Weighted summarization of Student Feedback using Sentiment Analysis

Sneha
Department of Information and Communication Technology
Manipal Institute of Technology
Manipal University, Manipal, India

B. Akshatha Bhat
Department of Information and Communication Technology
Manipal Institute of Technology
Manipal University, Manipal, India

Preetham Kumar, Ph.D
HOD, Department of Information and Communication Technology
Manipal Institute of Technology
Manipal University, Manipal, India

ABSTRACT

Every year massive amount of feedback is gathered from students regarding subjects and its respective faculty. The amount of time to analyze this data manually is a very tedious and time consuming. This is where the summarization feature comes into picture. It extracts important information found in every feedback document. Automatic summarization based on word frequency statistics takes comments and weights them to produce word frequency and then sentence frequency. Also, the sentiment information in these documents belongs to a wide spectrum ranging from positive to negative. SentiWordNet assigns sentiment numerical scores: positive or negative. Thus, providing clues for sentiment analysis. The spell-checker helps to rectify the incorrect words for proper implementation of those two concepts.

General Terms:

Edit-distance algorithm, SentiWordNet

Keywords:

Spell-checker, Sentiment Analysis, Text summarization

1. INTRODUCTION

With the increasing popularity of Internet and the diversity of information obtaining technologies, the amount of quickly growing information has gone beyond our imagination. People post comments which can be redundant or uninformative and that the sheer quantity of comments will quickly grow to an unmanageable size. Many techniques are present to help users to find the desired information from large data set quickly and accurately and automatic summarization is an effective approach to this problem statement. Automatic summarization is the process of reducing a text document with a computer program in order to create a summary that retains the most important points of the original document. As the problem of information overload has grown and as the quantity of data has increased, so has interest in automatic summarization. An example of the use of summarization technology is search engines such as Google. Document summarization is another.

At the same time, it is very common today for websites to allow readers to comment on articles posted on the Internet. These comment threads usually contain interesting opinions and give a good indication of the average reader's sentiment. The rise of social media such as blogs and social networks has fuelled interest in sentiment analysis. With the proliferation of reviews, ratings, recommendations and other forms of on-line expression, on-line opinion has turned into a kind of virtual currency for businesses looking to market their products, identify new opportunities and manage their reputations. As businesses look to automate the process of filtering out the noise, understanding the conversations, identifying the relevant content and auctioning it appropriately, many are now looking to the field of sentiment analysis.

The combination of automatic summarization and Sentiment analysis is one of the best as it combines the advantages of both the approaches yielding results highlighting only the important points, thus making it more compressed and informative at the same time.

Likewise, at the end of every semester, Manipal Institute of Technology collects "On-line Student Feedback" from the students and faculty regarding different subjects and its concerned lecturers. This data will be used throughout our research work.

2. LITERATURE SURVEY

2.1 Introduction

A review of existing literature was performed to support the study undertaken in this research. Quite a few papers have been referred to identify the problem statement. Also, the techniques adopted

for this research have been evaluated thoroughly to make sure the research's expected outcome matches the actual outcome.

Three important modules have been combined to make this application a success. The ideology behind every module has been briefly described below.

2.2 Sources for Spell-checker

The spell-checker is an integral part of the application. Now-a-days, with arising technologies like auto-correct and spelling suggestion, people are bound to make spelling mistakes. This originated idea of implementation of a spell-checker.

With the help of on-line sources like blogs and other informative articles [5], a spell-checker was put together for this application.

The module basically makes use of the concept of edit distance, one of the most sought-after algorithms in the making of spell-checker. E.g., Levenshtein distance [1] between "Hello" and "Hallo" is 1, as we just need to substitute 'a' in place of 'e'.

2.3 Sources for Sentiment Analysis

Another important module of this application is Sentiment Analysis. This topic is very interesting and is one of the hottest research areas in computer science. With the amount of information that is shared on social media, forums, blogs, etc, it is easy to see why we need to automate sentiment analysis: there is simply too much information to manually process. This is exactly why the module has been included as well. The information collected from "On-line Student Feedback" is huge in size.

An On-line journal titled "Techniques and Applications for Sentiment Analysis" [2] was referred for this module. It has briefly explained the concept of Sentiment Analysis and its different approaches.

Along with the journal, a paper titled "Sentiment Analysis-How to derive Prior polarities from SentiWordNet" [3] gave more insight to one of the approaches to Sentiment Analysis. It basically explains how Sentiment lexicon acquisition approach can be implemented using a sentiment dictionary called SentiWordNet. Also, certain blogs provided more information as to how to proceed further with the implementation of the module.

2.4 Sources for Text Summarization

Text/Automatic Summarization is the last module in this application. The increasing availability of online information has necessitated intensive research in the area of automatic text summarization within the NLP community. This field is observed as a major future research area with the exponential growth of information.

There are quite a few approaches to text summarization. One of the approaches to text summarization is weighted summarization that comes under extraction based summarization. It has been described in a paper entitled "Weighted summarization of music comments" [4] whose mentioned methodology has been applied in the application.

3. PROBLEM DEFINITION

The problem statement is stated as follows: At the end of every semester, thousands of students of the Institute contribute to "On-line Student feedback" by rating and commenting on every subject and its lecturer. The lecturers also take part in the event. The feedback thus consists of information from thousands of students which is nothing less than huge, important and redundant. Later, the information collected has to be evaluated. The results yielded by the evaluation process will be used for lecturer and subject evaluation. Manual evaluation is a very tedious and time-consuming task and also data to be evaluated is highly confidential. Hence the task of evaluation cannot be handled by any employee of the institute.

This paved the way for the development of "Weighted Summarization of Student feedback using Sentiment Analysis" application.

The proposed application will have three important modules. They include:

(1) Spell-checker: The proposed approach [5] is a perfectly naive approach to create a spell-checker. The accuracy of the spell-checker can be ranged between 80-90%. Advanced spell and grammar checkers use the concept of language models to find the probability that a sentence is correct.
(2) Sentiment Analysis: In this module, the approach used is Sentiment Lexicon acquisition [2]. It makes use of a generalised version of SentiWordNet with application specific words.
(3) Text Summarization: This module will be implemented using one of extractive based summarization approaches i.e. Weighted based summarization [4].

4. DESIGN

Design is one of the most important phases of research and is a graphical representation of a working system. This is the phase of system designing. It is the most crucial phase in the development of a system. The logical system design arrived at as a result of system analysis and is converted into physical system design.

Pertaining to this application, four diagrams have been designed. They include:

4.1 Use case diagram

In Fig. 1, at first, user interacting with the application uploads an input file. The accepted file goes through a series of steps i.e. checking of spelling, checking of polarity, writing into positive and negative comments only files that are used as input to summarization section leading to the last step which is display of output generated by the application that will be visible to the user.

4.2 Activity diagram

In Fig. 2, the correct flow of activities has been mentioned which involves uploading of input file, checking the spelling, checking the polarity, text summarization and finally display of output.

4.3 Sequence diagram

The Fig. 3 is sequence diagram of the system. The messages sent by the user and interactions between the application and the system

for performing different operations using the content of uploaded input file have been clearly shown.

4.4 Class diagram

In diagram Fig.4, the dependencies of classes on one another and their relations along with the important variables and methods used is shown. GUI calls ReadFile class which then makes a call to Check Spelling, Extract Sentiment and Summarization classes in the given order. Also, Check Spelling depends on EditDistance class and Summarization depends on Positive Feedback and Negative Feedback.

5. IMPLEMENTATION

5.1 Spell-Checker

The spell-checker module has been implemented with an aim to eliminate spelling errors to a greater extent. One of the most sought-after algorithms i.e. edit-distance algorithm is the most important part of this module. Also called as Levenshtein distance is a string metric for measuring the difference between two sequences. Levenshtein distance between two words is the minimum number of single-character edits (insertion, deletion, substitution) required to change one word into the other.

e.g., Levenshtein distance between ”kitten” and ”sitting” is 3, since the following three edits change one into the other, and there is no way to do it with fewer than three edits:

—kitten→ sitten (substitution of ”s” for ”k”)

—sitten→ sittin (substitution of ”i” for ”e”)

—sittin→ sitting (insertion of ”g” at the end)

The algorithm [1] uses dynamic programming. It solves problem by combining solutions from sub-problems. It uses bottom up approach. We compute D(i,j) based on the previously computed smaller value i.e. compute D(i,j) for all i$(0 < i < n)$ and j$(0 < j < m)$. Minimum of insertion, deletion and substitution is [6]

$$D(i,j) = \text{Min} \begin{cases} D(i-1,j)+1 & \text{- deletion} \\ D(i,j-1)+1 & \text{- insertion} \\ D(i-1,j-1) & +1; \text{if} x(i) \neq y(j) \\ & +0; \text{if} x(i) = y(j) \\ & \text{-substitution} \end{cases}$$

5.2 Sentiment Analysis

Sentiment analysis is the process of identifying people’s attitudes and emotional states from language.

The approach used in this application is Sentiment Lexicon acquisition [2]. The method makes use of a generalized version of SentiWordNet with application specific words which is a dictionary containing sentiment scores for words. Since this is a generalized version, no part-of-speech taggers were used to explicitly classify words based on their part of speech. Using SentiWordNet, the overall score is computed by extracting the sentiment score of that word from the dictionary with all possible part-of-speech values.

5.3 Text Summarization

Text Summarization can be defined as the process of identifying novel information from a collection of texts. Metaphorically, text mining is the process of mining precious nuggets of ore from a mountain of otherwise rock.

The module in this application is developed using extraction based summarization i.e. Weighted based summarization. The weight represents strength of each keyword. The algorithm used to develop this module has been briefly mentioned in [4].

6. RESULT ANALYSIS

6.1 Spell-checker

The accuracy of spell-checker is medium. It could identify incorrect words like ”excelent” with correct word ”excellent”. But it failed to identify words like ”doze”,”d” which is actually the text message lingo for ”does” and ”the” respectively. So, for such words we return the original word itself.

6.2 Sentiment Analysis

Since the data is a generalized version we have modified the SentiWordNet dictionary as per our application needs. The words like ”not”, ”fun”, ”loves”, ”jocular”, ”lacks”, ”dictates”, ”threatens” etc. which were originally not present in the SentiWordNet were added by us. It helped to better sentiment classification. But still there are limitations to this module. The words like ”not bad” is actually a positive phrase but it had a negative score.

6.3 Text Summarization

Frequently occurring words increase the probability of a sentence to be above threshold value. This helps us to obtain most frequently commented thoughts.

We have analysed the text summarization module with and without *stop words*. Without *stop words* removal, the value of maximum frequency was found to be 60.0 for the word ”the”. Whereas with *stop words* removal, the value of maximum frequency was 28.0 for the word ”teacher”. Hence, removal of stop words reduces the size of candidate keywords and hence prevents distortion of maximum frequency.

6.4 Final analysis

There are few spelling mistakes and most of them are rectified. Also, the sentiment score has accuracy as expected. We have analysed three types of files: positive, negative and neutral. The sentence frequency generally ranges from 0.125 to 3.0. In general, a properly framed sentence is above 1.0 and small phrases fall below 1.0. Thus we have adjusted the threshold value to 1.0. So the accuracy of the application is slightly above medium. Hence there is scope of improvement.

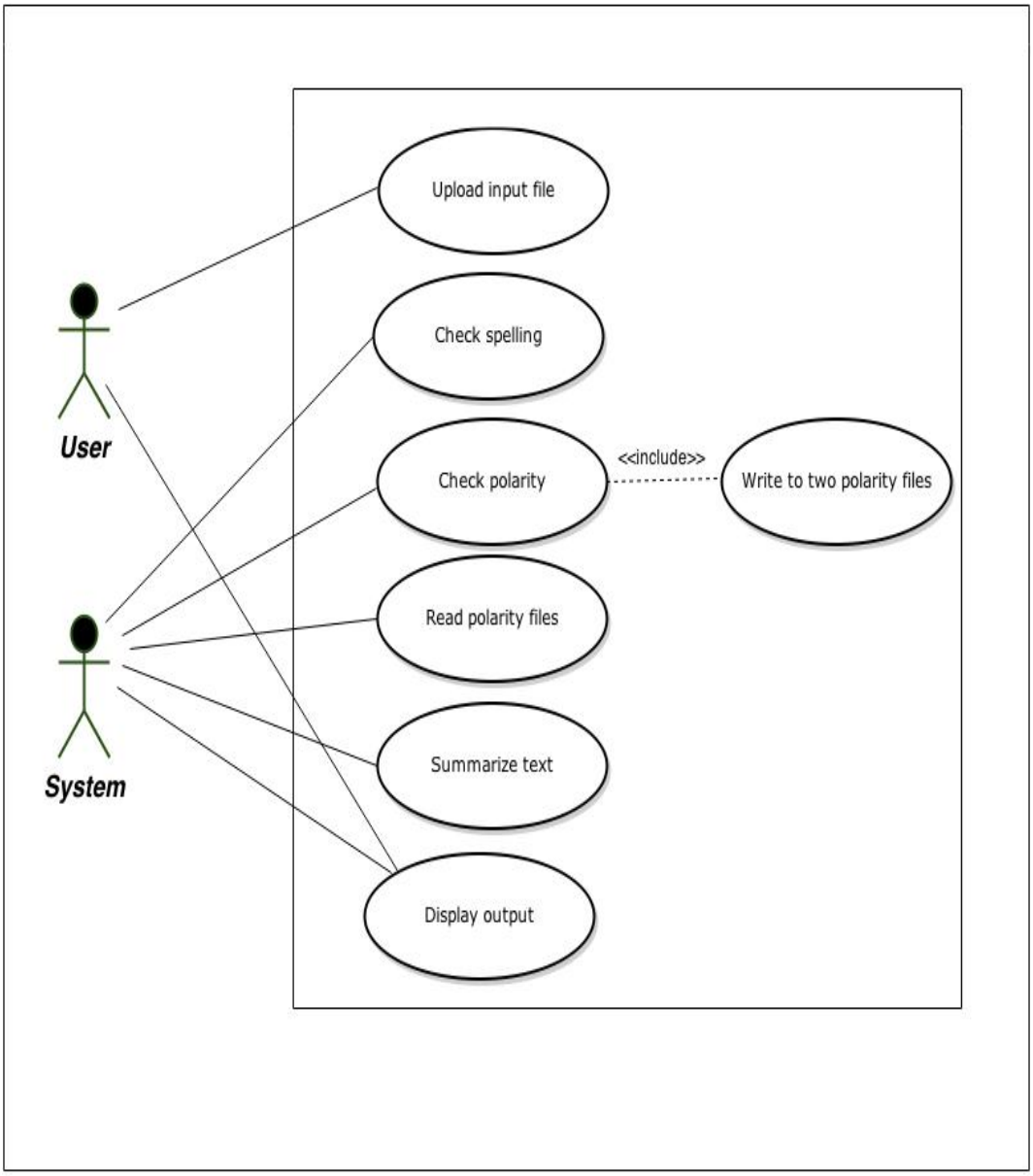

Fig. 1. Use case diagram

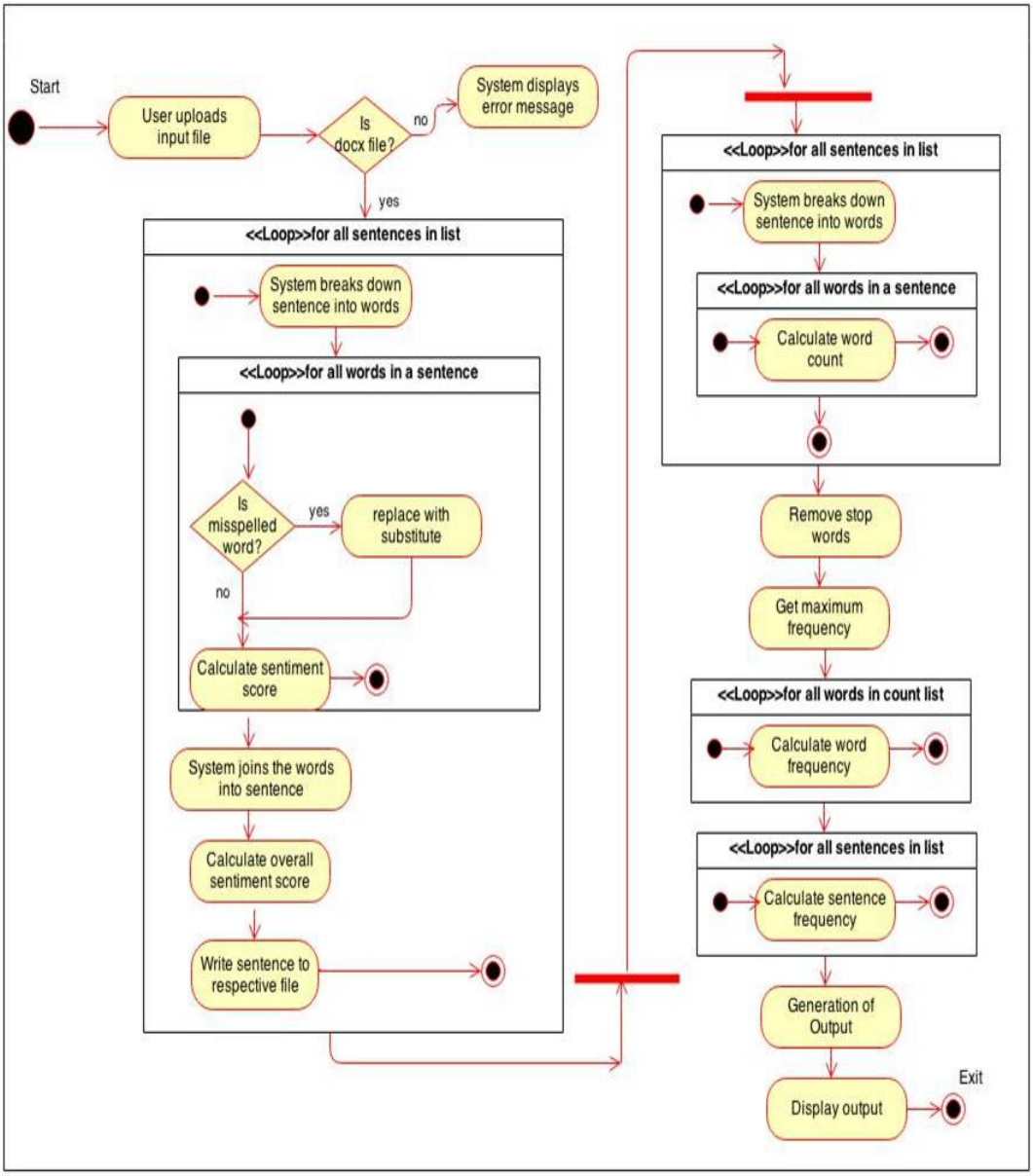

Fig. 2. Activity diagram

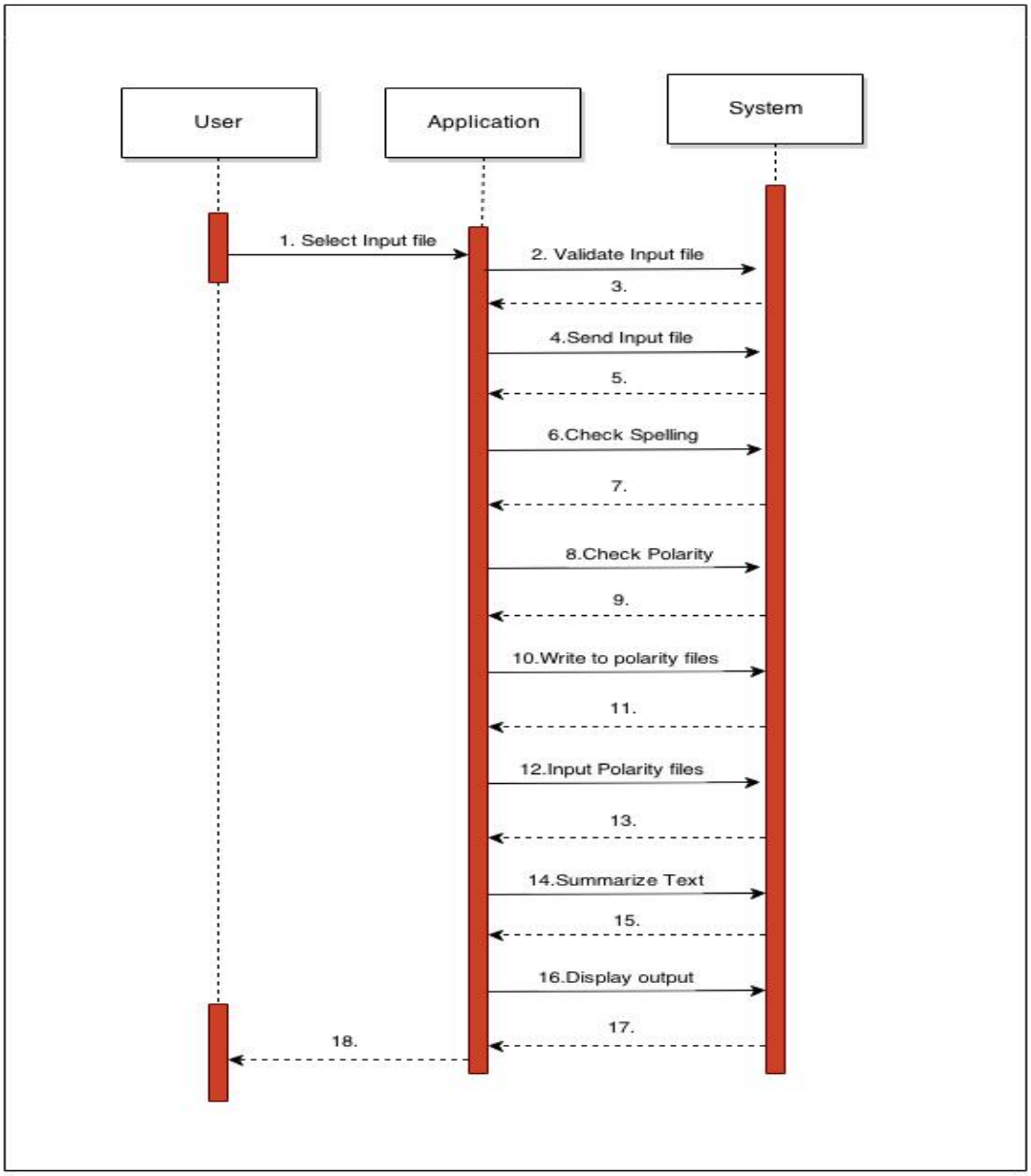

Fig. 3. Sequence diagram

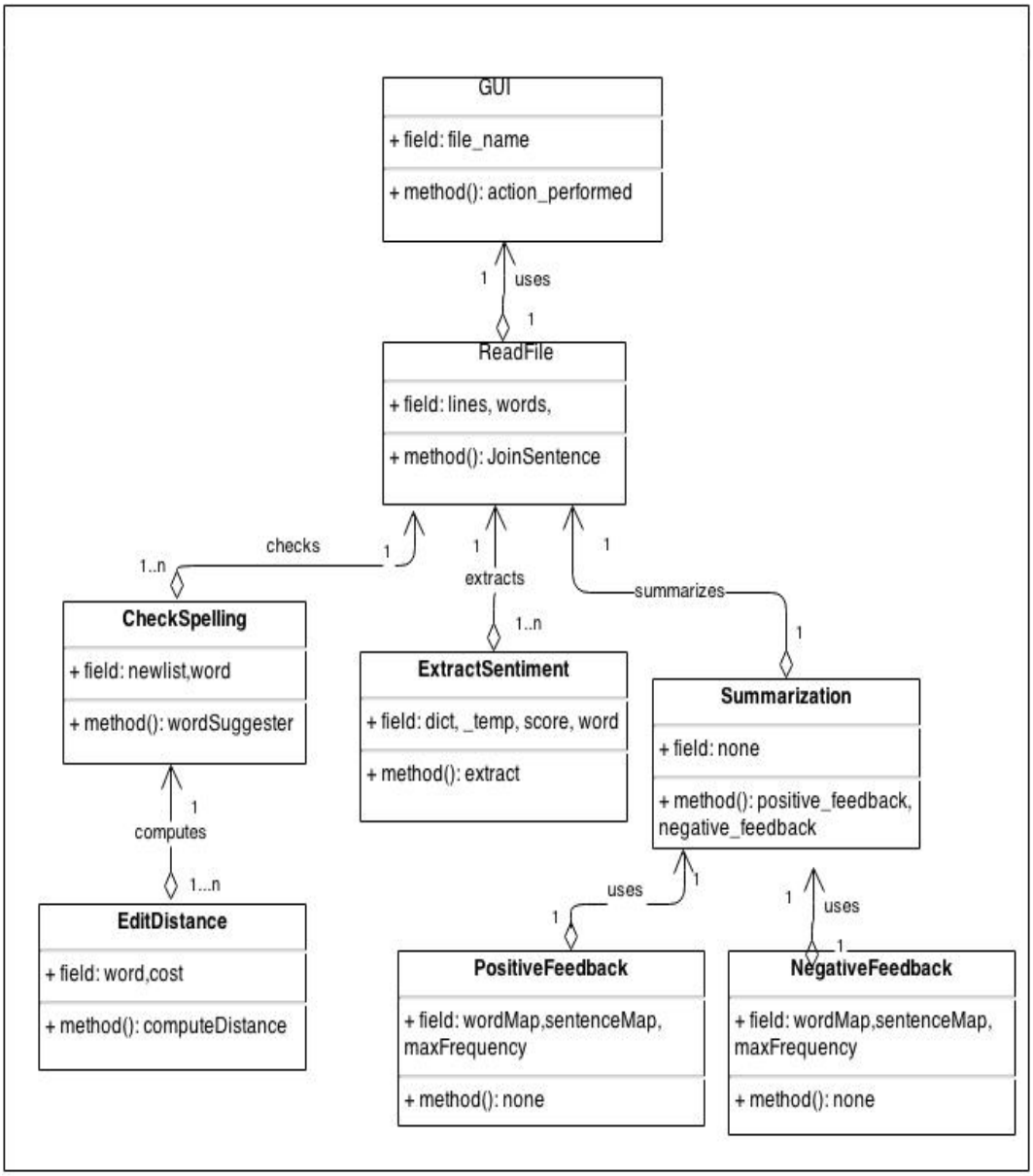

Fig. 4. Class diagram

7. CONCLUSION

At the end of our research, we were able to read a docx file and spell check it at an acceptable level. Also sentiment analysis was performed through this research. Lastly, the summary of the input file was obtained. Hence, the objective of the research has been successfully accomplished.

8. FUTURE WORK

Future work includes:

(1) Semantic analysis: This technique belongs to Natural Language processing. It can be applied to find the semantics between two dissimilar sentences and hence reduce redundancy.

(2) Sentiment analysis of sarcastic sentences: Many a times, it happens that people express their opinions with sarcasm which is quite tricky to be understood very easily. This is another challenging task.

9. REFERENCES

[1] Levenshtein distance. `http://en.wikipedia.org/wiki/Levenshtein_distance`.

[2] R. Feldman. Techniques and Applications for Sentiment Analysis. `http://cacm.acm.org/magazines/2013/4/162501-techniques-and-applications-for-sentiment-analysis/fulltext`, 2013.

[3] M. Guerini, L. Gatti, and M. Turchi. Sentiment analysis: How to derive prior polarities from sentiwordnet. September 2013.

[4] W. Hong, S. Jiang, H. Wang, and J. Shi. Weighted-based summarization of music comments. *The 8th International Conference on Computer Science and education(ICCSE'13)*, April 2013.

[5] Aditya Joshi. Simple Spell-Checker in JAVA. `http://bakedcircuits.wordpress.com/2013/08/10/simple-spell-checker-in-java/`, 2013.

[6] Dan Jurafsky. Minimum Edit Distance. `http://web.stanford.edu/class/cs124/lec/med.pdf`, 2014.