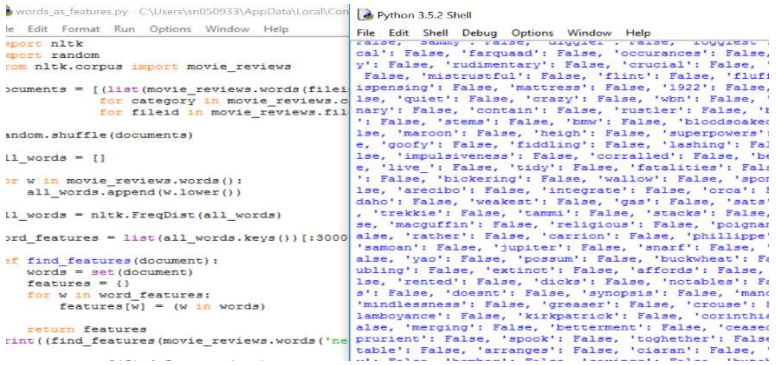

Figure 6. Identifying words and features of the data

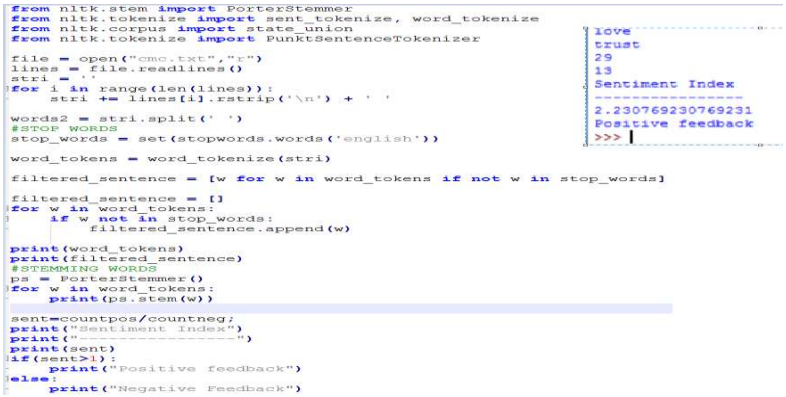

Figure 7. Sentiment Index obtained after performing sentiment analysis

4. RESULTS AND ANALYSIS

The data collected into destination is loaded to a text file and processed for sentiment analysis. Since the data is unstructured, the data is processed using the above stated algorithms. The data collection is done automatically using the above processes. The data source and its destination is given in Table 1.

Android Studio API 21 is used to push the comments into database dynamically. Page source of desired website is scraped and written to text files. Flow is processed by obtaining a token ID for Microsoft Azure and written to Excel Files which results in structured data. The data collected is stripped of unnecessary data using pre-processing techniques. The emails are retrieved as Power BI set which can further be used for predictive analysis. This could also be sent for text analytics in Flow (Office 365 Power apps).

Table 1. Data source and corresponding Destination

Data Source	Destination
Android apps	MySQL Database
Websites	Text files
Emails	Excel
Structured Data is produced	
Combined and stored in a single text file	

Time taken to collect data from each source manually and by using machine based algorithms is recorded and plotted in minutes as shown in Figure 8. Pre-processing the data is implemented in Python using NLTK library. A sentiment module was initially built into which the data collected is fed. Figure 9 shows the time taken to perform sentiment analysis on each set of data, manually and by using machine based algorithms.

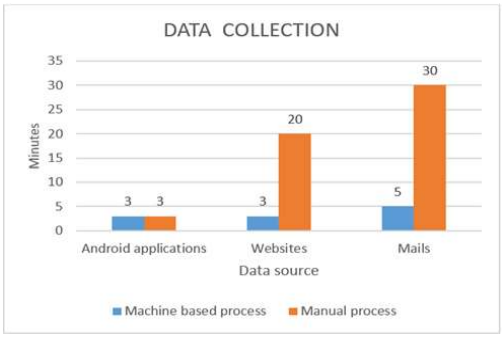

Figure 8. Machine based vs manual process for Data Collection

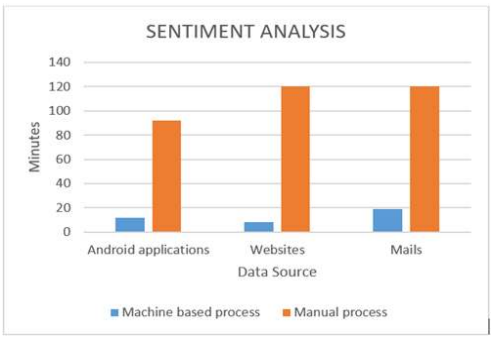

Figure 9. Machine based vs manual process for Sentiment analysis

Sentiment analysis generates a report with positive and negative aspects of the feedback. The rankings for the structured review is given as: excellent-5, good-4, average-3, poor-4, worst-5. The sentiment is calculated as per the number of reviews and rankings [12].

A survey was conducted in few hospitals where the patients were asked to post their feedback either through websites, mails or apps. The feedback was collected which reported about 94% accuracy when compare against manual and machine based analysis. The resultant positive and negative words are given in Table 2.

Hospital review data set collected and movie dataset present within the NLTK module are experimented and compared. Survey [13] suggests that hospital management is ready to consider machine-based sentiment analysis as a factor to improve their facilities in healthcare. In fact, they have come up with more uses that sentiment analysis could provide. It is stated that positive and negative words categorized for every domain could help them recognize the areas of pros and cons [14]. The words are collected and stored as positive and negative words [15] as per the usage of people all over. They also include commonly used phrases in social media which are considered as reference while analysing the data.

Table 2. Resultant positive and negative words

Patient Feedback	Positive	Negative
Regarding hospitality of the hospital management	clear	misconception
	work	frustration
	good	handicapped
	beloved	rude
	great	unkind
	well	horrible
	best	disgusting
		pathetic
Cost and value of the treatment	worth	bad
	best	fraud
	sweet	
Environment	clear	vulnerable
	beauty	pathetic
	conscious	foul
	noisy	

Manual analysis involves finding of positive and negative words in the reviews. The same perception is implemented in Python that resulted in 85% accuracy. Inaccuracy is caused due to the usage and context. Hence, the algorithms and classifiers are implemented to sense the usage and emotions by rating the adjectives which led to more accurate results. It has been tested for many data sets. Accuracy percentage is checked for every classifier which reported a similar accuracy as depicted in Figure 10. The data collected from the sources is passed through sentiment module and implemented in Python. The various classifiers discussed above are used in sentiment analysis.

NLTK [3] is a large family where any analytical algorithms could be built effectively. When data is sent through them, the accuracy percent differs for every dataset. Experimental results on two record set provided 65-70% accuracy. This assures that accuracy is more than 50% which can act as a major business-decision making statement.

The comments were collected from the websites, apps and mails and stored in text file. The given graphs measure the accuracy of manual analysis vs machine based sentiment analysis. From Figure 11,

the accuracy scale denotes the impact of machine based algorithms that is being done in no time. The results suggest a mechanism to address the flowing data especially feedback which is considered an asset of upgrading business. This can be advanced to replace traditional analysis. The figure shows the accuracy of sentiment analysis employed on the data sets acquired.

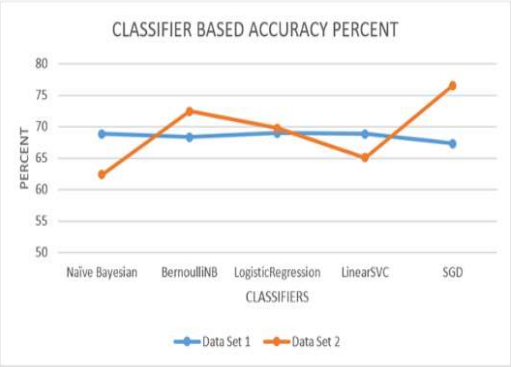

Figure 10. Classifier Based accuracy

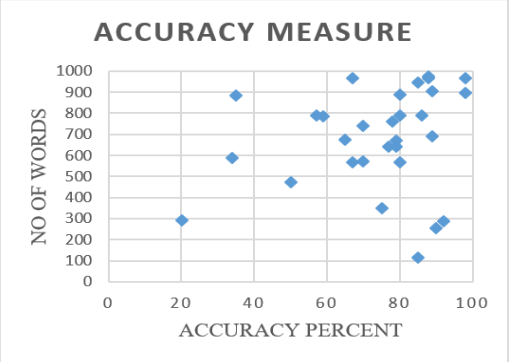

Figure 11. Accuracy measure of sentiment analysis

5. CONCLUSION

Healthcare is the field that has gained attention all over the world as it is important for survival. Since the feedback is provided in digital form, there is an urgent need in an automated analysis. This paper demonstrated the key processes of Data collection and Sentiment analysis. This plays an immense role in stream-lining the business processes. Feedback is collected from online websites, apps and mails and tested in real-environment. Sentiment analysis is done on the live-data procured and reported. All the algorithms have been implemented in Python and accurate results have been generated. Graphical representation showing variations has been provided for a wide-angle view.

In future, data collection would be done dynamically from websites where a timer would be set for every new comment. A service based schedule would be developed for the same. Sentiment analysis would be then done on every single review and updated. Also, an extreme optimization is being planned to increase the accuracy of the sentiment analysis.

An automatic alert would be sent to the hospital if there is an increased negativity for a feature. This is currently in development which uses MSG 91 as API to send automatic messages. This could help in addressing the issue at the earliest. This helps the hospital to concentrate on any domain in a right way.

REFERENCES

[1] B. Steven, et al., "Natural Language Processing with Python," pp. 16, 27, 79, 2009.
[2] Gao G. G., et al., "A changing landscape of physician quality reporting: analysis of patients' online ratings of their physicians over a period," 2012.
[3] Greaves F. and Millett C., "Consistently increasing numbers of online ratings of healthcare in England," *J Med Internet Res*, vol/issue: 14(3), pp. e94, 2012.
[4] P. Yerpude, et al., *International Journal on Natural Language Computing (IJNLC)*, vol/issue: 4(4), 2015.
[5] Greaves F., et al., "Associations between Internet-based patient ratings and conventional surveys of patient experience in the English NHS: an observational study," *BMJ Qual Saf*., vol/issue: 21(7), pp. 600-5, 2012.
[6] Fox S., "The Social Life of Health Information," Washington, DC: Pew Research Center, 2011.
[7] Kumar P. K. and Nandagopalan S, *International Journal of Electrical and Computer Engineering (IJECE)*, vol/issue: 7(5), pp. 2818-2822, 2017.
[8] M. Christopher D. and S. Hinrich, "Foundations of Statistical Natural Language Processing," Ch 4, pp. 575, 2003.
[9] P. Arora, et al., *International Journal of Electrical and Computer Engineering (IJECE)*, vol/issue: 7(2), pp. 967-974, 2017.
[10] R. Mário and T. António, "Advanced Applications of Natural Language Processing for Performing," Ch 1, 2, 4.
[11] F. Jeffrey E. F., "Mastering Regular Expressions," 2006.
[12] Scikit-learn. http://scikit-learn.org/stable/
[13] X. Fang and J. Zhan, *Journal of Big Data*, vol. 2, pp. 5, 2015.
[14] L. R. Nair, et al., *International Journal of Electrical and Computer Engineering (IJECE)*, vol/issue: 7(1), pp. 402-407, 2017.
[15] Liu B., "Sentiment Analysis and Opinion Mining. Synthesis Lectures on Human Language Technologies," 2012.