

PATTERN MATCHING FOR SOCIAL SENTIMENTAL ANALYSIS

Project Report Submitted To



**APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY
KERALA**

In Partial fulfillment of the requirements for the award of the degree

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

Submitted by

AMRUTHA SUDARSANAN (Reg.no:ICE16CS008)

RINU MONCY(Reg.no:ICE16CS043)

SARANYA BALAKRISHNAN (Reg.no:ICE16CS048)

SIMI JAMES(Reg.no:ICE16CS055)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ILAHIA COLLEGE OF ENGINEERING & TECHNOLOGY

MULAVOOR P. O, MUVATTUPUZHA-686673

JUNE 2020

ILAHIA COLLEGE OF ENGINEERING & TECHNOLOGY
MULAVOOR P.O, MUVATTUPUZHA-686673



CERTIFICATE

This is to certify that the report entitled **“PATTERN MATCHING FOR SOCIAL SENTIMENTAL ANALYSIS”** submitted by **AMRUTHA SUDARSANAN (Reg.no:ICE16CS008), RINU MONCY(Reg.no:ICE16CS043), SARANYA BALAKRISHNAN(Reg.no:ICE16CS048) & SIMI JAMES (Reg.no:ICE16CS055)** to the APJ Abdul Kalam Technological University, Kerala in partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology in Computer Science and Engineering is a bonafide record of the project work carried out by him/her under my/our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purposes.

Ms. Theresa Jose

Project Guide
Assistant professor
Dept of CSE, ICET

Prof. Dr.E Arun

Head Of the Department
Dept of CSE, ICET

Submitted for the final project presentation and viva – voce examination held on Ilahia College of Engineering and Technology.

Internal Examiner

External Examiner

DECLARATION

We affirm that the project work entitled “PATTERN MATCHING FOR SOCIAL SENTIMENTAL ANALYSIS” being submitted in partial fulfillment for the award of the degree of Bachelor of Technology in Computer Science and Engineering is the original work carried out by us. It has not formed the part of any other project work submitted for the award of any degree or diploma, either in this or any other university or institute or published earlier.

AMRUTHA SUDARSANAN (Reg.no:ICE16CS008)

RINU MONCY(Reg.no:ICE16CS043)

SARANYA BALAKRISHNAN (Reg.no:ICE16CS048)

SIMI JAMES (Reg.no:ICE16CS055)

I certify that the declarations made above by the students are true to the best of my knowledge and belief.

Ms.Theresa Jose

Project Guide

Assistant Professor

Dept of CSE

ICET, Muvattupuzha

ACKNOWLEDGEMENT

Words cannot express our gratitude to the scholars we know, yet we would like to thank those who enabled the successful completion of this project. First and foremost thank “**THE ALMIGHTY**” for his grace and mercy that enabled us to complete the work successfully.

We express our sincere thanks to **Prof. Dr. M. Mohamed Sitheeq**, Principal, Ilahia College of Engineering and Technology, Muvattupuzha for his kind patronage.

We express our sincere gratitude to **Prof.Dr. E Arun**, Head of the Department of Computer Science and Engineering for the encouragement and helping us to the facilities in the college for doing this project.

We express our sincere gratitude to **Ms.Theresa Jose** , Asst. Professor in the Department of Computer Science and Engineering for guiding us to do this project.

Next we express our sincere gratitude to our project coordinators **Dr. E. Arun, Ms.Hafsath CA, Ms.Nurjahan VA, Ms. Rosna P. Haroon, Ms. Theresa Jose** Asst. Professors in Computer Science & Engineering Department for constant encouragement during this project.

We also take the opportunity to thank all the person associated with this project ideas and cooperation have helping in doing this project. In addition, a special thanks to all my friends and parents for their consideration and motivation.

ABSTRACT

In modern era, most of the peoples are depending on social media. They are very valuable and important resources for people to understand and study the changing world. Social media enables users to easily post their opinions and perspectives regarding certain issues. But it is difficult to understand the general impression of people in such issues. This is especially a problem for the tweets sentiment analysis. This paper aims at using text mining techniques to explore public opinion contained in social media by analyzing the reader's emotion towards pieces of short text. With the booming of these social media, sentiment analysis has developed rapidly in recent years. Sentiment classification is a special task of text classification whose objective is to classify a text according to the sentimental polarities of opinions. It contains e.g. positive or negative. Pattern matching technique is used for the categorization of a text. In addition to the text we also consider emojis for the categorization. We use SVM(Support Vector Machine) to train our classifier. we combine a visualized analysis method for keywords that can provide a deeper understanding of opinions expressed on social media topics.

CONTENTS

CHAPTER NO	TITLE	PAGE NO
	LIST OF FIGURES	II
1	INTRODUCTION	1
2	LITERATURE REVIEW	4
3	FEASIBILITY ANALYSIS	8
4	SYSTEM DESIGN	11
5	SOLUTION METHODOLOG Y	24
6	IMPLEMENTATION	27
7	TESTING	34
8	RESULT AND DISCUSSION	36
9	CONCLUSION	40
	REFERENCE	41
	APPENDIX	42

LIST OF FIGURES

FIGURE NO.	TITLE	PAGE NO.
4.1	SYSTEM ARCHITECTURE	12
4.2	AUTOMATION REPRESENTATION	15
4.3	CLASSIFICATION GRAPH	16
4.4	SCENARIO-1	18
4.5	SCENARIO-2	18
4.6	GRAPH OF COMPARISON	19
4.7	SCENARIO-3	19
4.8	SCENARIO-4	20
4.9	SCENARIO-4 WITH HYPER- PLANE	20
4.10	SCENARIO-5	21
4.11	GRAPH WITH ADDITIONAL FEATURE	21
4.12	HYPER-PLANE IN INPUT SPACE	22
5.1	DATA FLOW DIAGRAM	25
5.2	FLOW CHART	26

CHAPTER 1

INTRODUCTION

Due to the booming of social media in the past few years, a spectacular amount of data has been produced. It is a very valuable and important resource for people to understand public opinion. Analyzing public opinion is critical to understanding the general impression of a given topic. It can be achieved through an investigation of social media. One of the numerous applications of this technology is to understand the trends in political elections. During the period of the election, a candidate can utilize the public opinions expressed on the social media to capture important issues and make corresponding adjustments in order to gain more support from the general public. For instance, during the Taipei mayoral election, the candidate Dr. Wen-je Ko's campaign used public opinion analysis to determine the public's favoured keywords. Policies and activities were then organized according to the interests of younger voters. The investment in public opinion analysis helped Dr. Ko win the election. This case demonstrates that exploring and analyzing social media can be a powerful means to understand the trends of public opinion.

Sentiment analysis is a significant area of research in natural language processing (NLP). Its purpose is to determine the attitudes and feelings expressed in words and their context. It can be separated into two categories, namely, writer emotion and reader emotion. The former refers to the emotion that the writer (author) wants to express when writing an article. The writer usually expresses emotion toward specific issues through emotive language. On the other hand, reader emotion corresponds to the feelings that may be triggered as one reads the articles. One important distinction is that writer and reader can have different perspectives on the same content, so their feelings may not be the same. It is not trivial to determine the reader's emotion directly through the writer's words. So, compared to the research of writer emotion, the research of reader emotion is more challenging. For example, consider a news title mentioning that the oil price will increase tomorrow such as, 'Gas Prices Rising Tomorrow!' Although there is nothing emotional in it, the reaction of the public toward this news title is presumably negative.

Unfortunately, recent emotional analysis research is mostly focused on writer emotion. Only a few researchers, including targeted reader emotion. Related research on reader emotion

classification mainly focused on the entire composition. They aim to learn a language model that identifies reader emotion, and subsequently use the learned model to assist in emotional resonance writing. The experimental results prove that this method can effectively recognize reader emotion through sentence and semantic structures in the compositions. Changet al. used the whole context of the news article to classify reader emotion, with special focus on products,movies and literatures reviews. In their research, they proposed an innovative reader emotion classification module through an effective use of emotional keywords. The technique was then applied to the news corpus as practical case study. The researchers calculated each candidate's publicity score based on reader emotion classification, and later predicted the voting trend. They were successful in correlating the percentage of votes obtained with publicity score and reader emotion.

Most of the content on social media consists of short texts with about 200 words. Due to the lack of context data, the efficiency of machine learning models are impaired. However, in the past few years, the research focusing on short text has prospered. Like BharathSriram et al extracted specific field's features from the author's profile and written words, then predefined the data as news, opinions, private messages, etc., in order to improve the efficiency of classifying the short text data. In order to improve the efficiency of dealing with the short text, Guo etal linked the short text and news corpus for expanding the content, letting the machine more easily understand the short text.

In consideration of the importance of social media analysis and the fact that no previous work was done on reader emotion analysis based on short text, this research aims at obtaining public opinion through an analysis of reader emotion. Consequently, we proposed a method which can analyze the reader emotion of short text. As the experiment result shows,our method can effectively recognize different reader emotion categories. Furthermore, we used the visualization method to understand more about the result. Our research can efficiently obtain the public opinion of related topics and more detailed information about it.

1.1 Objectives

The objectives of this project are:

- To implement an algorithm for automatic classification of text into positive and negative.
- Sentiment Analysis to determine the attitude of the mass is positive, negative or neutral towards the subject of interest.
- Graphical representation of the sentiment in form of Pie-Chart or Bar Diagram

1.2 Scope of project

This project will be helpful to the companies, political parties as well as to the common people. It will be helpful to political party for reviewing about the program that they are going to do or the program that they have performed. Similarly companies also can get review about their new product on newly released hardwares or softwares. Also the movie maker can take review on the currently running movie. By analyzing the tweets analyzer can get result on how positive or negative or neutral are peoples about it.

1.3 System Overview

This proposal entitled “PATTERN MATCHING FOR SOCIAL SENTIMENTAL ANALYSIS” is a web application which is used to analyze the tweets. We will be performing sentiment analysis in tweets and determine where it is positive, negative or neutral. This web application can be used by any organization office to review their works or by political leaders or by any others company to review about their products or any common people to known others opinion

1.4 System Features

The main feature of our web application is that it helps to determine the opinion about the peoples on products, government work, politics or any other by analyzing the tweets. Our system is capable of training the new tweets and analysed data is represented in the form of graph

CHAPTER 2

LITERATURE REVIEW

Dawei Li , Yujia Zhang , and Cheng Li[1] “Mining Public Opinion on Transportation Systems Based on Social Media Data ”: This paper focuses on text analysis using large data with temporal and spatial attributes of social network platform. Web crawler technology is used to obtain traffic-related text in mainstream social platforms. After basic treatment, the emotional tendency of the text is analyzed. Then, based on the probabilistic topic modeling (latent Dirichlet allocation model), the main opinions of the public are extracted, and the spatial and temporal characteristics of the data are summarized. Taking Nanjing Metro as an example, the existing problems are summarized from the public opinions and improvement measures are put forward, which proves the feasibility of providing technical support for public participation in public transport with social media big data.

In this paper, the extraction and processing of traffic-related microblog text is still imperfect, which may lead to subsequent results affected. Direct use of traditional LDA model for topic modeling of microblog, to a certain extent, is still affected by the size, content, scattered format, data noise, and other aspects. The efficiency of LDA topic model is also influenced by the length of documents. The lack of sufficient words in a short text will affect the effectiveness of topic modelling. Mining the temporal and spatial characteristics of text data is relatively simple, such as not using the geographic location information published by microblog users.

After the implementation of policies and projects, it can be used as a public supervision mechanism to collect public opinions in time to respond and deal with them as soon as possible. The timely handling of public opinions can also encourage the public to put forward their own valuable opinions for the development of urban traffic on the social network according to what they have seen and heard, and make up for the loopholes and shortcomings of planning.

George Stylios, Dimitris Christodoulakis, Jeries Besharat, Maria-Alexandra Vonitsanou, Ioanis Kotrotsos, Athanasia Koumpouri and Sofia Stamou Patras University, Greece [2] “Public Opinion Mining for Governmental Decisions’:

In this paper, it propose the exploration of text and data mining techniques towards capturing the public's opinion communicated online and concerning governmental decisions. The objective of this study is twofold and focuses on understanding the citizen opinions about eGovernment issues and on the exploitation of these opinions in subsequent governmental actions. They examine several features in the user-generated content discussing governmental decisions in an attempt to automatically extract the citizen opinions from online posts dealing with public sector regulations and thereafter be able to organize the extracted opinions into polarized clusters. There goal is to be able to automatically identify the public's stance against governmental decisions and thus be able to infer how the citizens' viewpoints may affect subsequent government actions. To demonstrate the usability and added value of this proposed approach they designed an interactive eGovernment infrastructure, the architecture of which it will present and discuss in this paper. Moreover, it will elaborate on the system details, its adaptation capacity and it will discuss its usage benefits for both citizens and public sector bodies.

This paper presented a method for extracting citizen opinions about governmental decisions from social media sites, as well as a technique for classifying opinion phrases in terms of their sentiment orientation. In addition, they proposed the architecture of an interactive eGovernment platform that encapsulates the mined user opinions and explores them in subsequent governmental actions. A metric for quantifying the impact of citizen opinions on governmental decisions is also proposed so that the former can be fruitfully employed in subsequent governmental regulations. The application of this proposed method over a set of real user content reveals that properly processed and analyzed opinion phrases can serve as useful indicators for the perception of governmental decisions by the public.

This method relies on the intuition that there is plentiful data available on social web sites that communicates implicit information about how citizens perceive governmental regulations and directives. Being able to collect, process and mine such data can provide decision-makers with valuable information about how the recipients of their actions evaluate the latter and it can also em-power citizens with the ability to actively participate in governmental decision making aspects. Today, all EU Member States have ICT policies and consider them a key contributor to national growth and jobs under the renewed Lisbon agenda. eParticipation is the strongest growing area of eGovernment Action Plan. "eParticipation" is about reconnecting ordinary people with politics and policy-making and

making the decision-making processes easier to understand and follow through the use of new ICT.

Chang et al. proposed a flexible principle-based approach (PBA) for reader-emotion classification and writing assistance. PBA can capture variations of similar expressions by generating distinctive emotion templates. For example, “{國家country}” : [發生occur] : [地震 earthquake] : {劫難disaster}” is generated by PBA for the emotion “worried” template. We can observe that this template captures prominent information about natural disasters (earthquake) that have happened in a specific country, and also expresses that it is related to the emotion “worried”. PBA can automatically learn emotion templates from raw data and produce the reasonable emotion templates for humans to understand. The writers can then follow the templates and compose articles that resonate with readers.

Bashaddadh et al. used Named Entities to perform topic detection and tracking (TPT), which is a useful method in the information retrieval field. They used keywords and name entities to cluster the vast information from the internet. The importance of keywords has been noticed by many researchers. Further experiments have been conducted based on critical words or keywords. Tang et al. examined the top 200 words with higher relative log frequency ratios in the categories of their target and linked those words to positive and negative reader and writer emotion. After employing relative log frequency ratios to mine sentiment words, they used those words to predict emotion transition by building 4-class and 2-class SVM classifiers. Their results show that using the sentiment keywords method is useful.

Furthermore, Chang et al. proposed an innovative document modeling method with emotional keyword embedding, which is called distributed emotion keyword vector (DEKV), to classify reader-emotion. In their research, they treat keywords of each category very seriously. They use keywords to represent articles and take a likelihood ratio as weighting. If there were some articles that couldn't be match with keywords, they calculated the cosine similarity of the word's vector and found the closest keywords to represent those articles. In the case study, they calculated the publicity score based on reader emotion classification for each candidate and successfully predicted the voting trend. Using hash tags to analyze social media data sounds like a good idea. Hashtags are specified keywords in social media posts.

Kouloumpis et al. used Twitter hashtags (e.g., #bestfeeling, #epicfail, #news) to identify positive, negative, and neutral tweets. These were then used for training the three-way sentiment classifiers. They wanted to evaluate their training data with labels derived from hashtags and emotions. This is useful for training sentiment classifiers for Twitter or other social media platforms. The results proved that using hashtags to collect training data is useful for classification.

A huge number of informal messages are posted every second on social media platforms, which are mostly in short text form. It is more difficult for machines to comprehend and classify short texts when compared to whole paragraphs (about 200 words). Bharath Sriram et al. bring up a method which is called SentiStrength. It can predict positive emotion with 60.6% accuracy and negative emotion with 72.8% accuracy on short text from social media. Different from the existing emotion classification methods that tend to be commercially used, they focused on the emotion of the users. Their research started from the users' behavior to classify the emotion, like the authors' profile or written words from the past.

Guo et al. noticed that the classification efficiency is quite low when the past research methods, which are designed for large context of text, used on the short texts like posts on Twitter. They proposed a method for short texts classification, which is called Linking-Tweets-to-News. It benefits most off-the-shelf NLP tools.

CHAPTER 3

FEASIBILITY ANALYSIS

A feasibility study is a preliminary study which investigates the information of prospective users and determines the resources requirements, costs, benefits and feasibility of proposed system. A feasibility study takes into account various constraints within which the system should be implemented and operated. In this stage, the resource needed for the implementation such as computing equipment, manpower and costs are estimated. The estimated are compared with available resources and a cost benefit analysis of the system is made. The feasibility analysis activity involves the analysis of the problem and collection of all relevant information relating to the project. The main objectives of the feasibility study are to determine whether the project would be feasible in terms of economic feasibility, technical feasibility and operational feasibility and schedule feasibility or not. It is to make sure that the input data which are required for the project are available. Thus we evaluated the feasibility of the system in terms of the following categories:

- Technical feasibility
- Operational feasibility
- Economic feasibility
- Schedule feasibility

3.1.1 Technical Feasibility

Evaluating the technical feasibility is the trickiest part of a feasibility study. This is because, at the point in time there is no any detailed designed of the system, making it difficult to access issues like performance, costs (on account of the kind of technology to be deployed) etc. A number of issues have to be considered while doing a technical analysis; understand the different technologies involved in the proposed system. Before commencing the project, we have to be very clear about what are the technologies that are to be required for the development of the new system. Is the required technology available? Our system is technically feasible since all the required tools are easily available. Java,html,MySQL can be easily handled. Although all tools seems to be easily available there are challenges too.

3.1.2 Operational Feasibility

Proposed project is beneficial only if it can be turned into information systems that will meet the operating requirements. Simply stated, this test of feasibility asks if the system will work when it is developed and installed. Are there major barriers to Implementation? The proposed was to make a simplified web application. It is simpler to operate and can be used in any webpages. It is free and not costly to operate.

3.1.3 Economic Feasibility

Economic feasibility attempts to weigh the costs of developing and implementing a new system, against the benefits that would accrue from having the new system in place. This feasibility study gives the top management the economic justification for the new system. A simple economic analysis which gives the actual comparison of costs and benefits are much more meaningful in this case. In addition, this proves to be useful point of reference to compare actual costs as the project progresses. There could be various types of intangible benefits on account of automation. These could increase improvement in product quality, better decision making, and timeliness of information, expediting activities, improved accuracy of operations, better documentation and record keeping, faster retrieval of information. This is a web based application. Creation of application is not costly.

3.1.4 Schedule Feasibility

A project will fail if it takes too long to be completed before it is useful. Typically, this means estimating how long the system will take to develop, and if it can be completed in a given period of time using some methods like payback period. Schedule feasibility is a measure how reasonable the project timetable is. Given our technical expertise, are the project deadlines reasonable? Some project is initiated 10 with specific deadlines. It is necessary to determine whether the deadlines are mandatory or desirable. A minor deviation can be encountered in the original schedule decided at the beginning of the project. The application development is feasible in terms of schedule.

3.2 Requirement Definition

After the extensive analysis of the problems in the system, we are familiarized with the requirement that the current system needs. The requirement that the system needs is categorized into the functional and non-functional requirements. These requirements are listed below:

3.2.1 Functional Requirements

Functional requirement are the functions or features that must be included in any system to satisfy the business needs and be acceptable to the users. Based on this, the functional requirements that the system must require are as follows: System should be able to analyze data and classify each tweet polarity

3.2.2 Non-Functional Requirements

Non-functional requirements is a description of features, characteristics and attribute of the system as well as any constraints that may limit the boundaries of the proposed system. The non-functional requirements are essentially based on the performance, information, economy, control and security efficiency and services. Based on these the non-functional requirements are as follows:

- User friendly
- System should provide better accuracy
- To perform with efficient throughput and response time

CHAPTER 4

SYSTEM DESIGN

This project aims at using text mining techniques to explore public opinion contained in social media by analyzing the readers emotion towards pieces of short text. The speculative model and its congruous functioning is being discussed in this section. Public opinion pattern matching for the mining of the short text and emotions and support vector machine classifier for the categorization of opinions. A visualized analysis method that can provide deeper understanding of opinions expressed on social media topics.

4.1 SYSTEM ARCHITECTURE

A system architecture is the conceptual model that defines the structure, behavior, and more views of a system.^[1] An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviors of the system. A system architecture can consist of system components and the sub-systems developed, that will work together to implement the overall system

First, we use pattern matching technique for each opinion category. Preprocessing used to transform the raw data in a useful and efficient format. Pattern matching is a process used to identify the number of occurrence of particular pattern in a given input data set which automate the categorization of pattern based on manner of input and their properties. Then by using support vector machine (SVM) classifier, we can target data from specific topics on social media and recognize public opinion. Finally, we propose a method of visualization, which can reveal more detail of each expressed public opinion. We will provide an in-depth explanation in the following sections. In the first section we are discussing about preprocessing. Then we look on to pattern matching and support vector machine classifier. Finally we discuss about the visualization.

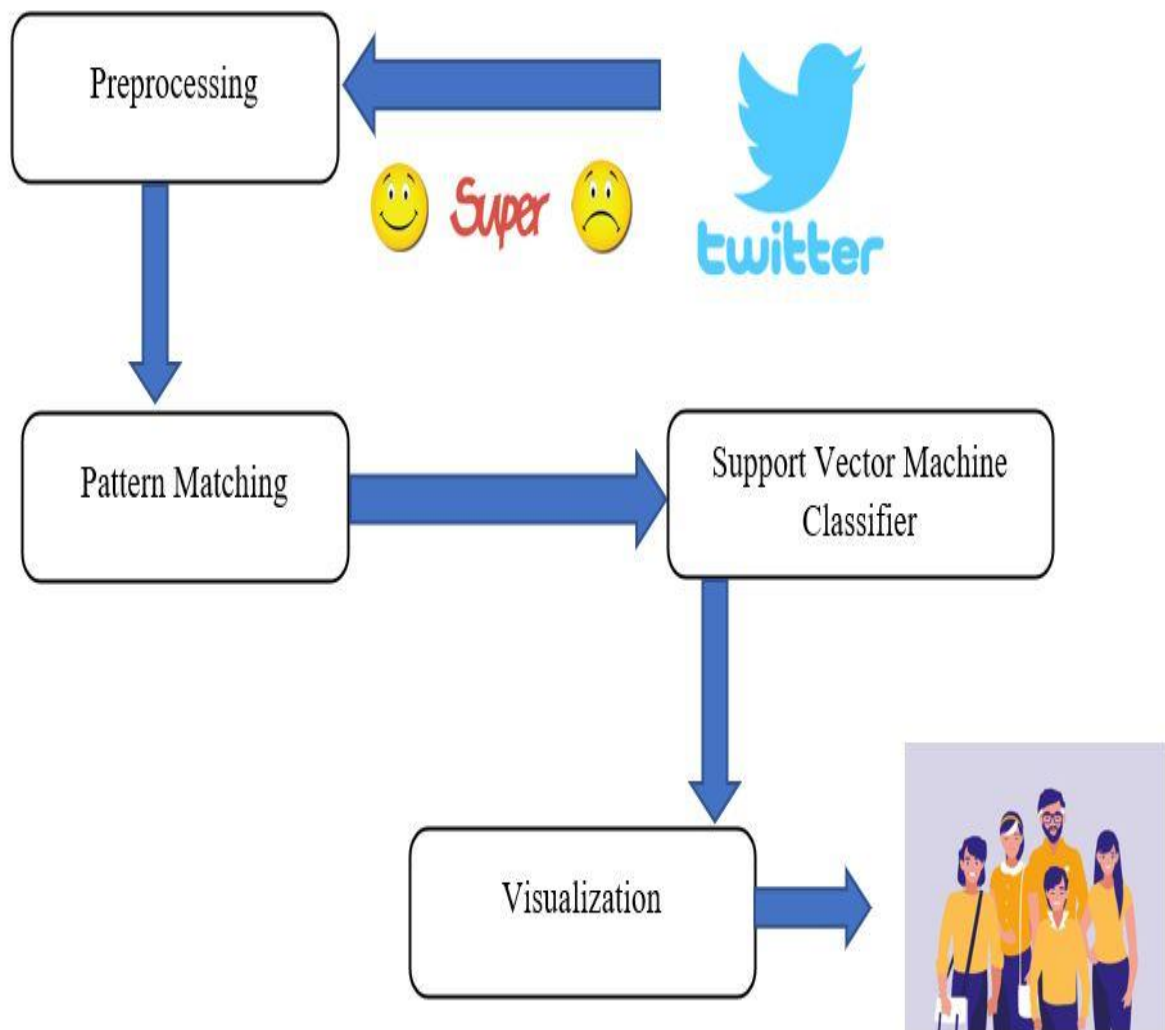
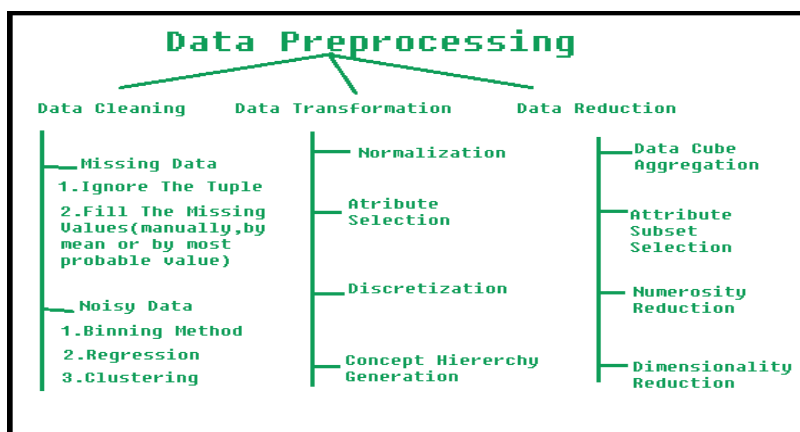


FIGURE 4.1 System Architecture

4.1.1 Preprocessing

Data preprocessing is a data mining technique which is used to transform the raw data in a useful and efficient format.



Steps Involved in Data Preprocessing:

1.DataCleaning:

The data can have many irrelevant and missing parts. To handle this part, data cleaning is done. It involves handling of missing data, noisy data etc.

- **(a).MissingData:**

This situation arises when some data is missing in the data. It can be handled in various ways.

Some of them are:

1. **Ignore the tuples:**

This approach is suitable only when the dataset we have is quite large and multiple values are missing within a tuple.

2. **Fill the missing values:**

There are various ways to do this task. You can choose to fill the missing values manually, by attribute mean or the most probable value.

- **(b).NoisyData:**

Noisy data is a meaningless data that can't be interpreted by machines. It can be generated due to faulty data collection, data entry errors etc. It can be handled in following ways :

1. **Binning Method:**

This method works on sorted data in order to smooth it. The whole data is divided into segments of equal size and then various methods are performed to complete the task. Each segment is handled separately. One can replace all data in a segment by its mean or boundary values can be used to complete the task.

2. **Regression:**

Here data can be made smooth by fitting it to a regression function. The regression used may be linear (having one independent variable) or multiple (having multiple independent variables).

3. **Clustering:**

This approach groups the similar data in a cluster. The outliers may be undetected.

2.DataTransformation:

This step is taken in order to transform the data in appropriate forms suitable for mining process. This involves following ways:

1. Normalization:

It is done in order to scale the data values in a specified range (-1.0 to 1.0 or 0.0 to 1.0)

2. AttributeSelection:

In this strategy, new attributes are constructed from the given set of attributes to help the mining process.

3. Discretization:

This is done to replace the raw values of numeric attribute by interval levels or conceptual levels.

4. Concept Hierarchy Generation:

Here attributes are converted from level to higher level in hierarchy.

3.DataReduction:

Since data mining is a technique that is used to handle huge amount of data. While working with huge volume of data, analysis became harder in such cases. In order to get rid of this, we uses data reduction technique. It aims to increase the storage efficiency and reduce data storage and analysis costs.

The cleaning is quite complicated in Twitter tweets but for our study, we use some regular expression techniques to clean our data. This means that we should be able to significantly decrease the amount of non-processed tweets. Using regular expression, we can remove all URLs and everything after by R regex. Regular expression typically specifies characters to seek out, possibly with information about repeats and location within the string. This is accomplished with the help of metacharacters that have a specific meaning: \$ * + . ? [] ^ { } | () \. Now, let us examine an example using the code below that was created using R programming language to remove all URLs and everything and after:

```
tweettext=lapply(tweet text, function(x)
```

```
gsub("htt.*",'',x))
```

The regular expression can be like a shot slang like “lol” (laughing out loud) that makes a great challenge when working with Twitter data. The cleaning is quite complicated in Twitter tweets but for our study, we use some regular expression

techniques to clean our data. In implementing automata theory, each character in the regular expression can represent a state to perform substring search which is finding a single string in a text. The regular expression is represented in finite automata below:

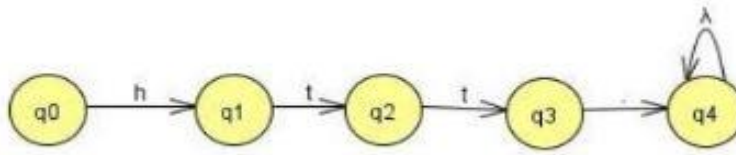


FIGURE 4.2 Automaton representation

Figure 4.2 is showing the R codes states is a directed graph, whose arcs are labeled by sets of characters. There is an arc from state q0 (considered as the initial state) to state q4 (final state), labeled by the set of characters h, t, t, . and the epsilon. To clean the data, the content of the twitter tweets will be parsed to do the pattern matching thru the regular expression. Since we are going to remove the URL in the tweets, the searching will look for the character h, then, t, t, . and can be any characters until it encounters the space that ends the removal process. Here is the other R code that implements regular expression to clean the datasets:

- # removes the hashtag
`tweettext=lapply(tweettext, function(x) gsub("#"," ",x))`
- # removes the punctuation
`singletweet = gsub("[[:punct:]]", "", singletweet)`
- # remove control characters
`singletweet = gsub("[[:cntrl:]]", "", singletweet)`
- # remove digits
`singletweet = gsub("\\d+", "", singletweet)`

4.1.2Pattern Matching

Pattern matching is a process used to identify the number of occurrence of particular pattern in a given input data set. Pattern matching which automate the categorization of pattern based on manner of input and their properties. Pattern matching is supervised learning and identifies the suitable match from different group of data and enhances number of events to improve the performance and efficiency of all events. In proposed system, generic pattern matching technique exclusive finds to confirm the identical measures between numbers of events.

Number of events stores and maintains all the identity of events internally. Patterns are checked with all stored patterns within the particular data set. Efficient computation techniques identify score function between two samples of pattern.

4.1.3 Support Vector Machine

A support vector machine (SVM) performs supervised learning for classification or regression of data groups. supervised learning systems provide both input and desired output data, which are labeled for classification. The classification provides a learning basis for future data processing. Support vector machines are used to sort two data groups by like classification. The algorithms draw lines (hyperplanes) to separate the groups according to patterns. An SVM builds a learning model that assigns new examples to one group or another. By these functions, SVMs are called a non-probabilistic, binary linear classifier. An SVM requires labeled data to be trained. Groups of materials are labeled for classification. Training materials for SVMs are classified separately in different points in space and organized into clearly separated groups. After processing numerous training examples, SVMs can perform unsupervised learning. The algorithms will try to achieve the best separation of data with the boundary around the hyperplane being maximized and even between both sides. A support vector machine (SVM) is a supervised machine learning model that uses classification algorithms for two-group classification problems. After giving an SVM model sets of labeled training data for each category, they're able to categorize new text. SVM is a supervised machine learning algorithm which can be used for classification or regression problems. It uses a technique called the kernel trick to transform your data and then based on these transformations it finds an optimal boundary between the possible outputs.

“Support Vector Machine” (SVM) is a supervised machine learning algorithm which can be used for both classification or regression challenges. However, it is mostly used in classification problems. In the SVM algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiates the two classes very well (look at the below snapshot).

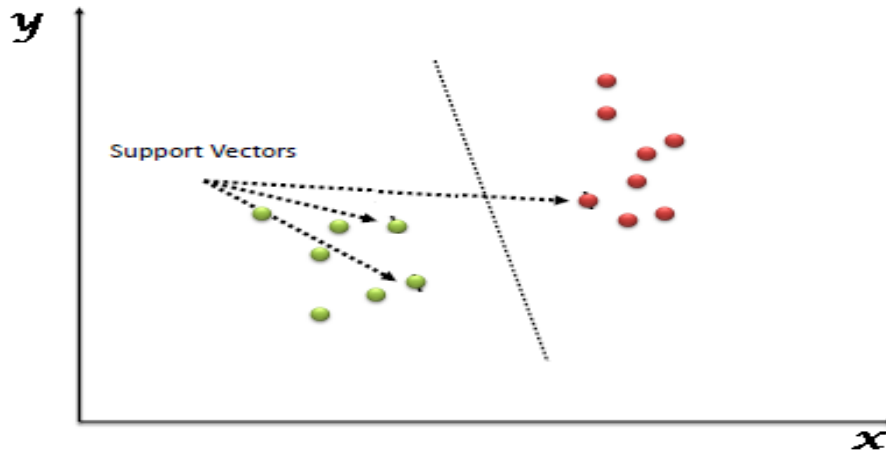


FIGURE 4.3 Classification graph

Support Vectors are simply the co-ordinates of individual observation. The SVM classifier is a frontier which best segregates the two classes (hyper-plane/ line)

How does it work?

Find a hyperplane that makes the margin between two categories.

SVM creates a hyper planes or a set of hyper planes in infinite dimension space. The SVM score z_j of a document is mathematically given as follows:

$$z_j = w_1x_{j1} + w_2x_{j2} + \dots + w_dx_{jd} + b$$

$$\text{i.e. } z_j = x_j^T w + b$$

where,

x_i is a p-dimensional real vector.

w is vector that contains the weights and is given as

$$\vec{w} = \sum_j \alpha_j c_j \vec{d}_j, \quad \alpha_j \geq 0, c_j = \{1, -1\}$$

b is a constant

- **Identify the right hyper-plane (Scenario-1):** Here, we have three hyper-planes (A, B and C). Now, identify the right hyper-plane to classify star and circle.

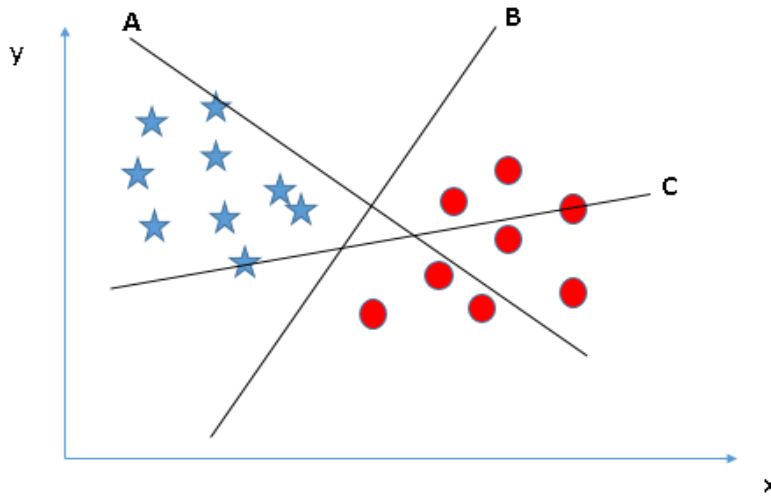


FIGURE 4.4 Scenario-1

- You need to remember a thumb rule to identify the right hyper-plane: “Select the hyper-plane which segregates the two classes better”. In this scenario, hyper-plane “B” has excellently performed this job.
- **Identify the right hyper-plane (Scenario-2):** Here, we have three hyper-planes (A, B and C) and all are segregating the classes well. Now, How can we identify the right hyper-plane?

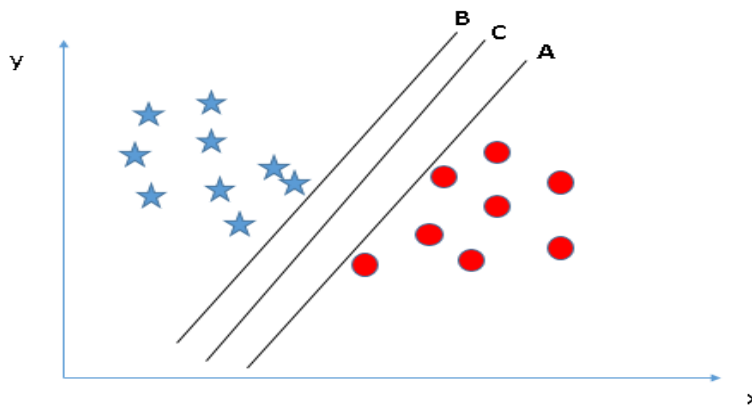


FIGURE 4.5 Scenario-2

Here, maximizing the distances between nearest data point (either class) and hyper-plane will help us to decide the right hyper-plane. This distance is called as **Margin**. Let's look at the below snapshot:

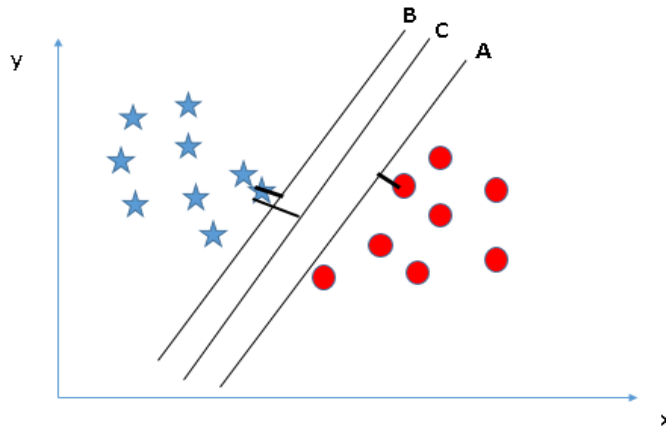


FIGURE 4.6 Graph of comparison

Above, you can see that the margin for hyper-plane C is high as compared to both A and B. Hence, we name the right hyper-plane as C. Another lightning reason for selecting the hyper-plane with higher margin is robustness. If we select a hyper-plane having low margin then there is high chance of miss-classification.

- **Identify the right hyper-plane (Scenario-3):**Hint: Use the rules as discussed in previous section to identify the right hyper-plane

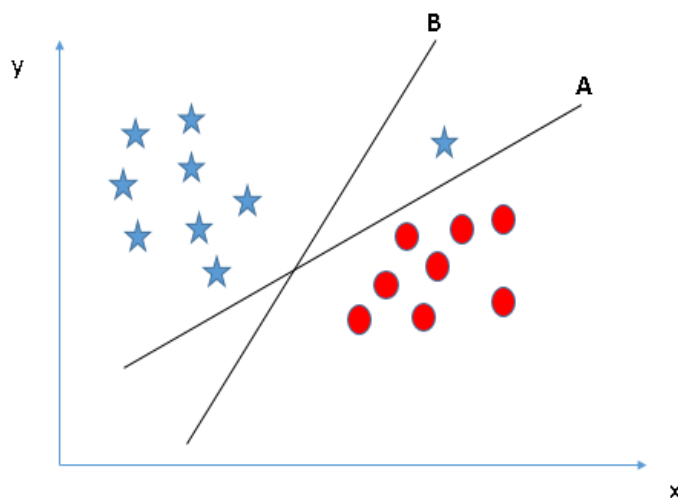


FIGURE 4.7 Scenario-3

Some of you may have selected the hyper-plane **B** as it has higher margin compared to **A**. But, here is the catch, SVM selects the hyper-plane which classifies the classes accurately prior to maximizing margin. Here, hyper-plane B has a classification error and A has classified all correctly. Therefore, the right hyper-plane is **A**.

- **Can we classify two classes (Scenario-4)?**: Below, I am unable to segregate the two classes using a straight line, as one of the stars lies in the territory of other(circle)class

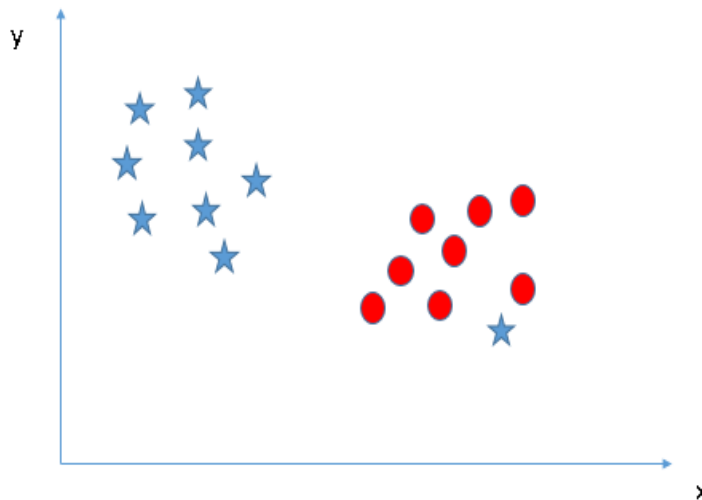


FIGURE 4.8 Scenario-4

- As I have already mentioned, one star at other end is like an outlier for star class. The SVM algorithm has a feature to ignore outliers and find the hyper-plane that has the maximum margin. Hence, we can say, SVM classification is robust to outliers.

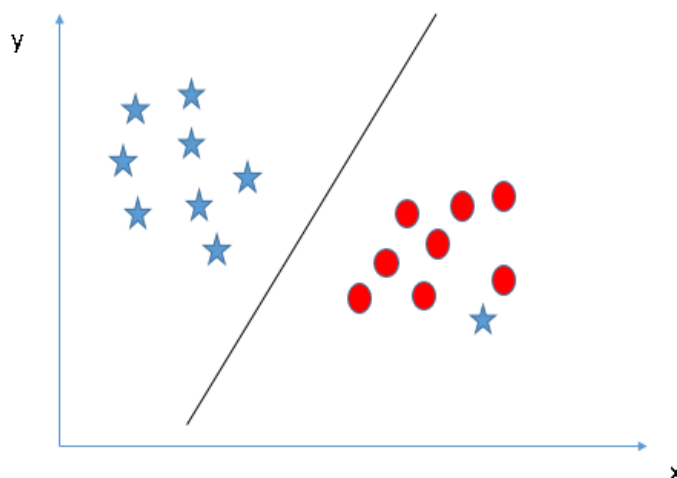


FIGURE 4.9 Scenario-4 with Hyper-plane

- **Find the hyper-plane to segregate to classes (Scenario-5):** In the scenario below, we can't have linear hyper-plane between the two classes, so how does SVM classify these two classes? Till now, we have only looked at the linear hyper-plane.

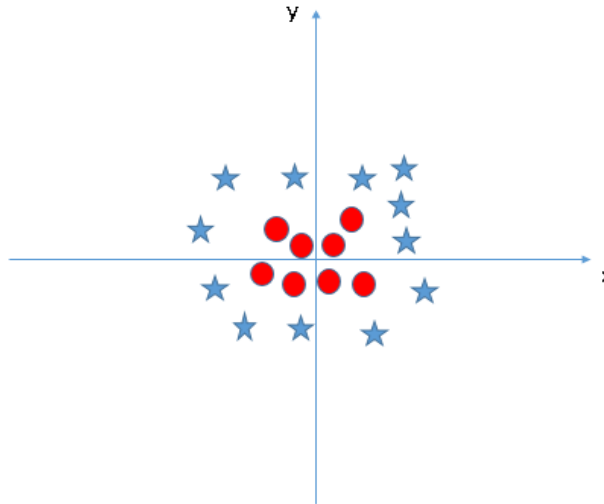


FIGURE 4.10 Scenario-5

- SVM can solve this problem. Easily! It solves this problem by introducing additional feature. Here, we will add a new feature $z = x^2 + y^2$. Now, let's plot the data points on axis x and z :

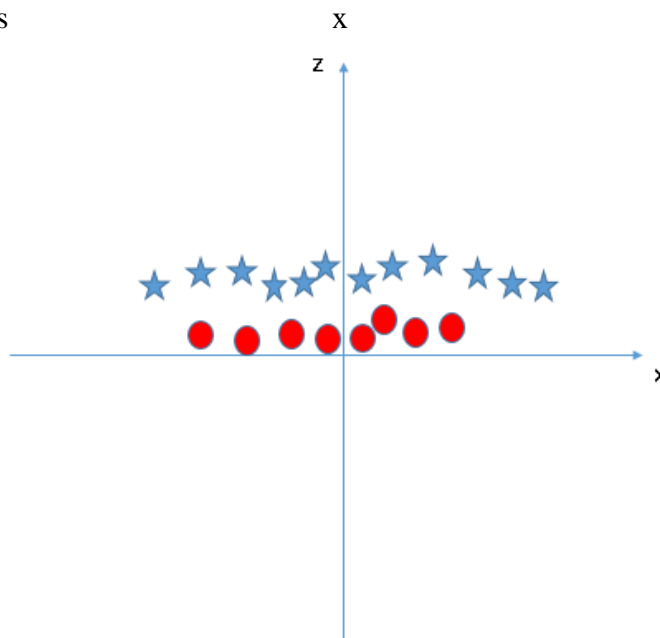


FIGURE 4.11 Graph With Additional feature

- In above plot, points to consider are:
 - All values for z would be positive always because z is the squared sum of both x and y

- In the original plot, red circles appear close to the origin of x and y axes, leading to lower value of z and star relatively away from the origin result to higher value of z.

In the SVM classifier, it is easy to have a linear hyper-plane between these two classes. But, another burning question which arises is, should we need to add this feature manually to have a hyper-plane. No, the SVM algorithm has a technique called the **kernel trick**. The SVM kernel is a function that takes low dimensional input space and transforms it to a higher dimensional space i.e. it converts not separable problem to separable problem. It is mostly useful in non-linear separation problem. Simply put, it does some extremely complex data transformations, then finds out the process to separate the data based on the labels or outputs you've defined.

When we look at the hyper-plane in original input space it looks like a circle:

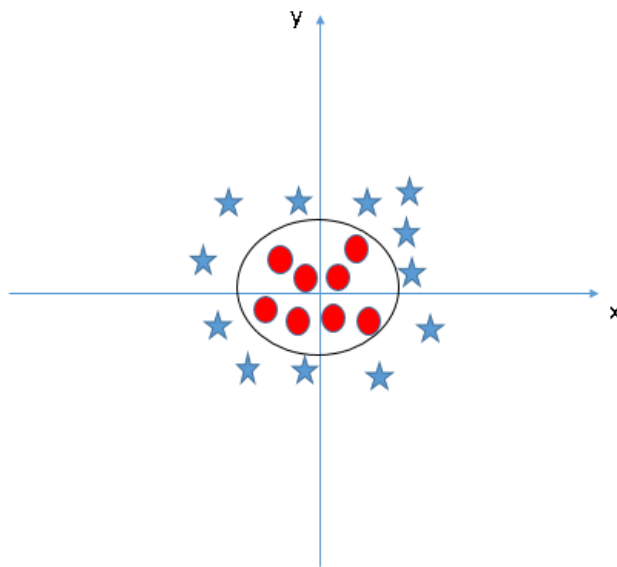


FIGURE 4.12 Hyper-plane in Input Space

4.1.4 Visualization

Visualizing the result of users' opinion mining on twitter using social network graph can play a crucial role in decision-making. One of the main components of the input file is the sentimental score of the users' opinion. This motivates us to develop a free and open source system that can take the opinion of users in raw text format and produce easy-to-interpret visualization of opinion mining and sentiment analysis result on a social network. With the concept of visualization, mining and analysis do play an important role as data mining is the idea of finding inferences by analyzing the data through patterns and those patterns can only be represented by different visualization techniques. It is the powerful way to explore data with presentable results. Techniques are Box plots, Histograms, Heatmaps, Charts, Treemaps. The test results show that our proposed system will be helpful to analyze and visualize the opinion of users.

CHAPTER 5

SOLUTION METHODOLOGY

5.1 DATA FLOW DIAGRAM

A data-flow diagram (DFD) is a way of representing a flow of a data of a process or a system (usually an information system). The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow, there are no decision rules and no loops. Specific operations based on the data can be represented by a flowchart.

There are several notations for displaying data-flow diagrams. For each data flow, at least one of the endpoints (source and / or destination) must exist in a process. The refined representation of a process can be done in another data-flow diagram, which subdivides this process into sub-processes.

DFD consists of processes, flows, warehouses, and terminators. There are several ways to view these DFD components.

Process

The process (function, transformation) is part of a system that transforms inputs to outputs. The symbol of a process is a circle, an oval, a rectangle or a rectangle with rounded corners (according to the type of notation).

Data Flow

Data flow (flow, dataflow) shows the transfer of information (sometimes also material) from one part of the system to another. The symbol of the flow is the arrow. The flow should have a name that determines what information (or what material) is being moved. Exceptions are flows where it is clear what information is transferred through the entities that are linked to these flows.

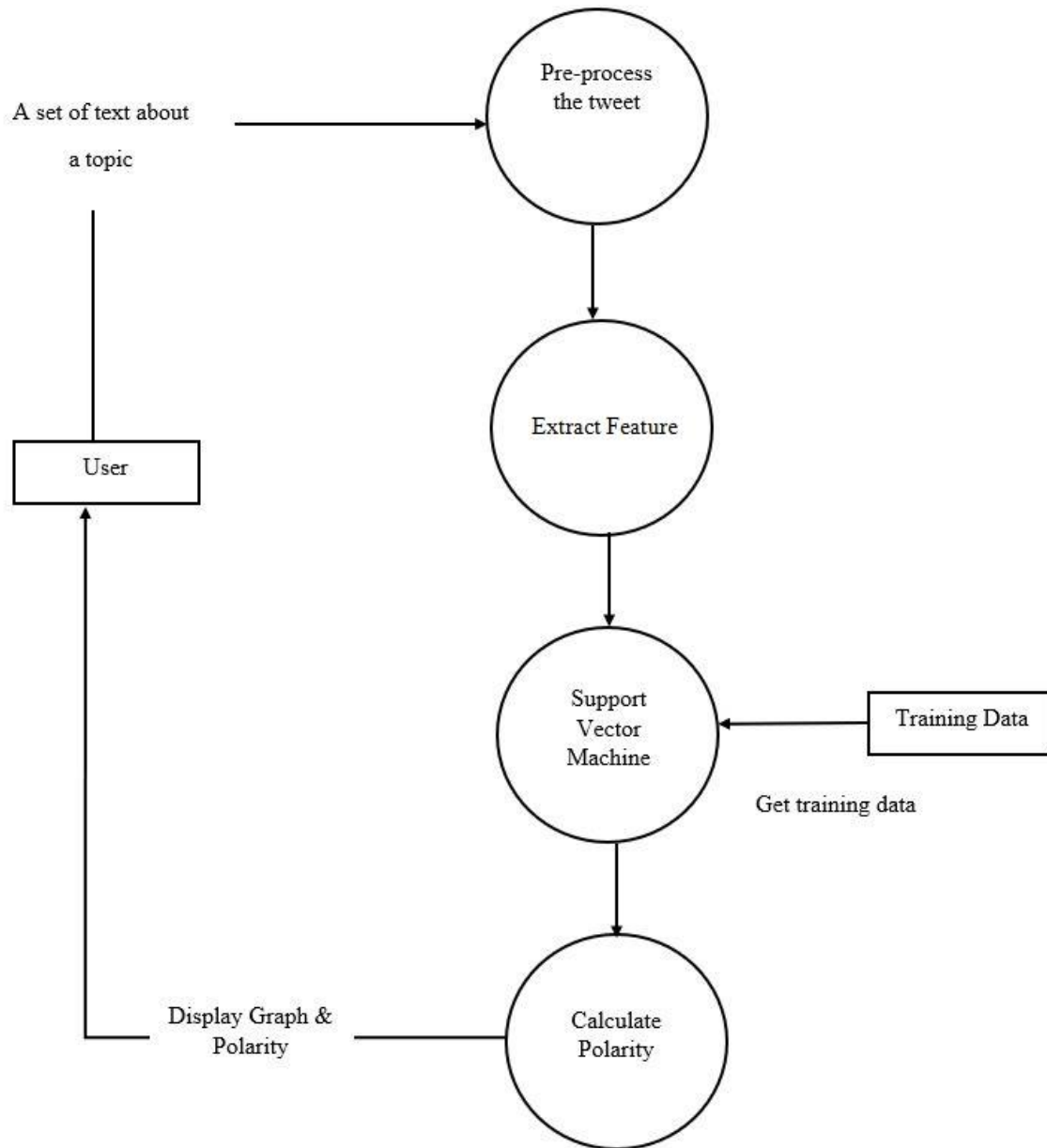


FIGURE 5.1 Data Flow Diagram

- Any one can tweet on their twitter accounts and those who follows him can comment on his tweet.
- If he need to know the public opinion about the tweet,he can click on the button”comment analyser” which is provided.When he click,system will extract all the comments and preprocess it.
- Then extract the feature by using pattern matching.
- Training data and the feature are provided to the svm to calculate the polarity.
- Finally public opinion whether it is positive or negative will be displayed.

5.2 FLOW CHART

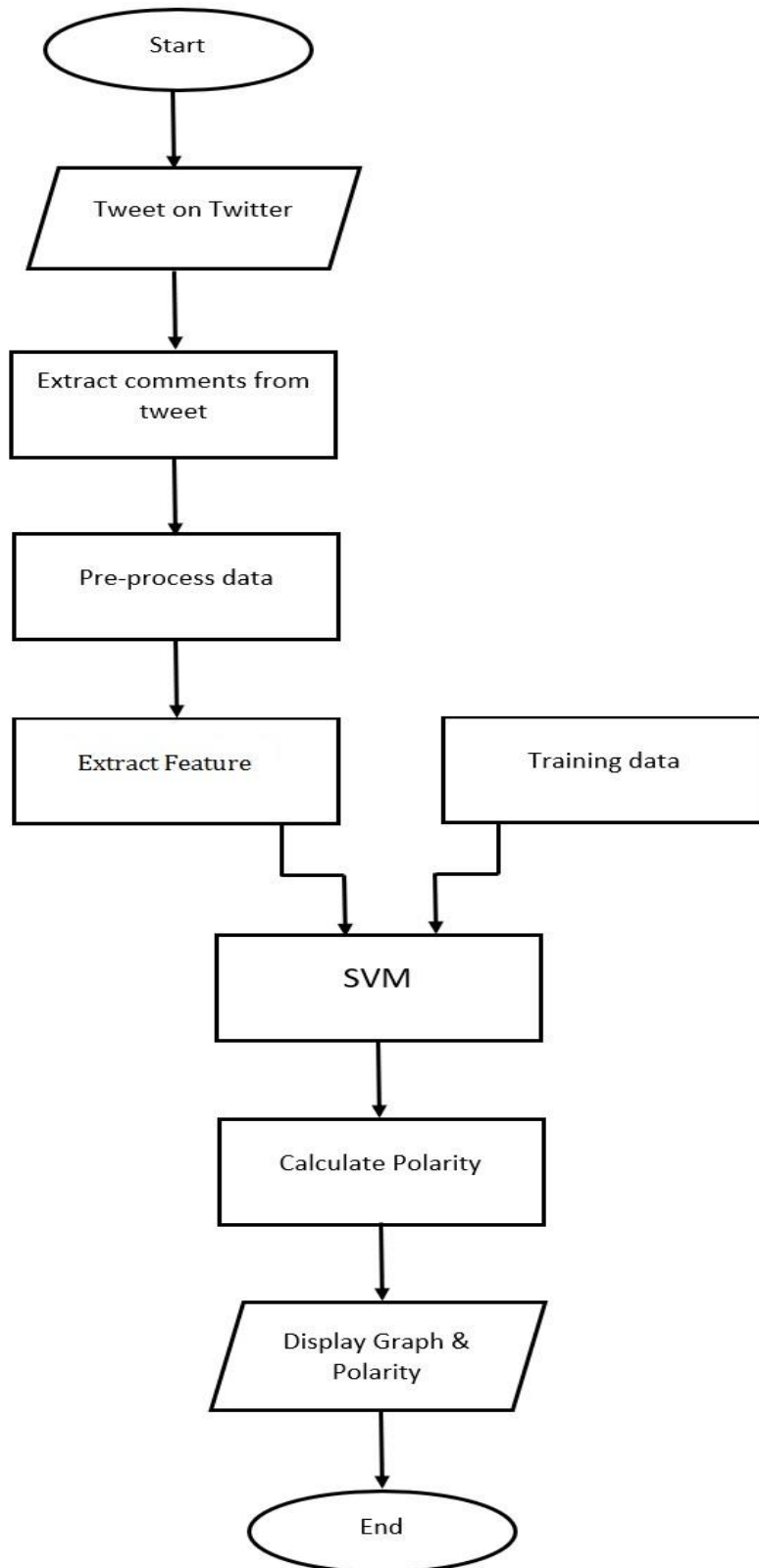


FIGURE 5.2 Flow Chart

CHAPTER 6

IMPLEMENTATION

6.1 TOOLS USED

The following tools are used to develop web application

Software and Hardware Requirements

6.1.1 Hardware Specification

Processor	: Core i3 or more
RAM	: 256 or more
Hard disk	: 4GB or more
System bus	:64 bit

6.1.2 software specification

Operating system	: Any windows OS above Windows 98
Platform	: Eclipse IDE
Language	:JAVA

6.2 Front End

Hypertext Markup Language (HTML) is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document. HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects such as interactive forms may be embedded into the rendered page. HTML provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes. Browsers do not display the HTML tags, but use

them to interpret the content of the page. HTML can embed programs written in a scripting language such as JavaScript, which affects the behavior and content of web pages. Inclusion of CSS defines the look and layout of content. The World Wide Web Consortium (W3C), former maintainer of the HTML and current maintainer of the CSS standards, has encouraged the use of CSS over explicit presentational HTML. Hypertext Markup Language, or HTML, is a programming language used to describe the structure of information on a web page.

Together, HTML, CSS, and JavaScript make up the essential building blocks of websites, with CSS controlling a page's appearance, and JavaScript programming its functionality. You can think of HTML as providing the bones of a web page, while CSS provides the skin, and JavaScript provides the brains. A web page can contain headings, paragraphs, images, videos, and many other types of data. Front-end developers use HTML elements to specify what kind of information each item on a web page contains — for instance, the “p” element indicates a paragraph. Developers also write HTML code to specify how different items relate to one another in the overall structure of the page. HTML plays a couple of significant roles in a web page. First, we use the structure created by our HTML code to reference, enhance, and manipulate elements on a web page using CSS and JavaScript. HTML lets us indicate the roles of different elements to search engines and other services that index the content and summarize it for other users.

6.3 Back End

Spring Boot is an open source Java-based framework used to create a micro Service. It is developed by Pivotal Team and is used to build stand-alone and production ready spring applications. Spring Boot provides a good platform for Java developers to develop a stand-alone and production-grade spring application that you can just run. You can get started with minimum configurations without the need for an entire Spring configuration setup.

Advantages

Spring Boot offers the following advantages to its developers –

- Easy to understand and develop spring applications

- Increases productivity
- Reduces the development time

Goals

Spring Boot is designed with the following goals –

- To avoid complex XML configuration in Spring
- To develop a production ready Spring applications in an easier way
- To reduce the development time and run the application independently
- Offer an easier way of getting started with the application

BENEFITS

- It provides a flexible way to configure Java Beans, XML configurations, and Database Transactions.
- It provides a powerful batch processing and manages REST endpoints.
- In Spring Boot, everything is auto configured; no manual configurations are needed.
- It offers annotation-based spring application
- Eases dependency management
- It includes Embedded Servlet Container

Spring Boot Auto Configuration automatically configure Spring application based on the JAR dependencies added in the project. if MySQL database is on the class path, but havenot configured any database connection, then Spring Boot auto configures an in-memory database.

Java is a general-purpose programming language that is class-based, object-oriented, and designed to have as few implementation dependencies as possible. It is intended to let application developers write once, run anywhere (WORA), meaning that compiled Java code can run on all platforms that support Java without the need for recompilation. Java applications are typically compiled to bytecode that can run on any Java virtual machine (JVM) regardless of the underlying computer architecture. The syntax of Java is similar to C and C++, but it has fewer low-level facilities than either of them.

java uses an automatic garbage collector to manage memory in the object lifecycle. The programmer determines when objects are created, and the Java runtime is responsible for recovering the memory once objects are no longer in use. Once no references to an object remain, the unreachable memory becomes eligible to be freed automatically by the garbage collector. Something similar to a memory leak may still occur if a programmer's code holds a reference to an object that is no longer needed, typically when objects that are no longer needed are stored in containers that are still in use. If methods for a non-existent object are called, a null pointer exception is thrown

.

One of the ideas behind Java's automatic memory management model is that programmers can be spared the burden of having to perform manual memory management. In some languages, memory for the creation of objects is implicitly allocated on the stack or explicitly allocated and deallocated from the heap. In the latter case, the responsibility of managing memory resides with the programmer. If the program does not deallocate an object, a memory leak occurs. If the program attempts to access or deallocate memory that has already been deallocated, the result is undefined and difficult to predict, and the program is likely to become unstable or crash.

6.4 Platform

Eclipse is an integrated development environment (IDE) used in computer programming.^[6] It contains a base workspace and an extensible plug-in system for customizing the environment. Eclipse is written mostly in Java and its primary use is for developing Java applications.

The initial codebase originated from IBM VisualAge.^[8] The Eclipse software development kit (SDK), which includes the Java development tools, is meant for Java developers. Users can extend its abilities by installing plug-ins written for the Eclipse Platform, such as development toolkits for other programming languages, and can write and contribute their own plug-in modules. Since the introduction of the OSGi implementation (Equinox) in version 3 of Eclipse, plug-ins can be plugged-stopped dynamically and are termed (OSGI) bundles. Eclipse software development kit (SDK) is free and open-source software.

Eclipse uses plug-ins to provide all the functionality within and on top of the run-time system. Its run-time system is based on Equinox, an implementation of the OSGi core framework specification.

In addition to allowing the Eclipse Platform to be extended using other programming languages, such as C and Python, the plug-in framework allows the Eclipse Platform to work with typesetting languages like LaTeX^[57] and networking applications such as telnet and database management systems. The plug-in architecture supports writing any desired extension to the environment, such as for configuration management. Java and CVS support is provided in the Eclipse SDK, with support for other version control systems provided by third-party plug-ins.

With the exception of a small run-time kernel, everything in Eclipse is a plug-in. Thus, every plug-in developed integrates with Eclipse in the same way as other plug-ins; in this respect, all features are "created equal".^[58] Eclipse provides plug-ins for a wide variety of features, some of which are from third parties using both free and commercial models. Examples of plug-ins include for Unified Modeling Language (UML), for Sequence and other UML diagrams, a plug-in for DB Explorer, and many more.

The Eclipse SDK includes the Eclipse Java development tools (JDT), offering an IDE with a built-in Java incremental compiler and a full model of the Java source files. This allows for advanced refactoring techniques and code analysis. The IDE also makes use of a *workspace*, in this case a set of metadata over a flat filesystem allowing external file modifications as long as the corresponding workspace *resource* is refreshed afterward. Eclipse implements the graphical control elements of the Java toolkit called Standard Widget Toolkit (SWT), whereas most Java applications use the Java standard Abstract Window Toolkit (AWT) or Swing. Eclipse's user interface also uses an intermediate graphical user interface layer called JFace, which simplifies the construction of applications based on SWT.

Server platform

Eclipse supports development for Tomcat, GlassFish and many other servers and is often capable of installing the required server (for development) directly from the IDE. It supports remote debugging, allowing a user to watch variables and step through the code of an application that is running on the attached server.

Web Tools Platform

The Eclipse Web Tools Platform (WTP) project is an extension of the Eclipse platform with tools for developing Web and Java EE applications. It includes source and graphical editors for a variety of languages, wizards and built-in applications to simplify development, and tools and APIs to support deploying, running, and testing apps.^[60]

Modeling platform

The Modeling project contains all the official projects of the Eclipse Foundation focusing on model-based development technologies. All are compatible with the Eclipse Modeling Framework created by IBM. Those projects are separated in several categories: Model Transformation, Model Development Tools, Concrete Syntax Development, Abstract Syntax Development, Technology and Research, and Amalgam.

Model Transformation projects uses Eclipse Modeling Framework (EMF) based models as an input and produce either a model or text as an output. Model to model transformation projects includes ATLAS Transformation Language (ATL), an open source transformation language and toolkit used to transform a given model or to generate a new model from a given EMF model. Model to text transformation projects contains Acceleo, an implementation of MOFM2T, a standard model to text language from the Object Management Group (OMG).

6.5 Database

MySQL is an open-source relational database management system (RDBMS). The abbreviation for Structured Query Language. MySQL is free and open-source software. MySQL is offered under two different editions: the open source MySQL Community Server and the proprietary Enterprise Server. MySQL Enterprise Server is differentiated by a series of proprietary extensions which install as server plugins, but otherwise shares the version numbering system and is built from the same code base.

A graphical user interface (GUI) is a type of interface that allows users to interact with electronic devices or programs through graphical icons and visual indicators such as secondary notation, as opposed to text-based interfaces, typed command labels or text navigation. GUIs are easier to learn than command-line interfaces (CLIs), which require commands to be typed on the keyboard.

Third-party proprietary and free graphical administration applications (or "front ends") are available that integrate with MySQL and enable users to work with database structure and data visually. **MySQL WorkBench** is a visual database design tool that integrates SQL development, administration, database design, creation and maintenance into a single integrated development environment for the MySQL database system. A command-line interface is a means of interacting with a computer program where the user issues commands to the program by typing in successive lines of text (command lines). MySQL ships with many command line tools, from which the main interface is the mysql client. MySQL Utilities is a set of utilities designed to perform common maintenance and administrative tasks. Originally included as part of the MySQL Workbench, the utilities are a stand-alone download available from Oracle.

CHAPTER 7

TESTING

7.1. Unit Testing

Unit testing is performed for testing modules against detailed design. Inputs to the process are usually compiled modules from the coding process. Each modules are assembled into a larger unit during the unit testing process. Testing has been performed on each phase of project design and coding. We carry out the testing of module interface to ensure the proper flow of information into and out of the program unit while testing. We make sure that the temporarily stored data maintains its integrity throughout the algorithm's execution by examining the local data structure. Finally, all error-handling paths are also tested.

7.2. System Testing

We usually perform system testing to find errors resulting from unanticipated interaction between the sub-system and system components. Software must be tested to detect and rectify all possible errors once the source code is generated before delivering it to the customers. For finding errors, series of test cases must be developed which ultimately uncover all the possibly existing errors. Different software techniques can be used for this process. These techniques provide systematic guidance for designing test that

- Exercise the internal logic of the software components,
- Exercise the input and output domains of a program to uncover errors in program function, behavior and performance.

We test the software using two methods:

White Box testing: Internal program logic is exercised using this test case design techniques.

Black Box testing: Software requirements are exercised using this test case design techniques.

Both techniques help in finding maximum number of errors with minimal effort and time.

7.3. Performance Testing

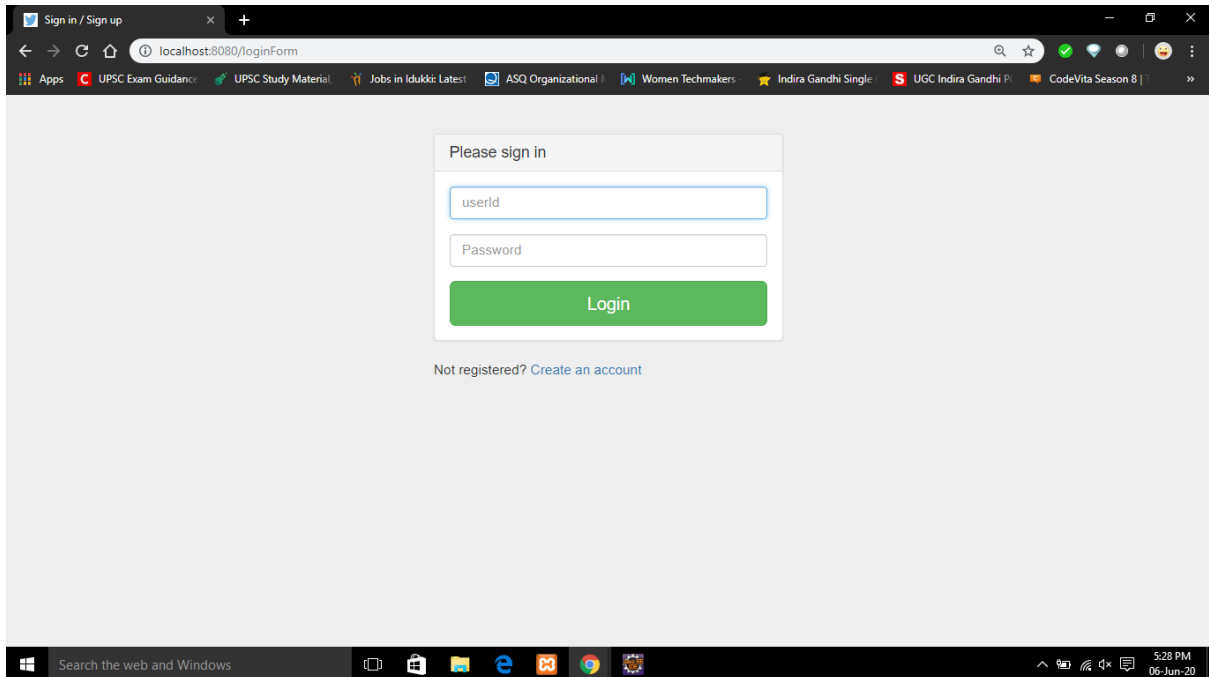
It is done to test the run-time performance of the software within the context of integrated system. These tests are carried out throughout the testing process. For example, the performance of individual module are accessed during white box testing under unit testing.

7.4. Verification and Validation

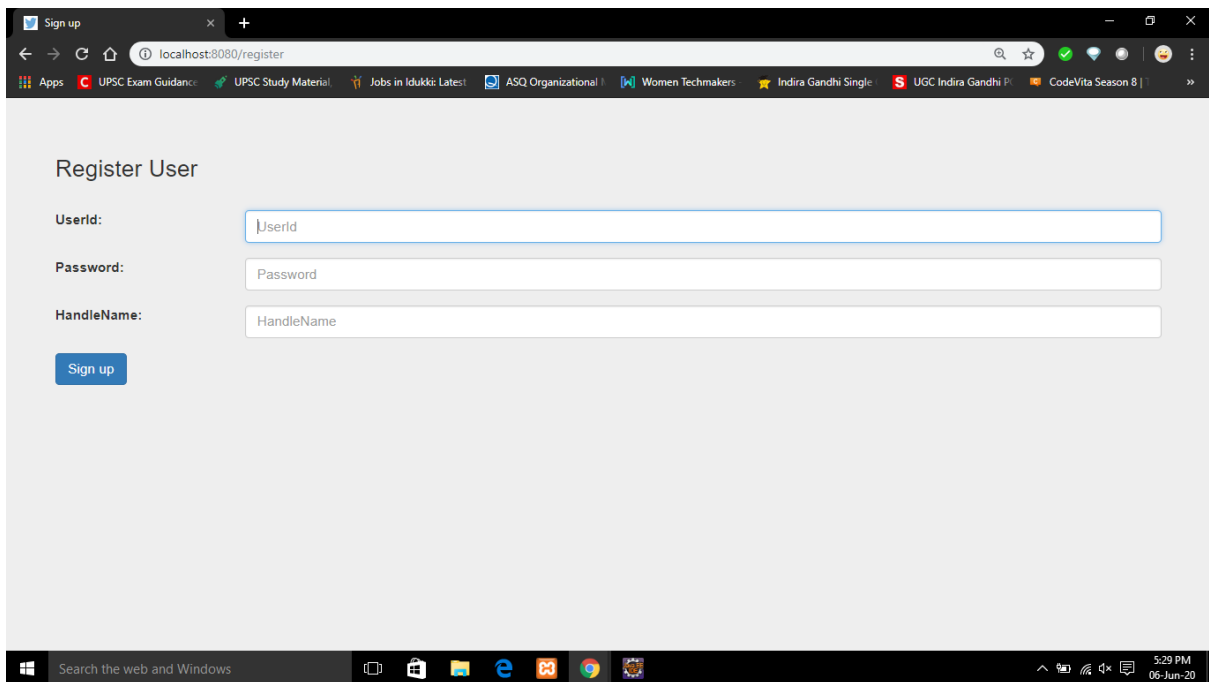
The testing process is a part of broader subject referring to verification and validation. We have to acknowledge the system specifications and try to meet the customer's requirements and for this sole purpose, we have to verify and validate the product to make sure everything is in place. Verification and validation are two different things. One is performed to ensure that the software correctly implements a specific functionality and other is done to ensure if the customer requirements are properly met or not by the end product. Verification is more like 'are we building the product right?' and validation is more like 'are we building the right product?'.

CHAPTER 8

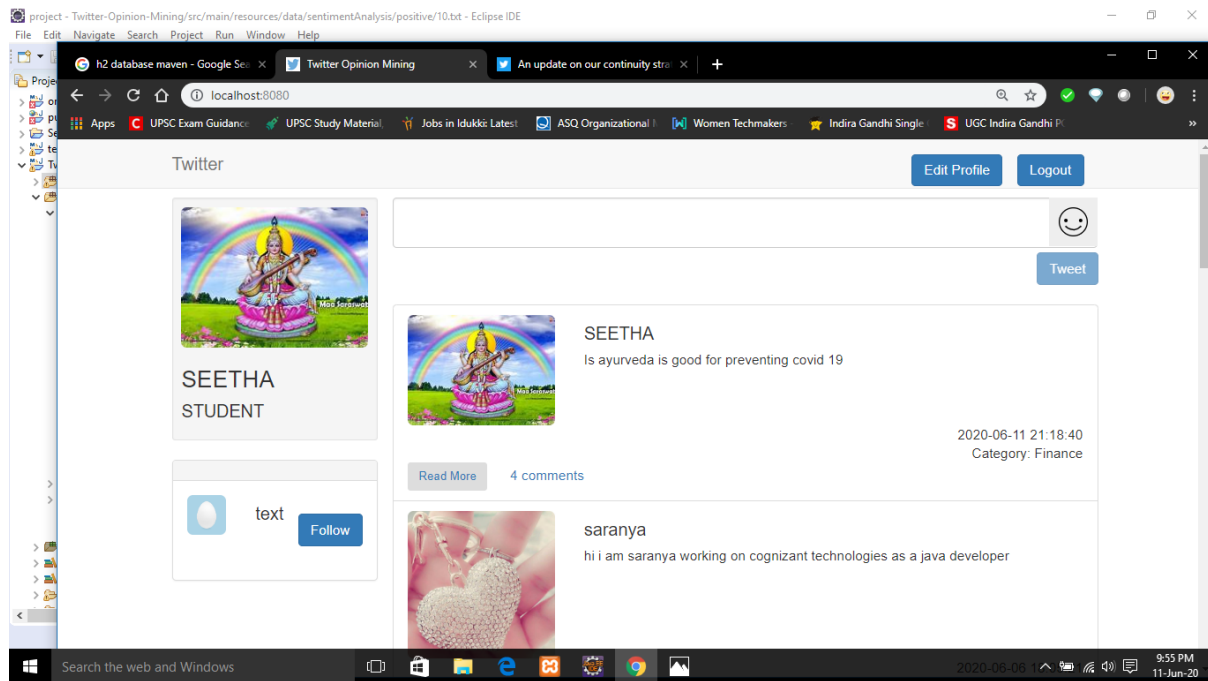
RESULT AND DISCUSSION



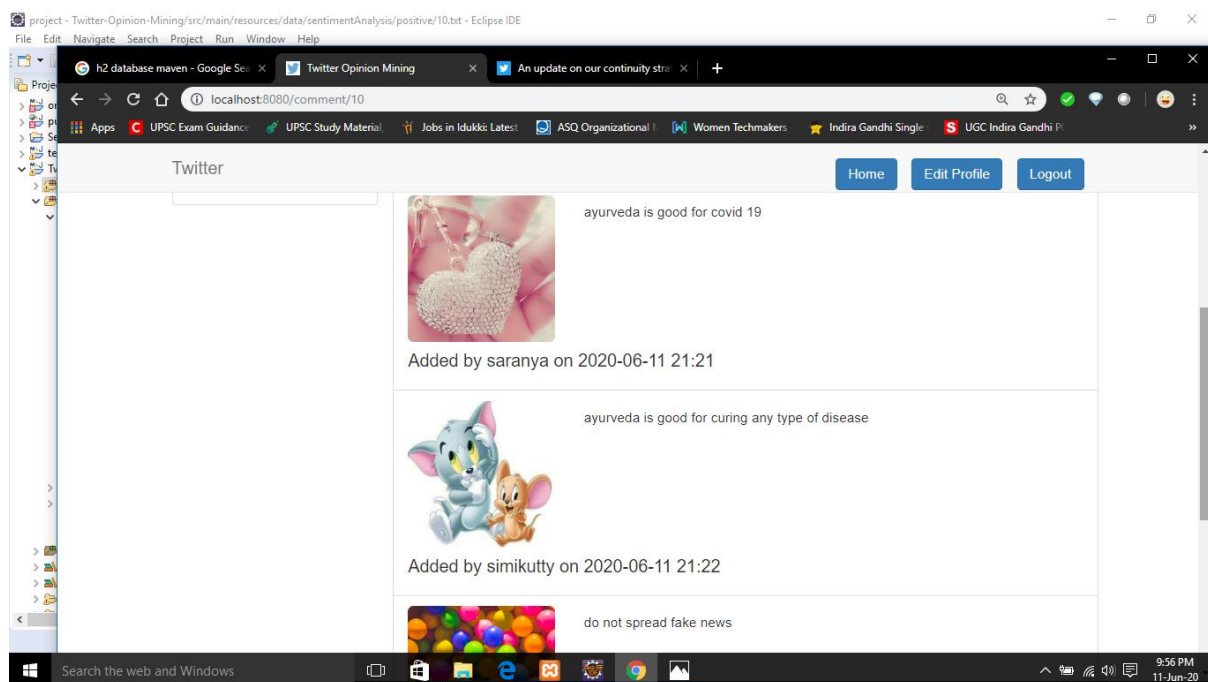
It is a home page.we can sign in or create an account.



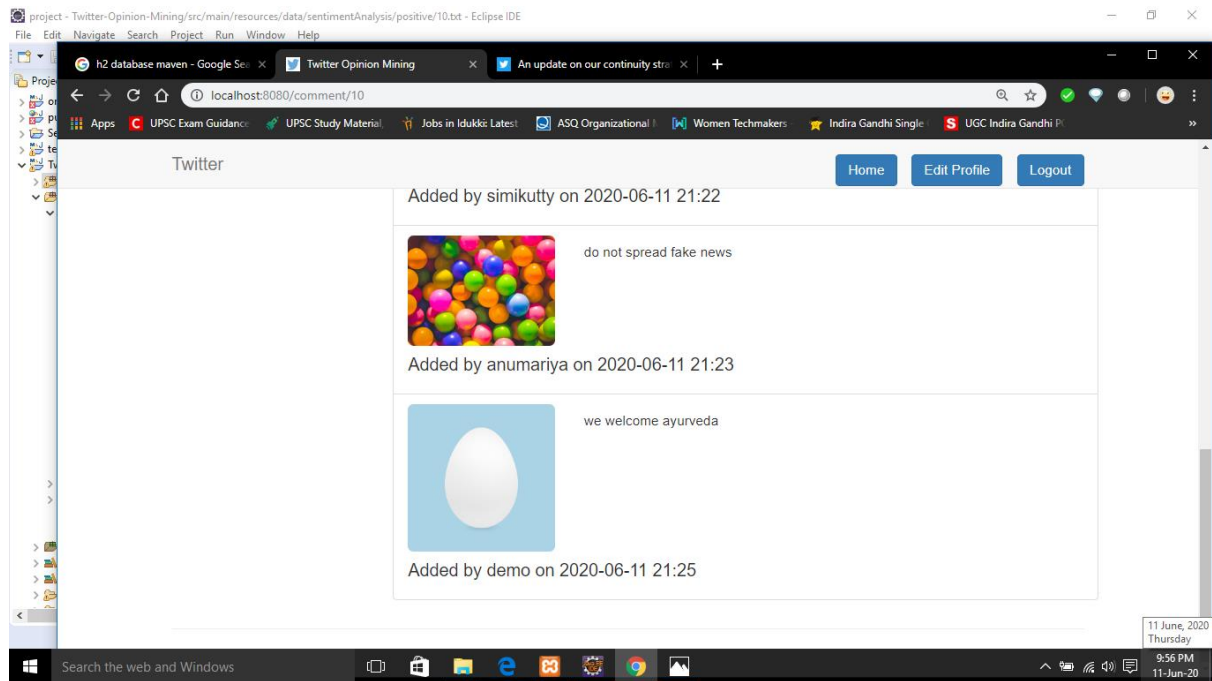
If we have no account then create an account and update the profile.



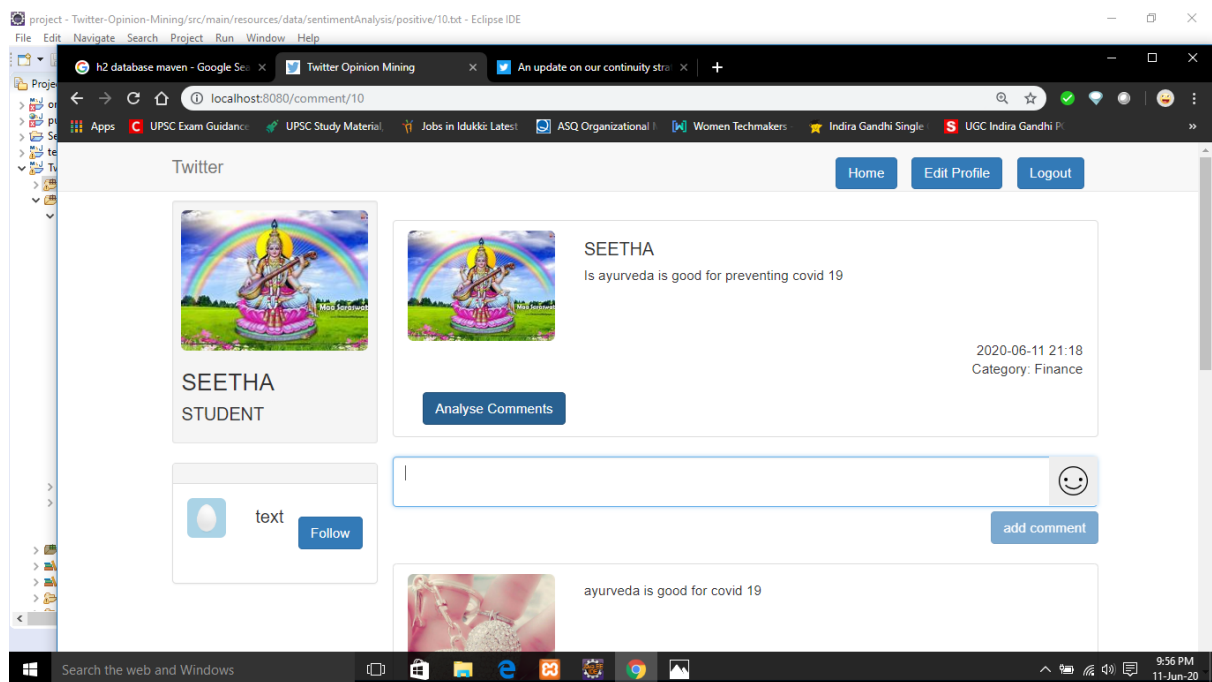
User Seetha is tweeting and followers comment on the tweet.



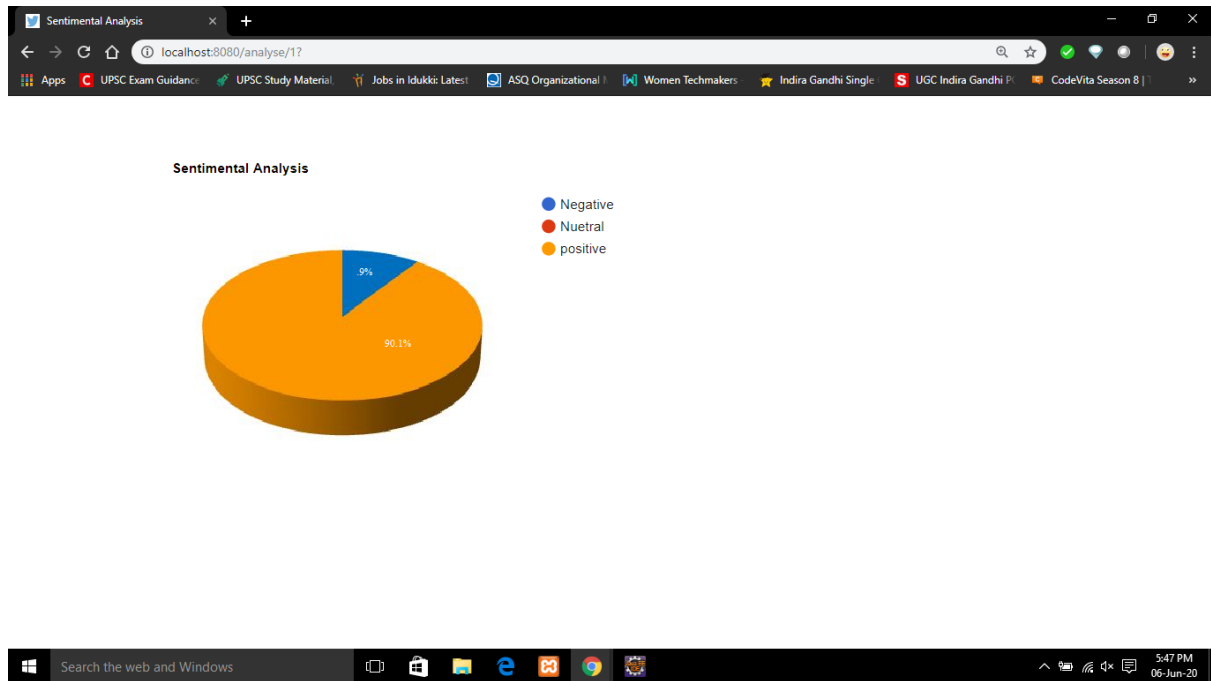
Comments of followers



Comments of followers



Click on the button “Analyse Comments” to see the public opinion.



Sentimental Analysis is shown in graph.

CHAPTER 9

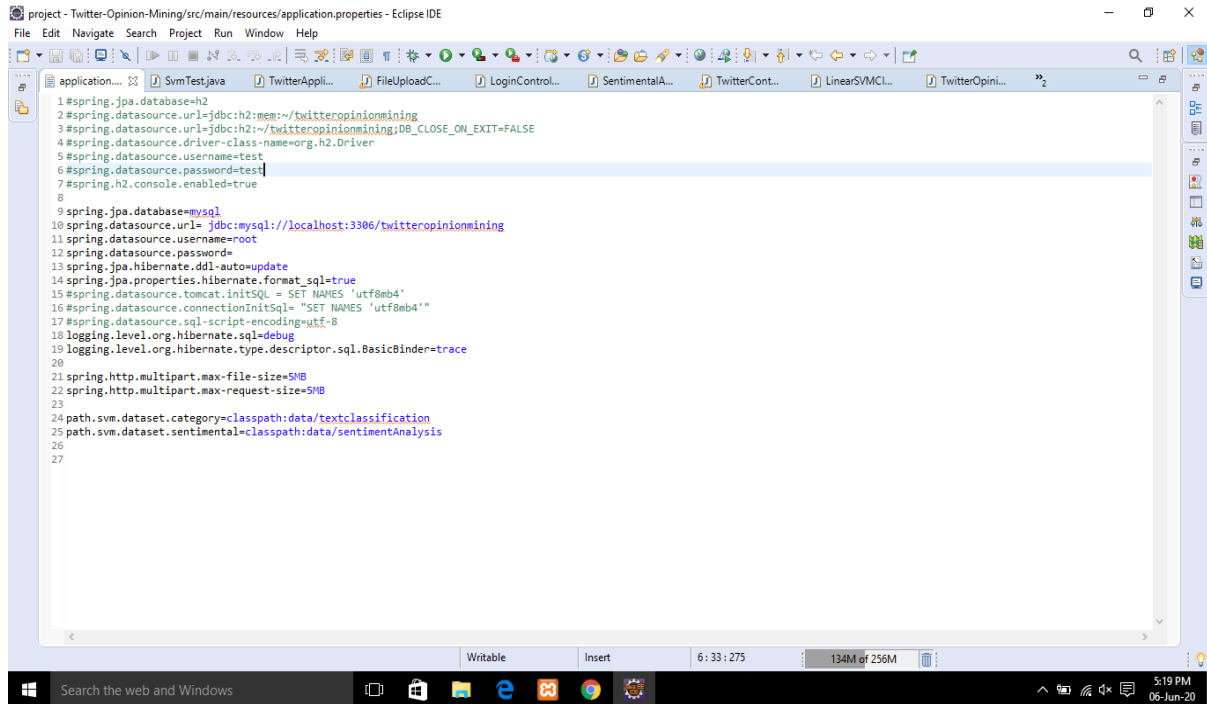
CONCLUSION

A novel research method that is different from existing work that mostly focus on classification of writer emotion analysis and long articles. We devise a model to analyze public opinion derived from reader emotion and classification mechanism which focuses on social media. In modern era, most of the peoples are depending on social media. They are very valuable and important resources for people to understand and study the changing world. Social media enables users to easily post their opinions and perspectives regarding certain issues. But it is difficult to understand the general impression of people in such issues. This is especially a problem for the tweets sentiment analysis. This paper aims at using text mining techniques to explore public opinion contained in social media by analyzing the reader's emotion towards pieces of short text. Sentiment classification is used to classify a text according to the sentimental polarities of positive and negative opinions. Pattern matching technique is used for the categorization of a text. Using SVM(Support Vector Machine) classifier. we combine a visualized analysis method for keywords that can provide a deeper understanding of opinions expressed on social media topics. It can classify public opinion, as well as analyze and visualize public opinion

REFERENCES

- Dawei Li , Yujia Zhang , and Cheng Li[1] “Mining Public Opinion on Transportation Systems Based on Social Media Data ”
- George Stylios, Dimitris Christodoulakis, Jeries Besharat, Maria-Alexandra Vonitsanou, Ioanis Kotrotsos, Athanasia Koumpouri and Sofia Stamou Patras University, Greece [2] “Public Opinion Mining for Governmental Decisions’
- B. Pang, L. Lee, and S. Vaithyanathan, “Thumbs up? : sentiment classification using machine learning techniques,” In Proceedings of the ACL02 Conference on Empirical Methods in Natural Language Processing, pp. 79-86, 2002.
- Y.-J. Tang and H.-H. Chen, “Mining sentiment words from microblogs for predicting writer- reader emotion transition,” In Proceedings of the 8th International Conference on Language Resources and Evaluation, pp. 1226–1229, 2012.
- “Sentiment Strength Detection in Short Informal Text”Mike Thelwall, Kevan Buckley, Georgios Paltoglou, and Di Cai Statistical Cybermetrics Research Group, School of Computing and Information Technology,University of Wolverhampton, Wulfruna Street, Wolverhampton WV1 1SB, UK.
- “Sentiment Analysis on Customer Feedback Data:Amazon Product Reviews” Pankaj, Prashant Pandey, Muskan, Nitasha Soni,Manav Rachna International Institute of Research and Studies,Faridabad, Haryana

APPENDIX

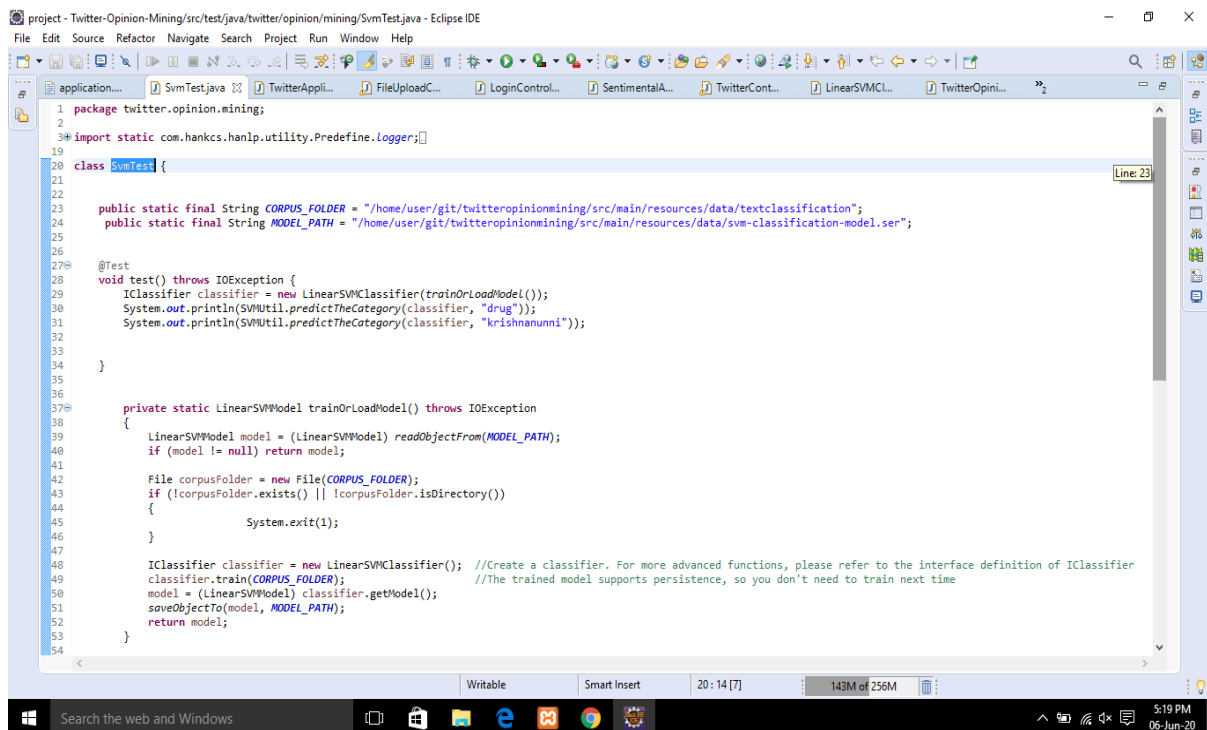


project - Twitter-Opinion-Mining/src/main/resources/application.properties - Eclipse IDE

```

1 #spring.jpa.database=h2
2 #spring.datasource.url=jdbc:h2:mem:~/twitteropinionmining
3 #spring.datasource.url=jdbc:h2:~/twitteropinionmining;DB_CLOSE_ON_EXIT=FALSE
4 #spring.datasource.driver-class-name=org.h2.Driver
5 #spring.datasource.username=test
6 #spring.datasource.password=test
7 #spring.h2.console.enabled=true
8
9 spring.jpa.database=mysql
10 spring.datasource.url=jdbc:mysql://localhost:3306/twitteropinionmining
11 spring.datasource.username=root
12 spring.datasource.password=
13 spring.jpa.hibernate.ddl-auto=update
14 spring.jpa.properties.hibernate.format_sql=true
15 #spring.datasource.tomcat.initSQL = SET NAMES 'utf8mb4'
16 #spring.datasource.connectionInitSql= "SET NAMES 'utf8mb4'"
17 #spring.datasource.sql-script-encoding=utf-8
18 logging.level.org.hibernate.sql=debug
19 logging.level.org.hibernate.type.descriptor.sql.BasicBinder=trace
20
21 spring.http.multipart.max-file-size=5MB
22 spring.http.multipart.max-request-size=5MB
23
24 path.svm.dataset.category=classpath:data/textclassification
25 path.svm.dataset.sentimental=classpath:data/sentimentAnalysis
26
27

```



project - Twitter-Opinion-Mining/src/test/java/twitter/opinion/mining/SvmTest.java - Eclipse IDE

```

1 package twitter.opinion.mining;
2
3 import static com.hankcs.hanlp.utility.Predefine.Logger;
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20 class SvmTest {
21
22
23     public static final String CORPUS_FOLDER = "/home/user/git/twitteropinionmining/src/main/resources/data/textclassification";
24     public static final String MODEL_PATH = "/home/user/git/twitteropinionmining/src/main/resources/data/svm-classification-model.ser";
25
26
27     @Test
28     void test() throws IOException {
29         IClassifier classifier = new LinearSVMClassifier(trainOrLoadModel());
30         System.out.println(SVMUtil.predictTheCategory(classifier, "drug"));
31         System.out.println(SVMUtil.predictTheCategory(classifier, "krishnanunni"));
32
33     }
34
35
36
37     private static LinearSVMModel trainOrLoadModel() throws IOException
38     {
39         LinearSVMModel model = (LinearSVMModel) readObjectFrom(MODEL_PATH);
40         if (model != null) return model;
41
42         File corpusFolder = new File(CORPUS_FOLDER);
43         if (!corpusFolder.exists() || !corpusFolder.isDirectory())
44         {
45             System.exit(1);
46         }
47
48         IClassifier classifier = new LinearSVMClassifier(); //Create a classifier. For more advanced functions, please refer to the interface definition of IClassifier
49         classifier.train(CORPUS_FOLDER); //The trained model supports persistence, so you don't need to train next time
50         model = (LinearSVMModel) classifier.getModel();
51         saveObjectTo(model, MODEL_PATH);
52         return model;
53     }
54

```

```

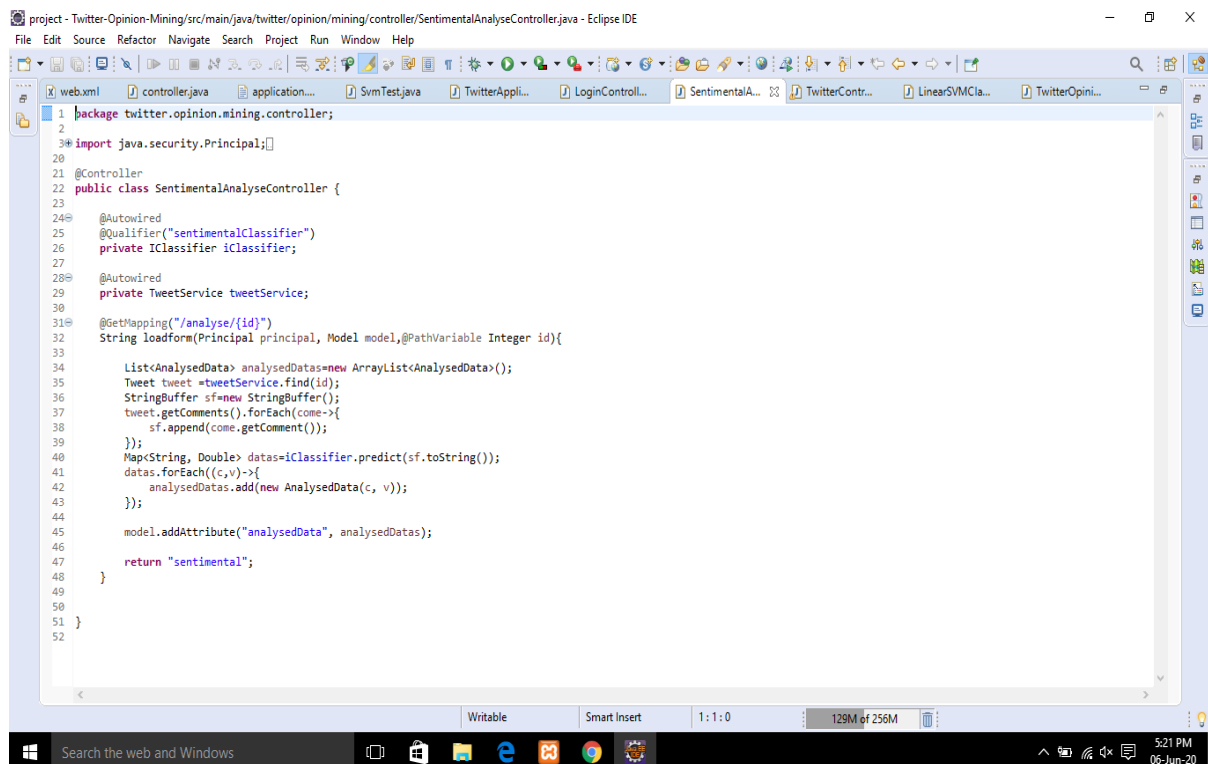
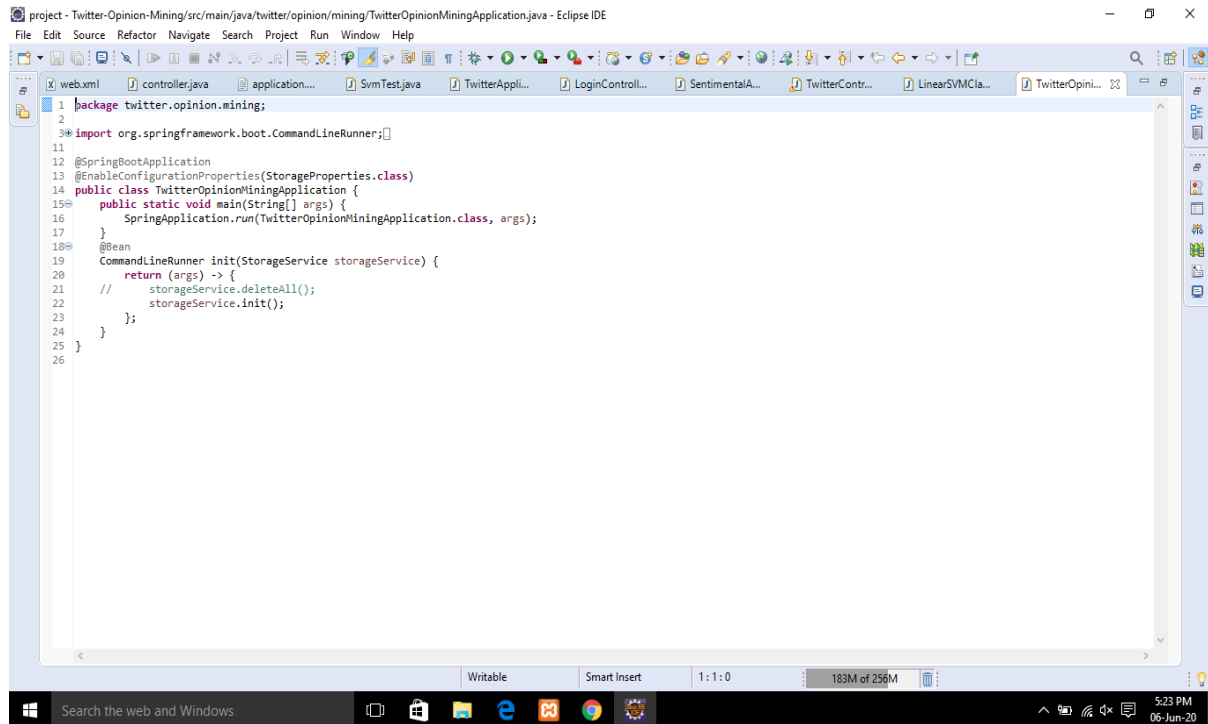
54
55
56 public static boolean saveObjectTo(Object o, String path)
57 {
58     try
59     {
60         ObjectOutputStream oos = new ObjectOutputStream(new FileOutputStream(path));
61         oos.writeObject(o);
62         oos.close();
63     }
64     catch (IOException e)
65     {
66         Logger.warning("Saving object" + o + "To" + path + "An exception occurred" + e);
67         return false;
68     }
69
70     return true;
71 }
72
73
74 public static Object readObjectFrom(String path)
75 {
76     ObjectInputStream ois = null;
77     try
78     {
79         ois = new ObjectInputStream(new FileInputStream(path));
80         Object o = ois.readObject();
81         ois.close();
82         return o;
83     }
84     catch (Exception e)
85     {
86         Logger.warning("In from" + path + "An exception occurred while reading the object" + e);
87     }
88
89     return null;
90 }
91
92

```

```

1 package twitter.opinion.mining.controller;
2
3 import org.springframework.stereotype.Controller;
4
5
6 @Controller
7 public class LoginController {
8     @RequestMapping("/loginForm")
9     String loginForm(){
10         return "login";
11     }
12 }
13

```



```

1
2
3
4 import java.security.Principal;
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47 @Controller
48 // @SessionAttributes(value = {"userinfo"})
49 public class TwitterController {
50
51     public static final Logger log = LoggerFactory.getLogger(TwitterController.class);
52
53     private TweetService tweetService;
54     private UserService userService;
55     private TwitterUserService userDetailsService;
56     private FileSystemStorageService fileSystemStorageService;
57     private IClassifier iClassifier;
58     private CommentRepository commentRepository;
59
60
61     @Autowired
62     public TwitterController(TweetService tweetService,
63                             UserService userService,
64                             TwitterUserService userDetailsService,
65                             FileSystemStorageService fileSystemStorageService,
66                             @Qualifier("categoryClassifier") IClassifier iClassifier,
67                             CommentRepository commentRepository) {
68
69         this.tweetService = tweetService;
70         this.userService = userService;
71         this.userDetailsService = userDetailsService;
72         this.fileSystemStorageService = fileSystemStorageService;
73         this.iClassifier = iClassifier;
74         this.commentRepository = commentRepository;
75
76     }
77
78     @GetMapping(value = "/")
79     String timeline(Principal principal, Model model) {
80         model.addAttribute("tweetForm", new TweetForm()); //attribute can be omitted.
81     }
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113

```

```

114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2
```

```

110
111     try {
112         tweetService.save(tweet);
113     } catch (Exception e) {
114         Set<String> err = new HashSet<>();
115         err.add("an error occurred. try again.");
116         model.addAttribute("errors", err);
117         Log.info(e.toString());
118         e.printStackTrace();
119         //return timeline(principal,model);
120         return "redirect:/";
121     }
122
123     return "redirect:/";
124 }
125
126 @GetMapping(value = "/comment/{id}")
127 String tweet(Principal principal, Model model, @PathVariable Integer id) {
128
129
130     User loginUser = Util.getLoginuserFromPrincipal(principal);
131     model.addAttribute("userInfo", loginUser);
132     Tweet tweet = tweetService.find(id);
133     model.addAttribute("tweet", tweet);
134     model.addAttribute("commentForm", new CommentForm());
135     model.addAttribute("recommend", userService.getUnFollowing10Users(loginUser, this));
136
137     Log.info("util.noicon: "+Util.getNoIcon());
138
139
140     return "comment";
141 }
142
143
144 @PostMapping("/comment")
145 public String addComment(Principal principal, CommentForm commentForm, BindingResult bindingResult) {
146     if (bindingResult.hasErrors()) {
147         Log.info("There was a problem adding a new comment.");
148     } else {

```

```

161
162 //register
163
164 @GetMapping(value = "/register")
165 String registerPage(Model model) {
166     model.addAttribute("registerForm", new RegisterForm());
167     return "register";
168 }
169
170 @PostMapping(value = "/register")
171 String register(@Validated RegisterForm form, BindingResult bindingResult, Model model) {
172     if (bindingResult.hasErrors()) {
173         Log.info("user:" + form.getUserId());
174         Log.info("pass:" + form.getPassword());
175         Log.info("scr:" + form.getScreenName());
176         Set<String> err = new HashSet<>();
177         bindingResult.getAllErrors().forEach(e -> err.add(e.getDefaultMessage()));
178         model.addAttribute("errors", err);
179         return "register";
180     }
181
182     Log.info("user:" + form.getUserId());
183     Log.info("pass:" + form.getPassword());
184     Log.info("scr:" + form.getScreenName());
185
186     BCryptPasswordEncoder encoder = new BCryptPasswordEncoder();
187     User user = new User(form.getUserId(), encoder.encode(form.getPassword()), form.getScreenName());
188     try {
189         userService.create(user);
190     } catch (UserIdAlreadyExistsException e) {
191         Set<String> errors = new HashSet<>();
192         errors.add(e.getMessage());
193         model.addAttribute("errors", errors);
194         return "register";
195     } catch (Exception e) {
196
197         Set<String> errors = new HashSet<>();
198         errors.add("unexpected error occurred. try again.");
199         model.addAttribute("errors", errors);

```

```

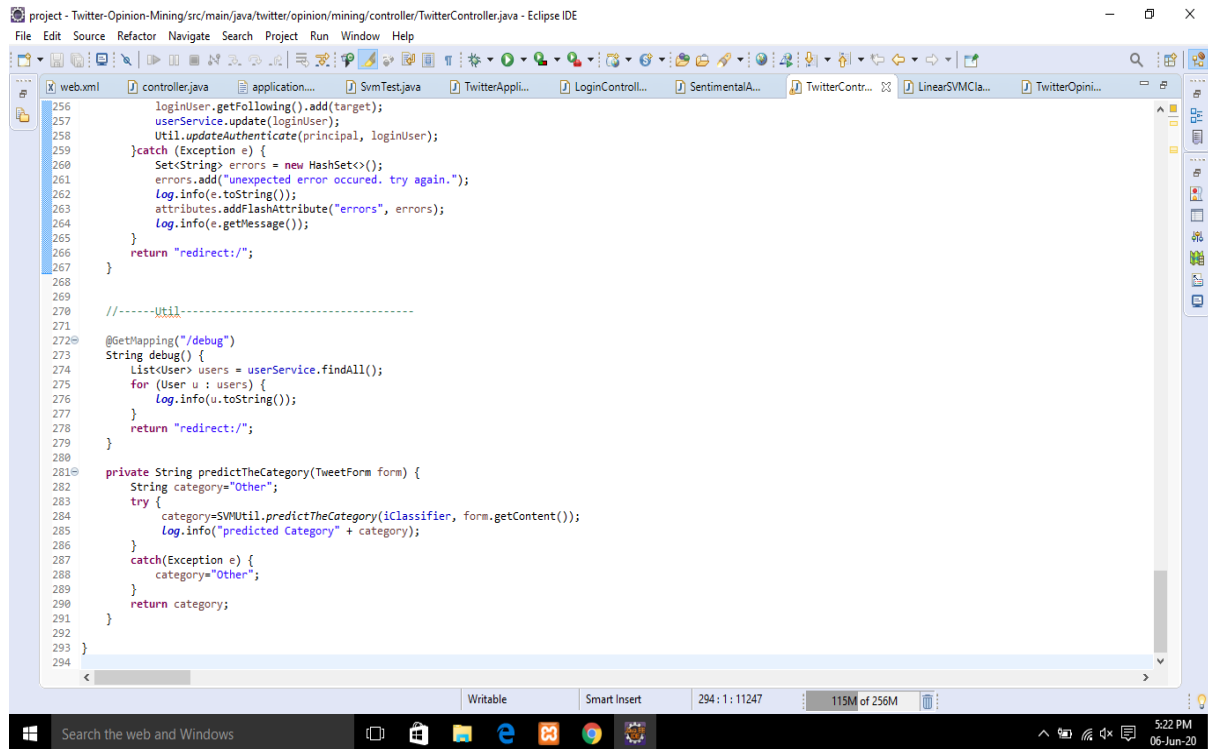
196 Set<String> errors = new HashSet<>();
197 errors.add("unexpected error occurred. try again.");
198 model.addAttribute("errors", errors);
199
200 Log.info(e.toString());
201 return "register";
202 }
203
204 return "redirect:/loginForm";
205 }
206
207 @GetMapping("/update")
208 String updateUserPage(Model model) {
209     model.addAttribute("userForm", new UserForm());
210     //model.addAttribute("uploadForm", new UploadFileForm());
211     return "mypage";
212 }
213
214 @PostMapping("/update")
215 String updateUserData(Principal principal, @Validated UserForm form, BindingResult bindingResult,
216                     Model model) {
217     if (bindingResult.hasErrors()) {
218         Set<String> err = new HashSet<>();
219         bindingResult.getAllErrors().forEach(e -> err.add(e.getDefaultMessage()));
220         model.addAttribute("errors", err);
221         return updateUserPage(model);
222         //return "mypage";
223     }
224
225     try {
226         User newUser = userService.find(Util.getLoginuserFromPrincipal(principal).getUserid());
227         if (!Objects.equals(form.getScreenName(), ""))
228             newUser.setScreenName(form.getScreenName());
229         if (!Objects.equals(form.getBiography(), ""))
230             newUser.setBiography(form.getBiography());
231         userService.update(newUser);
232
233         Util.updateAuthenticate(principal, newUser);
234
235         model.addAttribute("userinfo", newUser);
236     } catch (UserIdNotFoundException e) {
237         Set<String> errors = new HashSet<>();
238         errors.add(e.getMessage());
239         model.addAttribute("errors", errors);
240         return "mypage";
241     } catch (Exception e) {
242         Set<String> errors = new HashSet<>();
243         errors.add("unexpected error occurred. try again.");
244         model.addAttribute("errors", errors);
245         Log.info(e.getMessage());
246         return "mypage";
247     }
248     return "redirect:/";
249 }
250
251 @PostMapping(value = "/follow/{userid}")
252 String follow(Principal principal, @PathVariable("userid") String userid, RedirectAttributes attributes){
253     User loginuser=Util.getLoginuserFromPrincipal(principal);
254     try {
255         User target = userService.find(userid);

```

```

256         User target = userService.find(userid);
257     } catch (Exception e) {
258         Set<String> errors = new HashSet<>();
259         errors.add("unexpected error occurred. try again.");
260         model.addAttribute("errors", errors);
261         Log.info(e.getMessage());
262         return "mypage";
263     }
264     return "redirect:/";
265 }
266
267 @PostMapping(value = "/follow/{userid}")
268 String follow(Principal principal, @PathVariable("userid") String userid, RedirectAttributes attributes){
269     User loginuser=Util.getLoginuserFromPrincipal(principal);
270     try {
271         User target = userService.find(userid);
272     } catch (Exception e) {
273         Set<String> errors = new HashSet<>();
274         errors.add("unexpected error occurred. try again.");
275         model.addAttribute("errors", errors);
276         Log.info(e.getMessage());
277         return "mypage";
278     }
279     return "redirect:/";
280 }
281
282 @PostMapping(value = "/follow/{userid}")
283 String follow(Principal principal, @PathVariable("userid") String userid, RedirectAttributes attributes){
284     User loginuser=Util.getLoginuserFromPrincipal(principal);
285     try {
286         User target = userService.find(userid);
287     } catch (Exception e) {
288         Set<String> errors = new HashSet<>();
289         errors.add("unexpected error occurred. try again.");
290         model.addAttribute("errors", errors);
291         Log.info(e.getMessage());
292         return "mypage";
293     }
294     return "redirect:/";
295 }

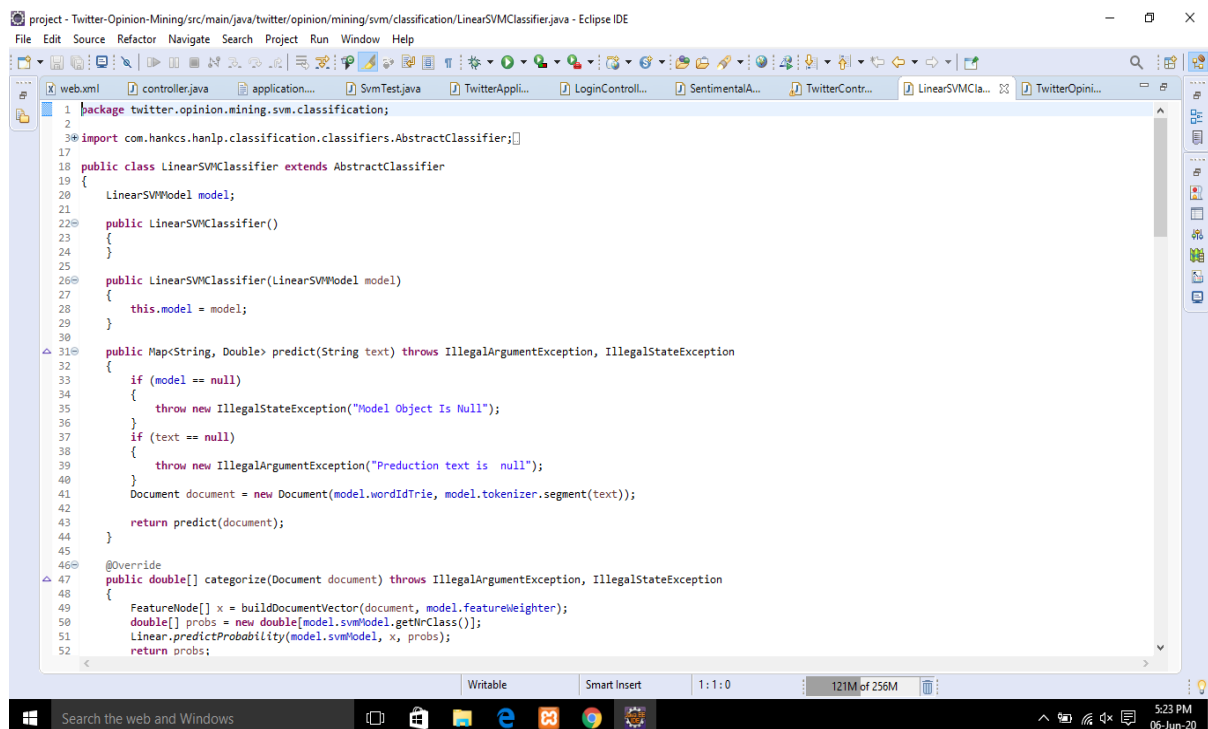
```



```

256 loginUser.getFollowing().add(target);
257 userService.update(loginUser);
258 Util.updateAuthenticate(principal, loginUser);
259 }catch (Exception e) {
260     Set<String> errors = new HashSet<>();
261     errors.add("unexpected error occurred. try again.");
262     Log.info(e.toString());
263     attributes.addFlashAttribute("errors", errors);
264     Log.info(e.getMessage());
265 }
266 return "redirect:/";
267 }
268
269 //-----Util-----
270
271 @GetMapping("/debug")
272 String debug() {
273     List<User> users = userService.findAll();
274     for (User u : users) {
275         Log.info(u.toString());
276     }
277     return "redirect:/";
278 }
279
280 private String predictTheCategory(TweetForm form) {
281     String category="Other";
282     try {
283         category=SVMUtil.predictTheCategory(iClassifier, form.getContent());
284         Log.info("predicted Category" + category);
285     }
286     catch (Exception e) {
287         category="Other";
288     }
289     return category;
290 }
291
292 }
293
294

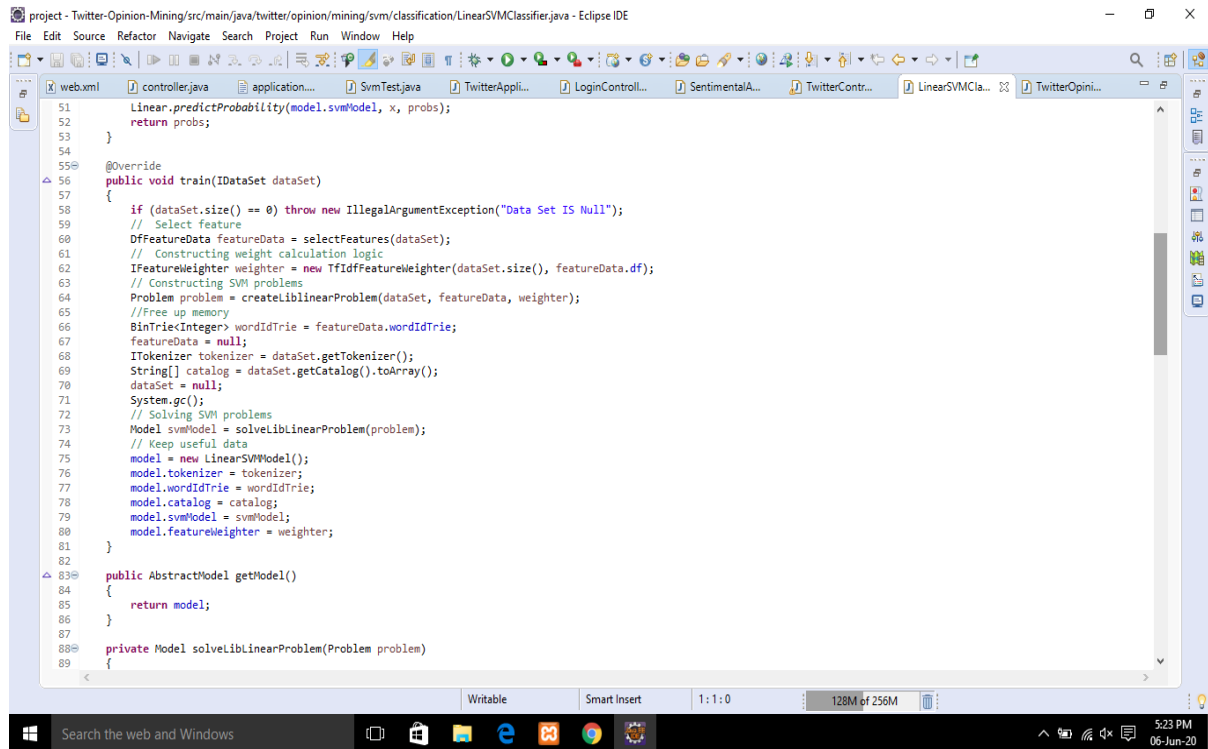
```



```

1 package twitter.opinion.mining.svm.classification;
2
3 import com.hankcs.hanlp.classification.classifiers.AbstractClassifier;
4
5 public class LinearSVMClassifier extends AbstractClassifier
6 {
7     LinearSVMModel model;
8
9     public LinearSVMClassifier()
10     {
11     }
12
13     public LinearSVMClassifier(LinearSVMModel model)
14     {
15         this.model = model;
16     }
17
18     public Map<String, Double> predict(String text) throws IllegalArgumentException, IllegalStateException
19     {
20         if (model == null)
21         {
22             throw new IllegalStateException("Model Object Is Null");
23         }
24         if (text == null)
25         {
26             throw new IllegalArgumentException("Preduction text is null");
27         }
28         Document document = new Document(model.wordIdTrie, model.tokenizer.segment(text));
29
30         return predict(document);
31     }
32
33     @Override
34     public double[] categorize(Document document) throws IllegalArgumentException, IllegalStateException
35     {
36         FeatureNode[] x = buildDocumentVector(document, model.featureWeighter);
37         double[] probs = new double[model.svmModel.getClass().getNbClasses()];
38         Linear.predictProbability(model.svmModel, x, probs);
39         return probs;
40     }
41
42 }
43
44

```



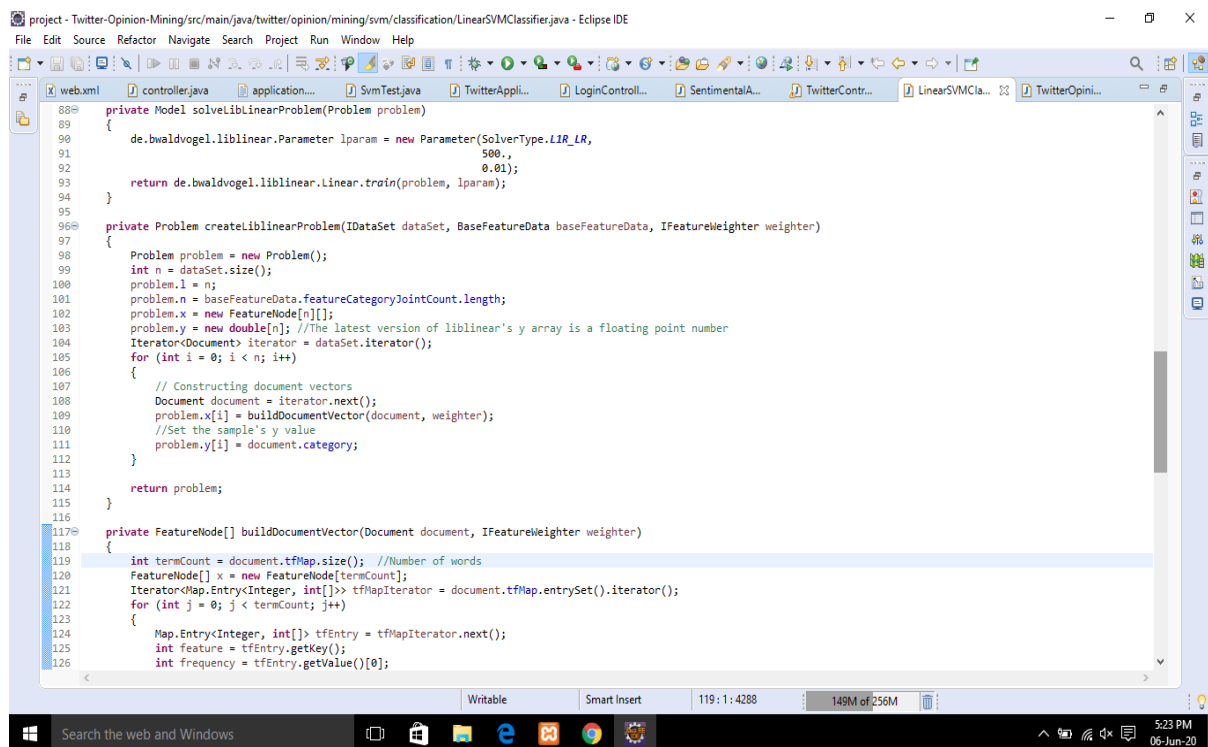
```

project - Twitter-Opinion-Mining/src/main/java/twitter/opinion/mining/svm/classification/LinearSVMClassifier.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

web.xml controller.java application... SvmTest.java TwitterAppli... LoginControll... SentimentalA... TwitterContr... LinearSVMCla... TwitterOpini...

51 Linear.predictProbability(model.svmModel, x, probs);
52 return probs;
53 }
54
55 @Override
56 public void train(IDataset dataSet)
57 {
58     if (dataSet.size() == 0) throw new IllegalArgumentException("Data Set IS Null");
59     // Select feature
60     DfFeatureData featureData = selectFeatures(dataSet);
61     // Constructing weight calculation logic
62     IFeatureWeighter weighter = new TfIdfFeatureWeighter(dataSet.size(), featureData.df);
63     // Constructing SVM problems
64     Problem problem = createLibLinearProblem(dataSet, featureData, weighter);
65     //Free up memory
66     BinTrie<Integer> wordIdTrie = featureData.wordIdTrie;
67     featureData = null;
68     ITokenizer tokenizer = dataSet.getTokenizer();
69     String[] catalog = dataSet.getCatalog().toArray();
70     dataSet = null;
71     System.gc();
72     // Solving SVM problems
73     Model svmModel = solveLibLinearProblem(problem);
74     // Keep useful data
75     model = new LinearSVMModel();
76     model.tokenizer = tokenizer;
77     model.wordIdTrie = wordIdTrie;
78     model.catalog = catalog;
79     model.svmModel = svmModel;
80     model.featureWeighter = weighter;
81 }
82
83 public AbstractModel getModel()
84 {
85     return model;
86 }
87
88 private Model solveLibLinearProblem(Problem problem)
89 {

```



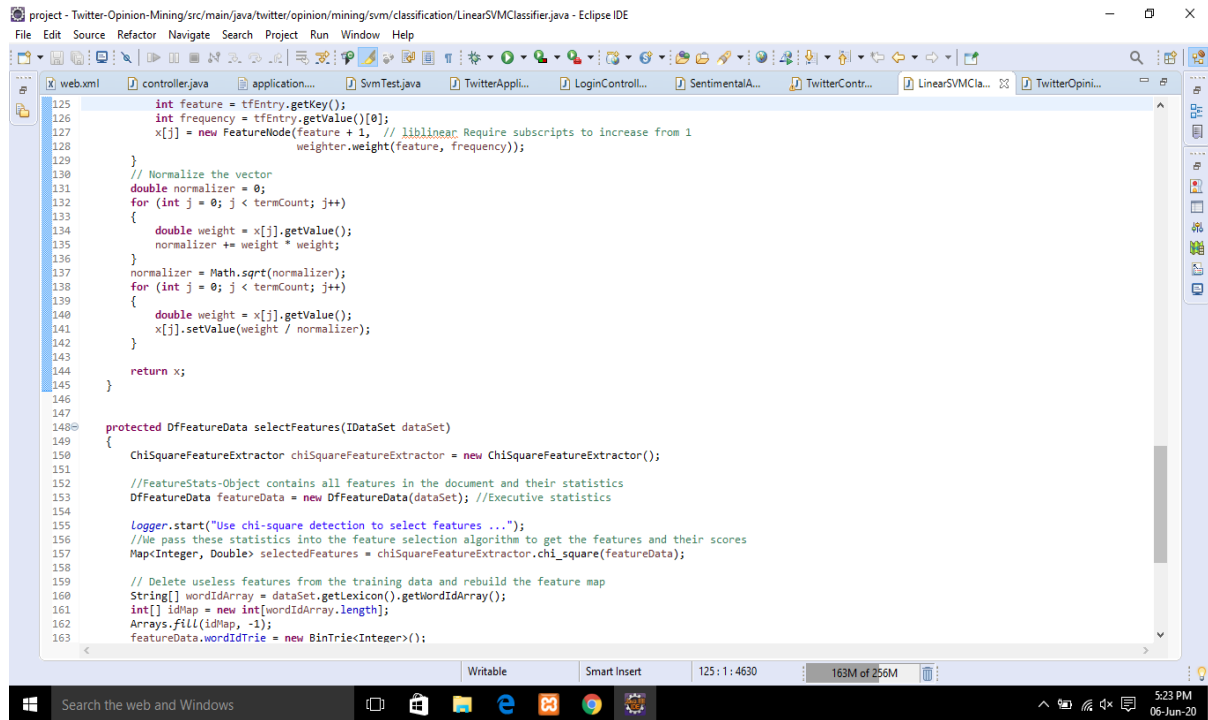
```

project - Twitter-Opinion-Mining/src/main/java/twitter/opinion/mining/svm/classification/LinearSVMClassifier.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

web.xml controller.java application... SvmTest.java TwitterAppli... LoginControll... SentimentalA... TwitterContr... LinearSVMCla... TwitterOpini...

88 private Model solveLibLinearProblem(Problem problem)
89 {
90     de.bwaldvogel.liblinear.Parameter lparam = new Parameter(SolverType.LIR_LR,
91                                                             500.,
92                                                             0.01);
93     return de.bwaldvogel.liblinear.Linear.train(problem, lparam);
94 }
95
96 private Problem createLibLinearProblem(IDataset dataSet, BaseFeatureData baseFeatureData, IFeatureWeighter weighter)
97 {
98     Problem problem = new Problem();
99     int n = dataSet.size();
100     problem.l = n;
101     problem.n = baseFeatureData.featureCategoryJointCount.length;
102     problem.x = new FeatureNode[n][];
103     problem.y = new double[n]; //The latest version of liblinear's y array is a floating point number
104     Iterator<Document> iterator = dataSet.iterator();
105     for (int i = 0; i < n; i++)
106     {
107         // Constructing document vectors
108         Document document = iterator.next();
109         problem.x[i] = buildDocumentVector(document, weighter);
110         //Set the sample's y value
111         problem.y[i] = document.category;
112     }
113     return problem;
114 }
115
116 private FeatureNode[] buildDocumentVector(Document document, IFeatureWeighter weighter)
117 {
118     int termCount = document.tfMap.size(); //Number of words
119     FeatureNode[] x = new FeatureNode[termCount];
120     Iterator<Map.Entry<Integer, int[]>> tfMapIterator = document.tfMap.entrySet().iterator();
121     for (int j = 0; j < termCount; j++)
122     {
123         Map.Entry<Integer, int[]> tfEntry = tfMapIterator.next();
124         int feature = tfEntry.getKey();
125         int frequency = tfEntry.getValue()[0];

```

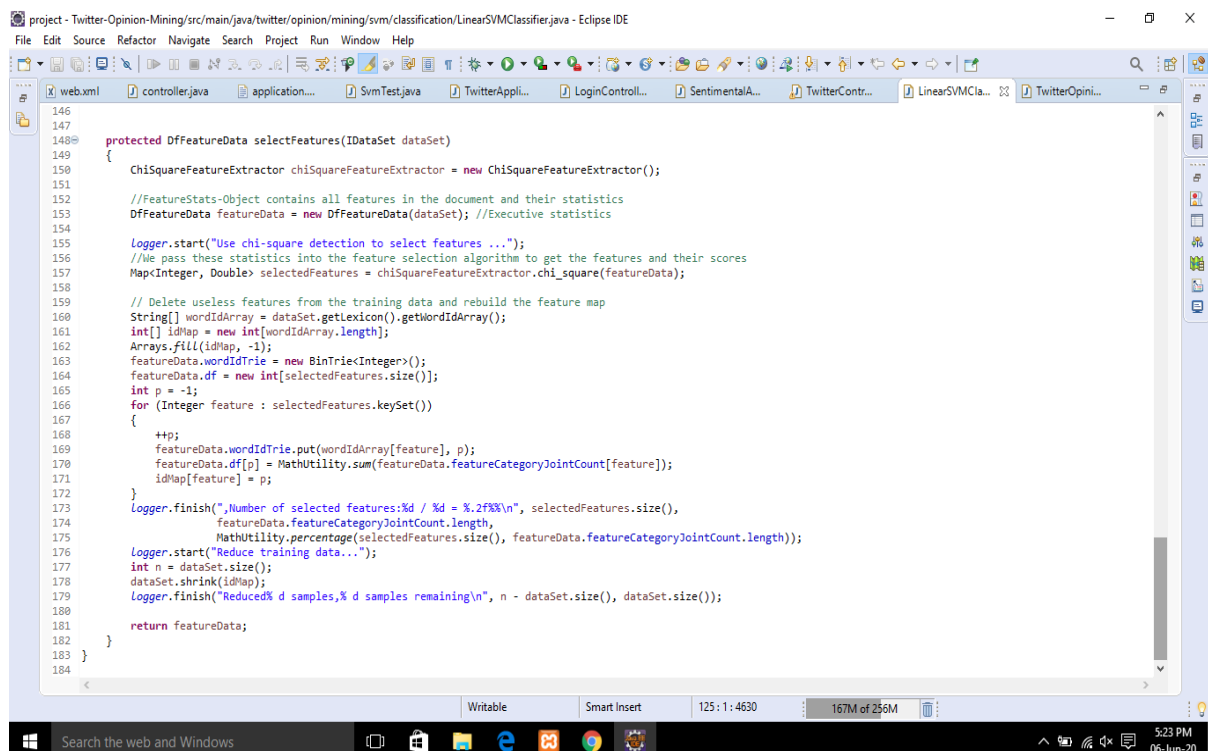



The screenshot shows the Eclipse IDE with the file `LinearSVMClassifier.java` open. The code is in the `src/main/java/twitter/opinion/mining/svm/classification` package. It implements a linear SVM classifier for sentiment analysis. The `selectFeatures` method uses a Chi-Square feature extractor to select features from a dataset. The code includes comments explaining the steps: normalizing the vector, selecting features using Chi-Square, and deleting useless features from the training data. The IDE interface shows the project structure on the left, the code editor in the center, and the Windows taskbar at the bottom.

```

125 int feature = tfEntry.getKey();
126 int frequency = tfEntry.getValue()[0];
127 x[j] = new FeatureNode(feature + 1, // liblinear Require subscripts to increase from 1
128     weighter.weight(feature, frequency));
129 }
130 // Normalize the vector
131 double normalizer = 0;
132 for (int j = 0; j < termCount; j++)
133 {
134     double weight = x[j].getValue();
135     normalizer += weight * weight;
136 }
137 normalizer = Math.sqrt(normalizer);
138 for (int j = 0; j < termCount; j++)
139 {
140     double weight = x[j].getValue();
141     x[j].setValue(weight / normalizer);
142 }
143 }
144 return x;
145 }
146
147
148 protected DfFeatureData selectFeatures(IDataset dataSet)
149 {
150     ChiSquareFeatureExtractor chiSquareFeatureExtractor = new ChiSquareFeatureExtractor();
151
152     //FeatureStats-Object contains all features in the document and their statistics
153     DfFeatureData featureData = new DfFeatureData(dataSet); //Executive statistics
154
155     logger.start("Use chi-square detection to select features ...");
156     //We pass these statistics into the feature selection algorithm to get the features and their scores
157     Map<Integer, Double> selectedFeatures = chiSquareFeatureExtractor.chi_square(featureData);
158
159     // Delete useless features from the training data and rebuild the feature map
160     String[] wordIdArray = dataSet.getLexicon().getWordIdArray();
161     int[] idMap = new int[wordIdArray.length];
162     Arrays.fill(idMap, -1);
163     featureData.wordIdTrie = new BinTrie<Integer>();
164     featureData.df = new int[selectedFeatures.size()];
165     int p = -1;
166     for (Integer feature : selectedFeatures.keySet())
167     {
168         ++p;
169         featureData.wordIdTrie.put(wordIdArray[feature], p);
170         featureData.df[p] = MathUtility.sum(featureData.featureCategoryJointCount[feature]);
171         idMap[feature] = p;
172     }
173     logger.finish("Number of selected features: %d / %d = %.2f%%\n", selectedFeatures.size(),
174         featureData.featureCategoryJointCount.length,
175         MathUtility.percentoge(selectedFeatures.size(), featureData.featureCategoryJointCount.length));
176     logger.start("Reduce training data...");
177     int n = dataSet.size();
178     dataSet.shrink(idMap);
179     logger.finish("Reduced %d samples, %d samples remaining\n", n - dataSet.size(), dataSet.size());
180
181     return featureData;
182 }
183 }
184

```



The screenshot shows the Eclipse IDE with the file `LinearSVMClassifier.java` open. The code is in the `src/main/java/twitter/opinion/mining/svm/classification` package. It implements a linear SVM classifier for sentiment analysis. The `selectFeatures` method uses a Chi-Square feature extractor to select features from a dataset. The code includes comments explaining the steps: normalizing the vector, selecting features using Chi-Square, and deleting useless features from the training data. The IDE interface shows the project structure on the left, the code editor in the center, and the Windows taskbar at the bottom.

```

146
147
148 protected DfFeatureData selectFeatures(IDataset dataSet)
149 {
150     ChiSquareFeatureExtractor chiSquareFeatureExtractor = new ChiSquareFeatureExtractor();
151
152     //FeatureStats-Object contains all features in the document and their statistics
153     DfFeatureData featureData = new DfFeatureData(dataSet); //Executive statistics
154
155     logger.start("Use chi-square detection to select features ...");
156     //We pass these statistics into the feature selection algorithm to get the features and their scores
157     Map<Integer, Double> selectedFeatures = chiSquareFeatureExtractor.chi_square(featureData);
158
159     // Delete useless features from the training data and rebuild the feature map
160     String[] wordIdArray = dataSet.getLexicon().getWordIdArray();
161     int[] idMap = new int[wordIdArray.length];
162     Arrays.fill(idMap, -1);
163     featureData.wordIdTrie = new BinTrie<Integer>();
164     featureData.df = new int[selectedFeatures.size()];
165     int p = -1;
166     for (Integer feature : selectedFeatures.keySet())
167     {
168         ++p;
169         featureData.wordIdTrie.put(wordIdArray[feature], p);
170         featureData.df[p] = MathUtility.sum(featureData.featureCategoryJointCount[feature]);
171         idMap[feature] = p;
172     }
173     logger.finish("Number of selected features: %d / %d = %.2f%%\n", selectedFeatures.size(),
174         featureData.featureCategoryJointCount.length,
175         MathUtility.percentoge(selectedFeatures.size(), featureData.featureCategoryJointCount.length));
176     logger.start("Reduce training data...");
177     int n = dataSet.size();
178     dataSet.shrink(idMap);
179     logger.finish("Reduced %d samples, %d samples remaining\n", n - dataSet.size(), dataSet.size());
180
181     return featureData;
182 }
183 }
184

```