Live Sentiment Application

Prepared by
Yashvi Soni(16IT135)
Abhishek Vyas(16IT146)

Under the supervision of

Mr. Kamlesh Makwana

A Report Submitted to
Charotar University of Science and Technology
for Partial Fulfillment of the Requirements for the
Degree of Bachelor of Technology
in Information Technology
IT350 Software Group Project-III (6th sem)

Submitted at

DEPARTMENT OF INFORMATION TECHNOLOGY
Chandubhai S. Patel Institute of Technology
At: Changa, Dist: Anand – 388421
April 2019

CHARUSAT
CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY

CERTIFICATE

This is to certify that the report entitled "**Live Sentiment Application**" is a bonafied work carried out by **Miss Yashvi Soni (16IT135) and Mr. Abhishek Vyas (16IT146)** under the guidance and supervision of **Mr. Kamlesh Makwana** for the subject **Software Group Project-III (IT350)** of **6th** Semester of Bachelor of Technology in **Information Technology** at Faculty of Technology & Engineering – CHARUSAT, Gujarat.

To the best of my knowledge and belief, this work embodies the work of candidate themselves, has duly been completed, and fulfills the requirement of the ordinance relating to the B.Tech. Degree of the University and is up to the standard in respect of content, presentation and language for being referred to the examiner.

Under supervision of,

Mr. Kamlesh Makwana
Assistant Professor
Dept. of Information Technology
CSPIT, Changa, Gujarat.

Prof. Parth Shah
Head & Associate Professor
Department of Information Technology
CSPIT, Changa, Gujarat.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

This report demonstrates the production of a Live Sentiment Analysis System, with the following main objective is to build an engine adaptable to real time sentiment classification reporting. As a secondary objective, a graphical user interface is developed to enhance the interaction between the users and the system. In order to produce the software artefacts presented in the report, computer science knowledge as well as machine learning and natural language processing techniques were employed. Consequently, the concepts and techniques, which contributed to the development of the project, such as the Naïve Bayes algorithm, are explained. Furthermore, a high level view and a low level view of the system produced are detailed in subsequent chapters.

ACKNOWLEDGEMENT

We are very thankful to God to make our effort successful, then we would like to Thank our parents who gave us hope in our low times, it is our pleasure to take this opportunity to make all those who directly or indirectly helped us in our project.

We are extremely grateful to **Dr. Parth Shah**, head of department of information technology for his excellent guidance, valuable suggestion and encouragement to carry out this project.

We extend our sincere thanks to the project guide **Mr. Kamlesh Makwana** for his constructive criticism valuable suggestion and guidance to bring out the best from this project.

Finally, we would like to express our deep sense of gratitude to our classmates and many other friends of us who have supported and enriched us by sharing their ideas and storing out our doubts having discussion with us. I thank one and all.

CHAPTER:-1

INTRODUCTION

1.1 Motivation

- The emergence in the last decade of social media platforms such as Twitter, Facebook, and Instagram, enabled people to engage in social activities to express their opinions, thoughts, and emotions on a variety of topics. On such platforms, large amounts of data are produced (e.g: 6000 tweets per second), this is representing an opportunity for companies to assess their social influence and people's opinion towards their products. Consequently, a computational framework is desirable to perform opinion mining and sentiment analysis which can adapt to the activity domain of the user.

1.2 Project Summary

- In the project Live Sentiment Application, we explore the application of Natural Language Processing techniques to identify sentiment of a person through his/her tweet.
- We use python libraries to fetch live tweets from twitter and then process it to get the sentiment of the tweet.

1.3 Purpose

- The project focuses on identifying sentiment of the fetched tweet originating from twitter.
- The main purpose of this project is to serve beginner of machine learning student, a head start and better estimate that how sentiment analysis works.

1.4 Scope

- This application is targeted mainly to the audience those who want to dive-in to data science or machine learning field.
- Brands can use this data to measure the success of their products in an objective manner.

1.5 Objective

- The objective of this tool is to provide the live visualization of the sentiment of the searched word of the tweet and provide sentiment of provided line.

1.6 Technology Used

- Front-end: HTML, CSS, Javascript
- Back-end: Python Libraries like flask(for server), numpy, tweepy, textblob and many more Python Libraries
- Dataset source – Twitter
- Dataset source type - json

1.7 Drawback of Existing System:

- Lack of clean data to directly work with might have slowed down our progress.
- The loss to value of information in a real scenario for tweet analysis is very high.

CHAPTER:-2
THEORY

- The following chapter aims to clarify the techniques used throughout the project. A broad definition will be given for the core concepts involved in the development of the artefacts: Text Mining, Natural Language Processing and Machine Learning. Furthermore, specialized terminology will be explained.

2.1 Text Mining

- Text mining refers to the analysis of data contained in natural language text, (e.g. messages retrieved from twitter).
- It can be defined as the practice of extracting meaningful knowledge from unstructured text sources.
- The application domain of text mining varies from biomedical applications to marketing applications and sentiment analysis.
- In marketing, text mining is relevant to the analysis of the customer relationship management.
- This way a company can improve their predictive analytics models for customer turnover (keep track of customer opinions).
- The main goal of text mining is to process data into a structured format ready for analysis, via application of natural language processing and other analytical methods.
- Albeit, there are many aspects within the field of study of text mining, information extraction (IE) is relevant for this project.
- Consequently, the following material aims to explain the challenges and terminology associated with information extraction and subsequent processing.

2.2 Natural Language Processing(NLP)

- For the purpose of explaining further concepts, Twitter will be used as a running example.
- The data retrieved from twitter presents a certain amount of structuring, in the sense that the maximum length of a tweet is 140 characters long.
- The advantage of the length limit is reflected in the complexity of the analysis for an individual piece of text.
- However, this project aims to analyses data in a continuous manner, where a large amount of data (e.g.: 200 tweets per minute) will be analyzed.
- Furthermore, there is no certainty that all the tweets will follow a formal structure, neither that they will be grammatically correct.
- It is also expected that abbreviations and short forms of words, as well as slang will be encountered in the text analyzed.
- Moreover, sentences describing the same or similar ideas may have very different syntax and employ very different vocabularies.
- Given the aforementioned textual limitation, a predefined textual format has to be produced at

processing time.

- The techniques presented below were used in the development of the project.

2.2.1 Tokenization

- The first task, that must be completed before any processing can occur, is to divide the textual data into smaller components.
- This is a common step in a Natural Language Processing (NLP) application, known as tokenization.
- At a higher level, the text is initially divided into paragraphs and sentences.
- As a consequence of the length limitation of 140 characters imposed by twitter, it is rarely the case that a tweet will contain more than a paragraph.
- In these regards, the project aim at this step is to correctly identify sentences.
- This can be done by interpreting the punctuation marks such as a period mark ".", within the text analysed.
- The next step is to extract the words (tokens) from sentences.
- The challenge at this step is to handle the orthography within a sentence.
- Consequently, spelling errors have to be corrected, URLs and punctuation shall be excluded from the resulting set of tokens.
- As it can be observed in Figure 2.1, after tokenizing a tweet the returned result is is an array containing a set of strings.

2.2.2 Part of Speech Tagging (POS)

- In order to understand the complete meaning of a sentence, the relationship between its words have to be established.
- This can be done by assigning every word a category that identifies syntactic functionality of that word.
- Also known as part of speech tagging (POS), this step can be seen as an auxiliary requirement for n-grams selection and lemmatization.
- Table 2.1 covers the part of speech notations used in the project.

ADJ : adjective	PART : particle
ADV : adverb	PRON : pronoun
AUX : adjective	PROPN : proper noun
CONJ: conjunction	PUNCT : punctuation
DET: determiner	SYM : symbol
NOUN: noun	VERB: verb
NUM: numeral	X : other

Table 2.2.2.1: Part of speech tags used throughout the project

2.2.3 Stemming and Lemmatization

- The goal of both stemming and lemmatization is to reduce inflectional forms and derivations of a word to a common base form.
- For example the following words: “connection”, “connections”, “connective”, “connected”, “connecting” will have the same base, which is “connect”.
- Stemming, is a crude heuristic process that chops off the ends of words, so that only the base form is kept.
- By contrast, lemmatization uses the morphological analysis of the words, returning their dictionary form (base), commonly referred as the lemma.
- However, for a language like English as opposed to more morphologically rich languages, this process relies on a dictionary being available.
- In addition, a lemmatizer can introduce ambiguity by proposing all possible lemmas for a word form, or by choosing the wrong proposal from two competing lemmas (e.g., is axes the plural of axe or of axis?).

2.2.4 N-grams

- N-grams is a common technique in text mining, where word subsets of length n within a sentence are formed. From the sentence “This is a six words sentence!” the following n-grams can be formed:
- 1-grams (unigrams): “this”, “is”, “a”, “six”, “words”, “sentence”
- 2-grams (bigrams): “this is”, “is a”, “a six”, “six words”, “words sentence”
- 3-grams (trigrams): “this is a”, “is a six”, “a six words”, “six words sentence”
- As such, the example sentence above will produce 6 unigrams, 5 bigrams, and 4 trigrams. On a bigger data set, producing bigrams and trigrams will considerably contribute to the size of the data set, consequently, slowing down the system.

2.3 Machine Learning Classification

- The rest of this section will present machine learning algorithms used to classify the polarity (positive, negative, neutral) of the tweets in their normalized form. The term machine learning refers to the “automated detection of meaningful patterns in data” [5]. Alongside with the growing size of the data produced, machine learning has become a common technique for information extraction. From spam filtering and personalized advertising, to search engines and face detection software, machine learning is applied in a wide range of domains [5]. While the variety of present algorithms depends on the learning task, specialized literature makes the distinction according to the nature of interaction between the computer and the

environment. As such, the separation is made between supervised and unsupervised machine learning algorithms:

- **Supervised Learning**: In a supervised machine learning algorithm the training data "comprises examples of the input vectors along with their corresponding target vectors (classes)". For example, in a supervised learning manner a computer can be thought to distinguish between pictures of cats and pictures of dogs. In the training phase, a set of labelled pictures will be processed by the algorithm. At this point, the computer 'knows' which pictures contain cats and which contain dogs. When presented with new unlabeled pictures, the algorithm will decide based on what it 'saw' before, the type of animal in the picture. Hence, the goal is to 'learn' a general rule that maps input to output.

- **Unsupervised Learning:** Unsupervised machine learning algorithms have the same scope as supervised learning, which is to map input to output. However, the difference is that in the training phase the input is not labelled, consequently, the computer has to find structure in the input, without specifically being told how to classify.

- As part of the project, a supervised approach was desirable. Consequently, two algorithms were used, one implemented and the other taken from a pre-implemented library which serves as a comparison base for the evaluation process. The rest of this chapter will explain in detail the Naïve Bayes algorithm, and offer a brief description of the Support Vector Machine algorithm.

CHAPTER-3

SYSTEM DESIGN

3.1 Flow Chart of System

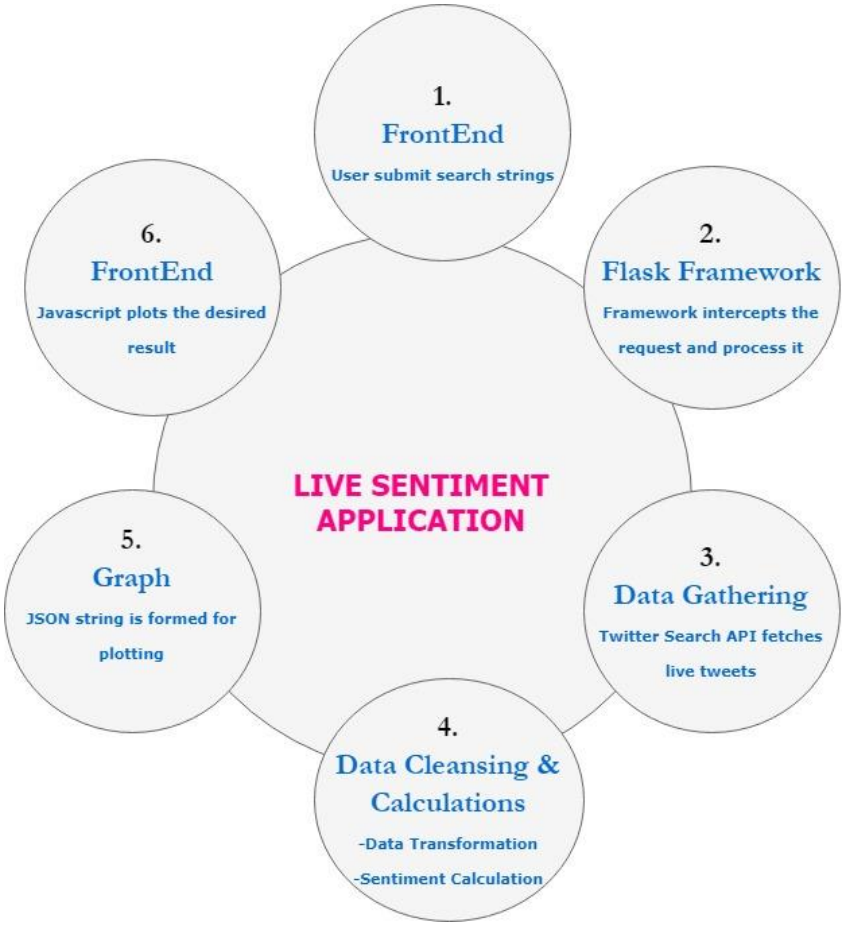

Figure 3.1.1 Frontend WorkFow of our Tool

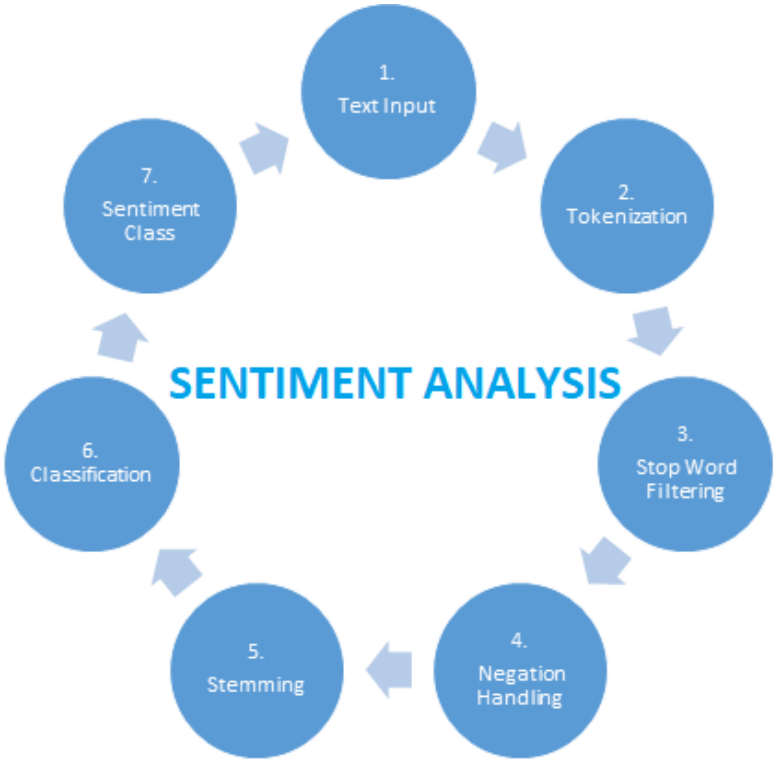

Figure 3.1.2 Backend WorkFow of our Tool

3.2 Major Functionality

- This tool provides sentiment in the form of positivity, negativity or neutrality.
- Brands can use this tool to measure the success of their products in an objective manner.
- This tool can be used for customer feedback monitoring.
- This tool can be used for Product Analysis.
- This tool can also be used in Social Media Monitoring.

3.2 GUI Snapshot

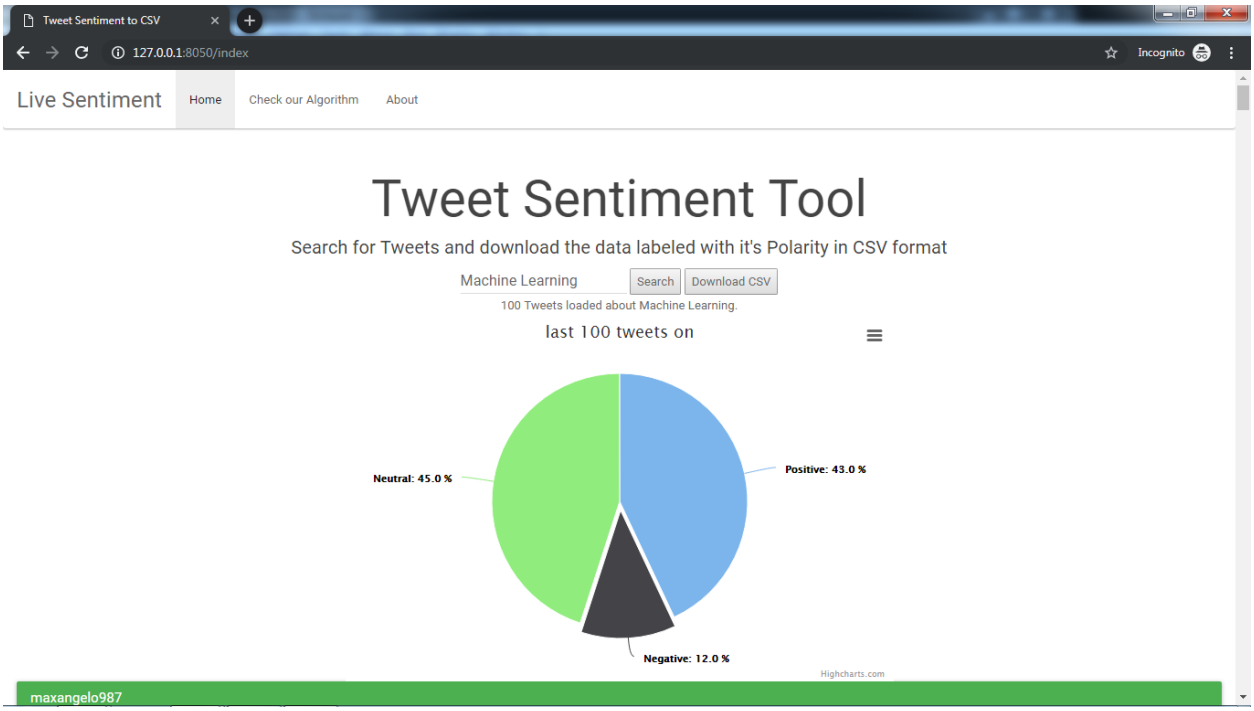

Figure 3.2.1 Front page-1

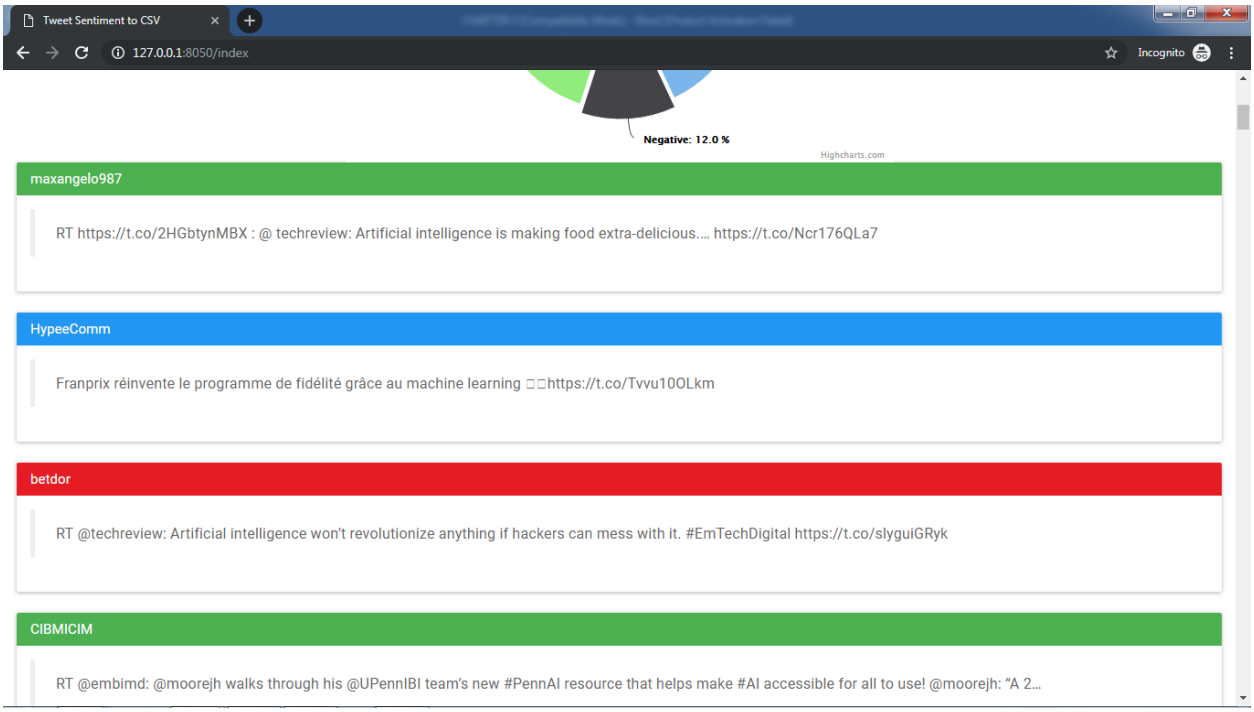

Figure 3.2.2 Front page-2

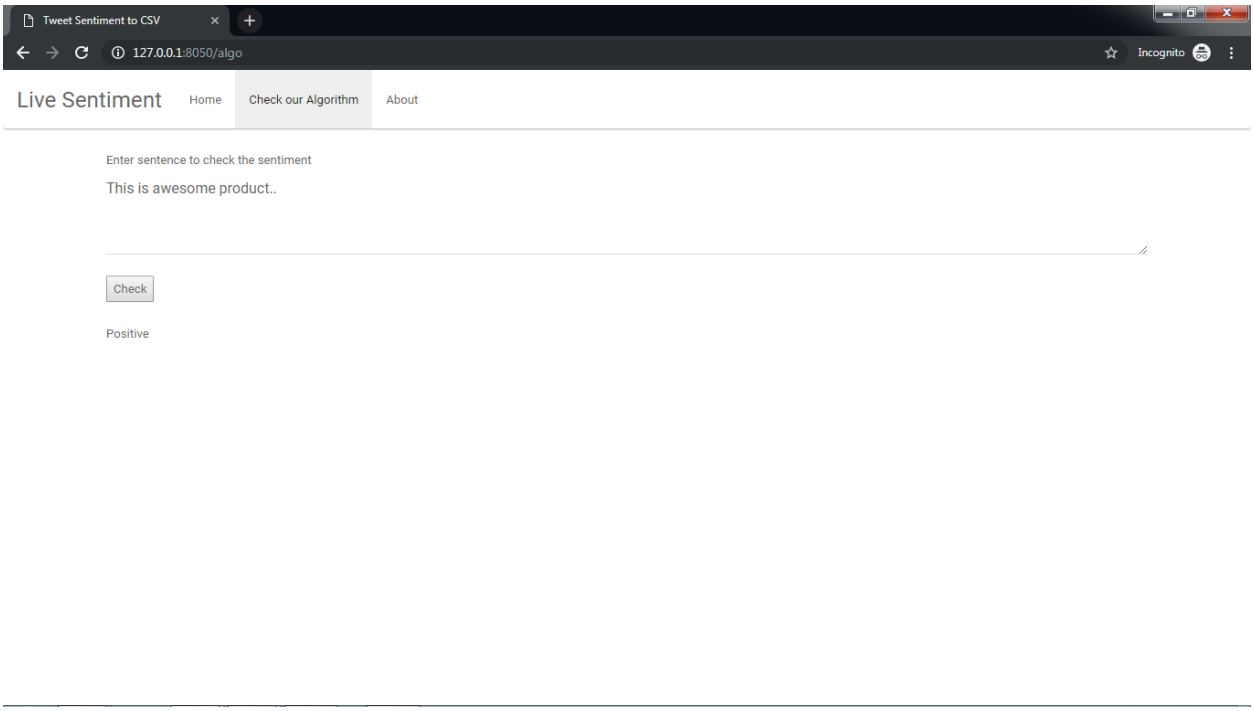

Figure 3.2.3 Check Algorithm

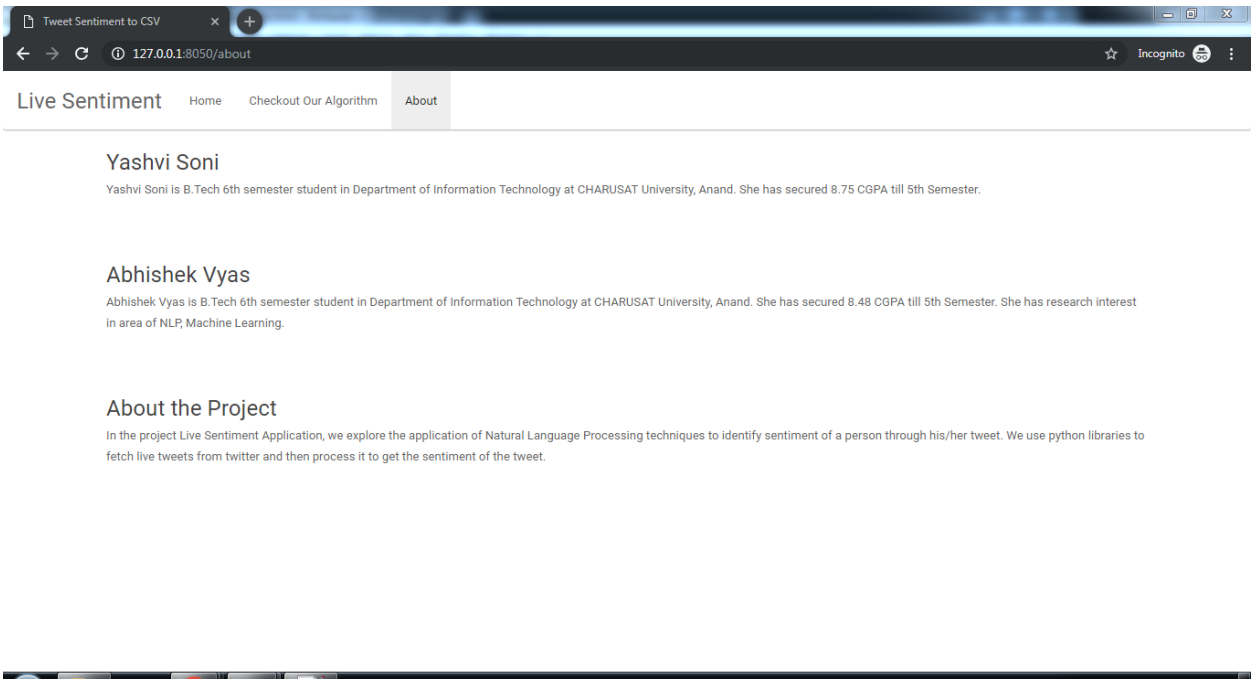

Figure 3.2.4 About page

CHAPTER:-4
IMPLEMENTATION AND PLANNING DETAILS

4.1 IMPLEMENTATION ENVIRONMENT:

- The project was a result of duo hard work. Decision on the problem was made mutually by me, my partner and my guide. Communication among us was horizontal.

4.1.1 Implementation Planning:

- Implementation phase requires precise planning and monitoring mechanism in order to ensure schedule and completeness. We implemented created a tool that gives us the accurate sentiment.

4.2 CODING STANDARD:

- Indeed, coding and applying logic is the foundation of any programming language but there's also another factor that every coder must keep in mind while coding and that is the coding style.
- Keeping this in mind, Python maintains a strict way of order and format of scripting.
- Following this sometimes mandatory and is a great help on the user's end, to understand.
- Making it easy for others to read code is always a good idea, and adopting a nice coding style helps tremendously for that.
- For Python, PEP 8 has emerged as the style guide that most projects adhere to; it promotes a very readable and eye-pleasing coding style. Every Python developer should read it at some point; here are the most important points extracted for you:

1. Use 4-space indentation and no tabs.
2. Use doc strings: There are both single and multi-line doc strings that can be used in Python. However, the single line comment fits in one line, triple quotes are used in both cases. These are used to define a particular program or define a particular function.
3. Wrap lines so that they don't exceed 79 characters: The Python standard library is conservative and requires limiting lines to 79 characters. The lines can be wrapped using parenthesis, brackets, and braces. They should be used in preference to backslashes.
4. Use of regular and updated comments are valuable to both the coders and users: There are also various types and conditions that if followed can be of great help from programs and users point of view. Comments should form complete sentences. If a comment is a full sentence, its first word should be capitalized, unless it is an identifier that begins with a lower case letter. In short comments, the period at the end can be omitted. In block comments, there are more than one paragraphs and each sentence must end with a period. Block comments and inline comments can be written followed by a single '#'.
5. Use of trailing commas: This is not mandatory except while making a tuple.
6. Use Python's default UTF-8 or ASCII encodings and not any fancy encodings, if it is meant for international environment.

7. Use spaces around operators and after commas, but not directly inside bracketing constructs.
8. Naming Conventions: There are few naming conventions that should be followed in order to make the program less complex and more readable. At the same time, the naming conventions in Python is a bit of mess, but here are few conventions that can be followed easily.
9. Characters that should not be used for identifiers: 'l' (lowercase letter el), 'O' (uppercase letter oh), or 'I' (uppercase letter eye) as single character variable names as these are similar to the numerals one and zero.
10. Don't use non-ASCII characters in identifiers if there is only the slightest chance people speaking a different language will read or maintain the code.
11. Name your classes and functions consistently: The convention is to use CamelCase for classes and lower_case_with_underscores for functions and methods. Always use self as the name for the first method argument.
12. While naming of function of methods always use self for the first argument to instance methods and cls for the first argument to class methods. If a functions argument name matches with reserved words then it can be written with a trailing comma. For e.g., class_

CHAPTER- 5
FUTURE ENHANCEMENT

5.1 Future Enhancement:

- Analysing sentiments on emo/smiley.
- Potential improvement can be made to our data collection and analysis method.
- Future research can be done with possible improvement such as more refined data and more accurate algorithm.

CHAPTER- 6
CONCLUSION

CONCLUSION

- We have completed our project using python as language, with Html and Javascript for output presentation. Although there was a problem in integration of python and javascript, through numbers of tutorial we were able to integrate it.
- We were able to determine the positivity, negativity and neutrality of each tweet. Based on those tweets we represented them in a diagrams like Pie-chart.
- All the diagrams related to outcome are shown in fig(4.1).
- All displaying results are displayed in webpage.

REFERENCES

- Python Flask Tutorials - https://www.tutorialspoint.com/flask/index.htm
- Sentiment Analysis - https://www.analyticsvidhya.com/blog/2018/07/hands-on-sentiment-analysis-dataset-python/
- Twitter APIs. - https://dev.twitter.com/start.
- Sisira Neti, S. (2011). SOCIAL MEDIA AND ITS ROLE IN MARKETING. 1st ed. [ebook] International Journal of Enterprise Computing and Business Systems. Available at: http://www.ijecbs.com/July2011/13.pdf [Accessed 15 Feb. 2019].
- Nlp.stanford.edu. (2016). Stemming and lemmatization. [online] Available at: http://nlp.stanford.edu/IR-book/html/htmledition/stemming-and-lemmatization-1.html[Accessed 14 Feb. 2019].