A Preliminary Project Report

on

"SARCASM DETECTION USING TWITTER ANALYSIS"

Submitted to the

Savitribai Phule Pune University

In partial fulfilment for the award of the Degree of

Bachelor of Engineering

in

Information Technology

by

ABUZAR TAMBOLI

SAKSHI MORE

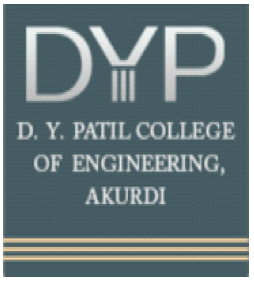
SATYAM JAIMINI

KASTURI POKHARKAR

GANESH KARLE

Under the guidance of

Mrs. AMITA JAJOO



Department of Information Technology

D. Y. Patil College of Engineering

Akurdi, Maharashtra, India

2020-2021

CERTIFICATE

This is to certify that the project based seminar report entitled ***"SARCASM DETECTION USING TWITTER ANALYSIS"*** *being submitted by* ***ABUZAR TAMBOLI, SAKSHI MORE, SATYAM JAIMINI, KASTURI POKHARKAR, GANESH KARLE*** *is a record of bonafide work carried out by him/her under the supervision and guidance of Mrs. AMITA JAJOO in partial fulfilment of the requirement for BE (Information Technology Engineering) – 2015 course of Savitribai Phule Pune University, Pune in the academic year 2020-2021.*

Date:

Place:

Mrs. Amita Jajoo
Guide

Dr. Preeti Patil
Head of the Department

This project based seminar report has been examined by us as per the Savitribai Phule Pune University, Pune, requirements at D. Y. PATIL College of Engineering on . . . . . .

(Name & Signature)
Internal Examiner

(Name & Signature)
External Examiner

Acknowledgement

We take this opportunity to thank our project guide Mrs. Amita Jajoo for their valuable guidance and constant supervision as well as for providing all the necessary facilities, which were indispensable in the completion of this project report.

We would like to express our gratitude towards the Head of I.T. Department Mrs. Preeti Patil for her kind cooperation and encouragement which helped us in completion of this Project and for always being ready to help with the most diverse problems that we have encountered along the way.

We are also thankful to all the staff members of the Department of Information Technology of D.Y. Patil College of Engineering, Akurdi, Pune for their valuable time, support, comments, suggestions and persuasion. We would also like to thank the institute for providing the required facilities, Internet access and important books.

Abuzar Tamboli
Sakshi More
Satyam Jaimini
Kasturi Pokharkar
Ganesh Karle

Abstract

Sarcasm is a sophisticated form of irony widely used in social networks and micro blogging websites. It is usually used to convey implicit information within the message a person transmits. Sarcasm might be used for different purposes, such as criticism or mockery. Twitter became one of the biggest web destinations for people to express their opinions, share their thoughts and report real-time events.

Keywords: Twitter, sentiment analysis, sarcasm detection, machine learning.

INDEX

CHAPTER NO.	TITLE	PAGE NO.
1.	Introduction	
	1.1 Overview	
	1.1.1 Motivation	
	1.1.2 Objective	
2.	Literature Survey	
	2.1 Study of Research Paper	
3.	Problem Statement	
4.	Project Requirement	
	4.1 External Interface Requirement	
	4.1.1 User Interface	
	4.1.2 Hardware Interface	
	4.1.3 Software Interface	
	4.2 Non Functional Requirement	
	4.2.1 Performance Requirements	
	4.2.2 Safety Requirement	
5.	System Analysis	
	5.1 System Architecture	
	5.1.1 Module	
	5.1.2 Data Flow Diagram	
	5.2 UML DIAGRAMS	

6.	Mathematical Model	
7.	Software Information	
8.	Project Plan	
	8.1 System Implementation Plan	
9.	Modules Description	
10.	Methodology and Algorithms	
	10.1 Methodologies	
	10.1.1 Support Vector Machine	
	10.1.2 Neural Networks	
11.	Conclusion	
	11.1 Conclusion	
12.	References	

List of Figures

Figue no.	Figures	Page no.
5.1	System Architecture	
5.2	Data Flow diagram	
5.3	Data Flow diagram	
5.4	Data Flow diagram	
5.5	Class Diagram Diagram	
5.6	Object Diagram	

5.7	Package Diagram	
5.8	State Diagram	
5.9	Use Case Diagram	
5.10	Activity Diagram	
5.11	Sequence Diagram	
5.12	Component Diagram	
5.13	Deployment Diagram	
10.1		
10.2		
10.3		

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

There are different trends opening in the era of sentiment analysis, which analyze the attitude and opinion of people in social media, which includes social sites like Facebook,Twitter, blogs, etc. The main aim of sentiment analysis is to identify the polarity (positive, negative or neutral) in a given text.

Sarcasm is a special type of sentiment which has the ability to flip the polarity of the given text. Sarcasm is defined as 'the use of irony to mock or convey contempt'. Sarcasm is a sophisticated form of sentiment expression where speakers express their opinions opposite of what they mean. Sarcasm is a contrast between positive sentiment words and negative situation.To detect sarcastic tweets. Although it does not need an already-built user knowledge base as in the work of Raj a desingu et al.

Our approach considers the different types of sarcasm and detects the sarcastic tweets regardless of their owners or their temporal context, with a precision that reaches 91.1%. sarcasm as a form of irony that is intended to express contempt. Since most of the focus on sarcasm is to enhance and the existing automatic sentiment analysis systems, we also use the two terms synonymously.

1.1.1 Motivation

Although some users indicate they are being sarcastic, most of them do not.Therefore, it might be indispensable to and a way to automatically detect any sarcastic messages.

1.1.2 Objective

- Primary objective is to analyze whether a tweet is sarcastic or not.
- To identify emotional patterns.
- To identify people's intentions.

CHAPTER 2

LITERATURE SURVEY

2.1 STUDY OF RESEARCH PAPER

1.Paper Name: Review of Automatic Sarcasm Detection

Author: Shalini Raghav , Ela Kumar, 2017

Abstract : Sarcasm is an alternate sort of assessment that comprises words which by and large methods something contrary to what you truly need to state and can be utilized in various circumstances, to affront, to show disturbance, or be clever. Here and there to pass on understood data, sarcasm is utilized inside the message as an individual conveys. Sarcasm can be used for different capacities, similar to analysis or joke. In any case, sarcasm is extremely debilitating in any event, for people to recognize. Therefore, the

detection of sarcasm leads to a better understanding of sentiment analysis of the user, from the data collected from websites such as Twitter and Facebook. The advantages of identifying sarcasm for opinion mining has prompted enthusiasm for automated sarcasm recognition as an exploration issue. In Automated sarcasm identification, the program tries to identify whether a text is sarcastic or neutral. The research paper is divided into two phases. In the first phase, it extracts features related to sentiments and punctuation, and then the chi-square test is used to shortlist the most useful features. In the second phase, 200 top tf-idf features are extracted and combined with sentiment related and punctuation related features to find sarcastic content within the tweet. In the first approach the best accuracy is achieved by using the Support Vector Machine algorithm which is equal to 74.59% and in the second approach the maximum accuracy is achieved by the voting classifier and it increases the accuracy to 83.53% .

2.Paper Name: Arabic Sarcasm Detection in Twitter

Author: Dana Al-Ghadhban,Eman Alnkhilan

Abstract: Sarcasm is a special form of irony or satirical wit in which people convey the opposite of what they mean. Sarcasm largely increases in social networks, especially on Twitter. Detecting sarcasm in tweets improves the automatic analysis tools that analyze the data to provide or enhance customer service and fabricate or enhance a product. Also, there are few studies that focus on detecting Arabic sarcasm in tweets. Consequently, we propose a classifier model that detects Arabic-sarcasm tweets by classifying them as sarcastic by setting some features that may declare a tweet as sarcastic usingWeka. We evaluated our model through recall, precision, and

f-score measurements that gave 0.659, 0.710, and 0.676 values, respectively, which these results are high especially when it comes to Arabic.

3.**Paper Name:** Automatic Sarcasm Detection: A Survey

Author: ADITYA JOSHI, PUSHPAK BHATTACHARYYA, 2018

Description : —In recent times, sarcasm analysis has been one of the toughest challenges in Natural Language Processing (NLP). The property of sarcasm that makes it difficult to analyze and detect is the gap between its literal and intended meaning.Detecting sarcastic sentiment in the domain of social media such as Facebook, Twitter, online blogs, reviews, etc. has become an essential task as they influence every business organization. In this article, a hyperbolic feature-based sarcasm detector for Twitter data is proposed. The

hyperbolic features consist of intensifiers and interjections of the text. The performance of the proposed system is analyzed using several standard machine learning approaches namely, Naive Bayes (NB), Decision Tree (DT), Support Vector Machine (SVM), Random Forest (RF), and AdaBoost. The system attains an accuracy (%) of 75.12, 80.27, 80.67, 80.79, and 80.07 using NB, DT, SVM, RF, and AdaBoost respectively.

4.Paper Name: Sarcasm Detection in Twitter

Author: Mondher Bouazizi,Tomoaki Ohtsuki

Description : Sarcasm is a special form of irony by which the person conveys implicit information, usually the opposite of what is said, within the message he transmits. Sarcasm is largely used in social networks and microblogging websites, where people mock or criticize in a way that makes it difficult even for humans to tell if what is said is what is meant. Recognizing sarcastic statements can be very useful when it comes to improving automatic sentiment analysis of data collected from social networks. It helps also enhance the efficiency of after-sales services or consumer assistance through understanding

the intentions and real opinions of consumers when browsing their feedback or complaints. In this paper we propose a method to detect sarcasm in Twitter that makes use of the different components of the tweet. We propose four sets of features that cover different types of sarcasm we defined, and that will be used to classify tweets into sarcastic and non-sarcastic. We evaluate the performances of our approach. We study the importance of each of the proposed sets of features and evaluate its added value to the classification.

5.Paper Name: A Pattern-Based Approach for Sarcasm Detection on Twitter

Author:MONDHER BOUAZIZI AND TOMOAKI OTSUKI

Abstract: Sarcasm is a sophisticated form of irony widely used in social networks and microblogging websites. It is usually used to convey implicit information within the message a person transmits. Sarcasm might be used for different purposes, such as criticism or mockery. However, it is hard even for humans to recognize. Therefore, recognizing sarcastic statements can be very useful to improve automatic sentiment analysis of data collected from

microblogging websites or social networks. Sentiment Analysis refers to the identification and aggregation of attitudes and opinions expressed by Internet users toward a specific topic. In this paper, we propose a pattern-based approach to detect sarcasm on Twitter. We propose four sets of features that cover the different types of sarcasm we defined. We use those to classify tweets as sarcastic and non-sarcastic. Our proposed approach reaches an accuracy of 83.1% with a precision equal to 91.1%. We also study the importance of each of the proposed sets of features and evaluate its added value to the classification. In particular, we emphasize the importance of pattern-based features for the detection of sarcastic statements.

CHAPTER 3

PROBLEM STATEMENT

- How to use the output of the current one to enhance the performances of sentiment analysis and opinion mining.

- The purpose is to find an efficient way to detect sarcastic tweets, and study how to use this information (i.e., whether the tweet is sarcastic or not) to enhance the accuracy of sentiment analysis.

- To detect sarcastic comments/statements from large text data.

- Sarcasm turns into social violence on social media platforms.

CHAPTER 4

PROJECT REQUIREMENT

4.1 EXTERNAL INTERFACE REQUIREMENT

4.1.1 User Interface

Application Based On sarcasm Detection.

4.1.2 Hardware Interfaces:

RAM : 8 GB

As we are using Machine Learning Algorithm and Various High Level Libraries

Laptop

RAM minimum required is 8 GB.

Hard Disk : 40 GB

Data Set of CT Scan images is to be used hence minimum 40 GB Hard Disk memory is required.

Processor : Intel i5 Processor

Pycharm IDE that Integrated Development Environment is to be used and data loading should be fast hence Fast Processor is required.

IDE : Pycharm

Best Integrated Development Environment as it gives possible suggestions at the time of typing code snippets that makes typing feasible and fast.

Coding Language : Python Version 3.5

Highly specified Programming Language for Machine Learning because of availability of High Performance Libraries.

Operating System : Windows 10

Latest Operating System that supports all type of installation and development Environment.

4.1.3 Software Interfaces

Operating System: Windows 10

IDE: Pycharm ,Spyder

Programming Language : Python

4.2 NON FUNCTIONAL REQUIREMENT

4.2.1 PerformanceRequirements

The performance of the functions and every module must be well. The overall performance of the software will enable the users to work efficiently.

Performance of encryption of data should be fast. Performance of the providing virtual environment should be fast.

4.2.2 Safety Requirement

The application is designed in modules where errors can be detected and fixed easily.

This makes it easier to install and update new functionality if required.

4.2.3 Software Quality Attributes

Our software has many quality attribute that are given below:-

Adaptability: This software is adaptable by all users.

Availability: This software is freely available to all users. The availability of the software is easy for everyone.

Maintainability: After the deployment of the project if any error occurs then it can be easily maintained by the software developer.

Reliability: The performance of the software is better which will increase the reliability of the Software.

User Friendliness: Since, the software is a GUI application; the output generated is much user friendly in its behavior.

Integrity: Integrity refers to the extent to which access to software or data by unauthorized persons can be controlled.

Security: Users are authenticated using many security phases so reliable security is provided.

Testability: The software will be tested considering all the aspects.

CHAPTER 5

SYSTEM ANALYSIS

5.1 SYSTEM ARCHITECTURE

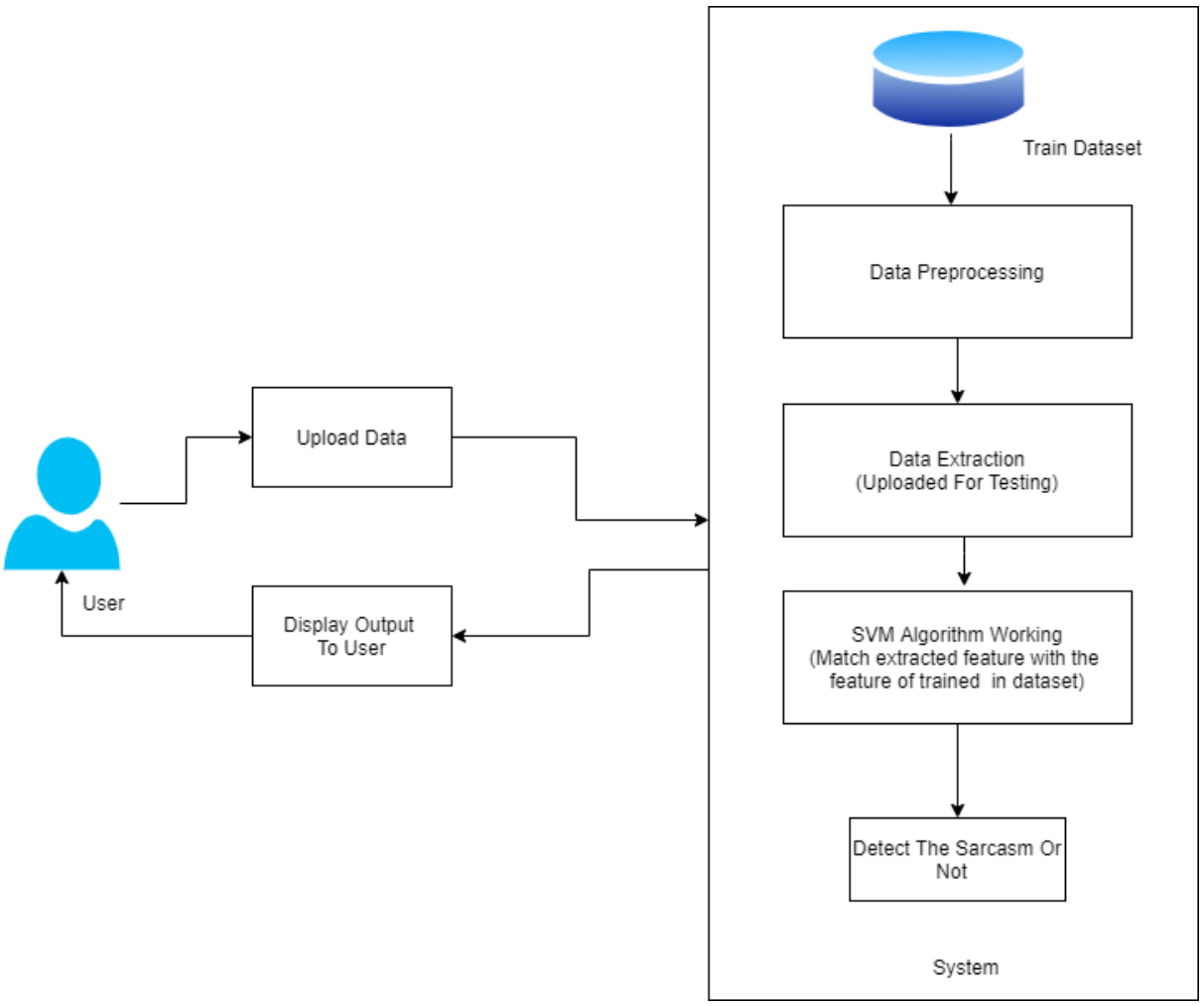


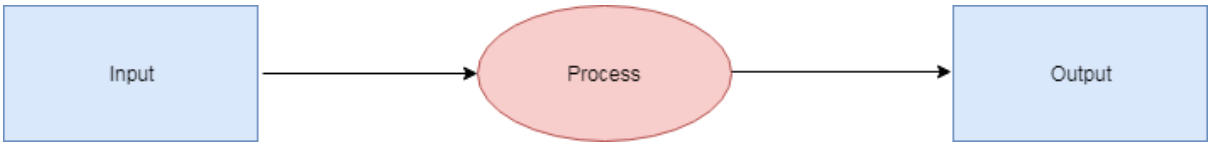
Figure 5.1: System Architecture

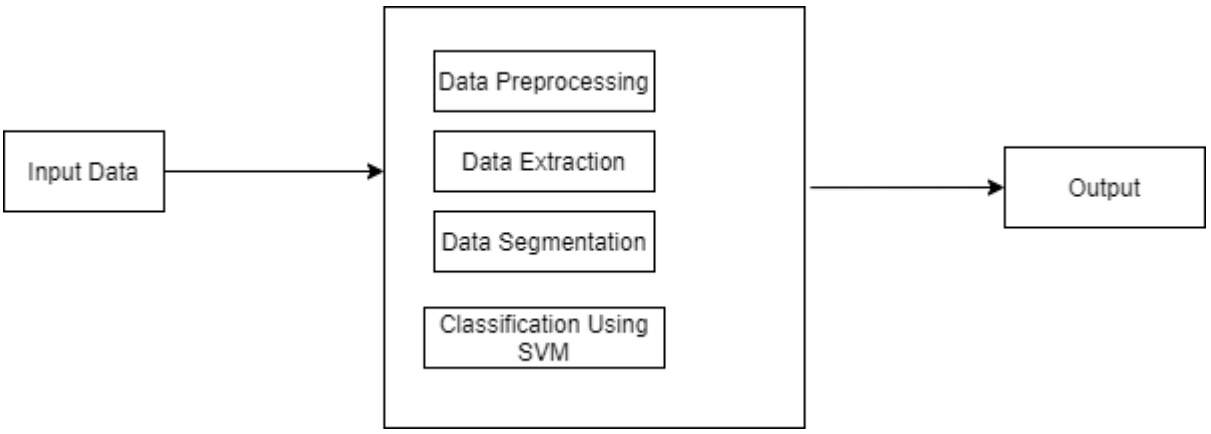
5.1.1 Module

• Admin

5.1.2 Data Flow Diagram

In Data Flow Diagram,we Show that flow of data in our system in DFD0 we show that base DFD in which rectangle present input as well as output and circle show our system,In DFD1 we show actual input and actual output of system input of our system is text or image and output is rumor detected likewise in DFD 2 we present operation of user as well as admin.





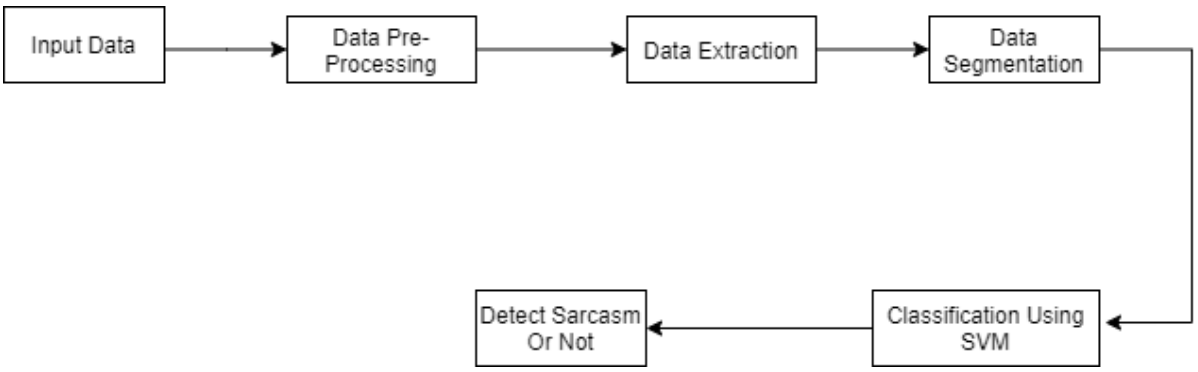


Figure 5.4: Data Flow diagram

5.2 UML DIAGRAMS

Unified Modeling Language is a standard language for writing software blueprints.The UML may be used to visualize, specify , construct and document the artifacts of a software intensive system.UML is process independent, although optimally it should be used in process that is use case driven, architecture-centric,iterative,and incremental. The Number of UML Diagrams is available.

Use case Diagram.

Class Diagram.

Object Diagram.

Package Diagram.

State Diagram.

Activity Diagram.

Sequence Diagram.

Component Diagram.

Deployment Diagram

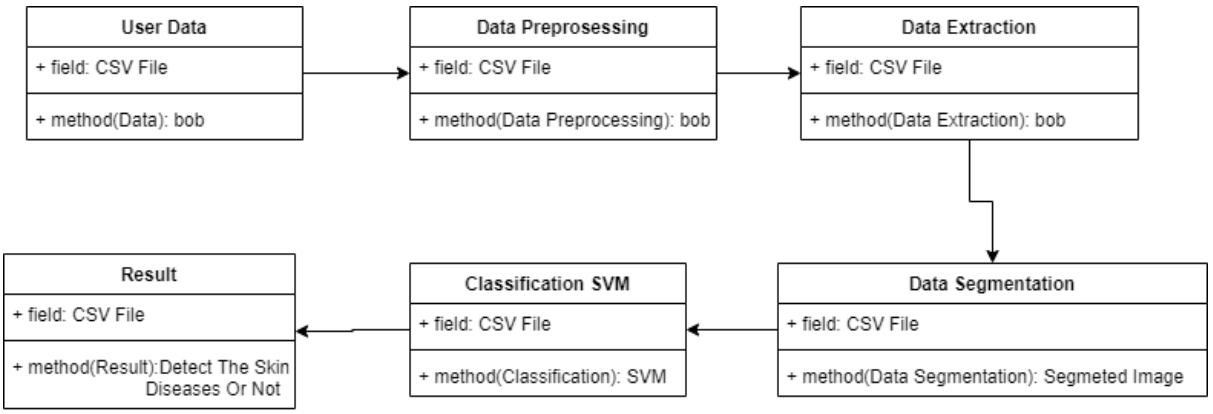


Figure 5.5: Class Diagram Diagram

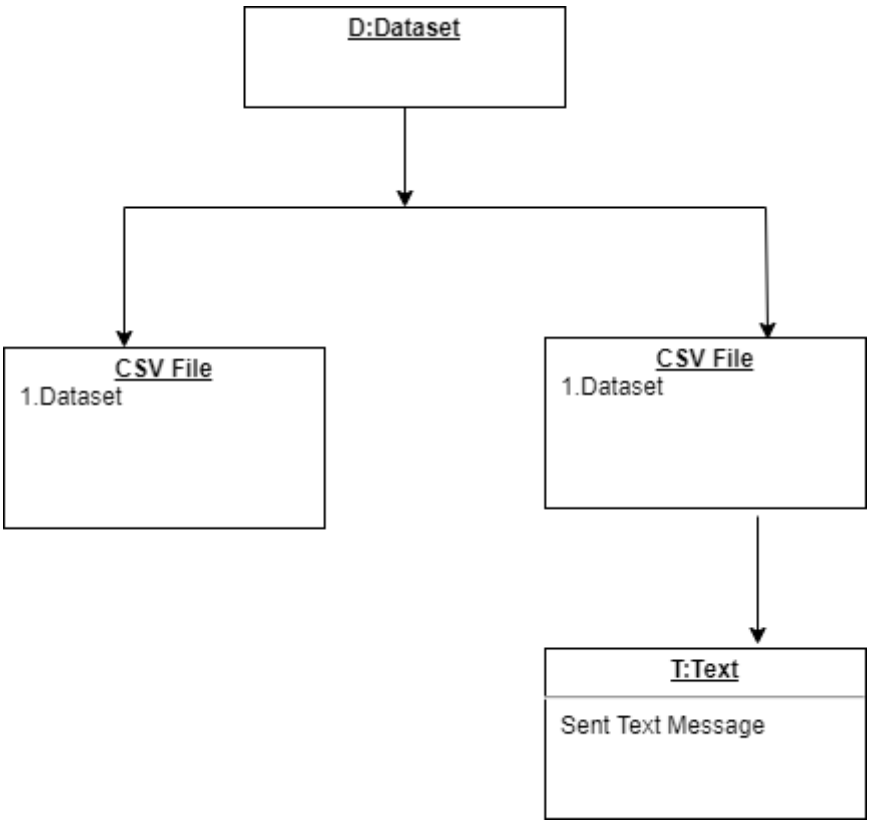


Figure 5.6: Object Diagram

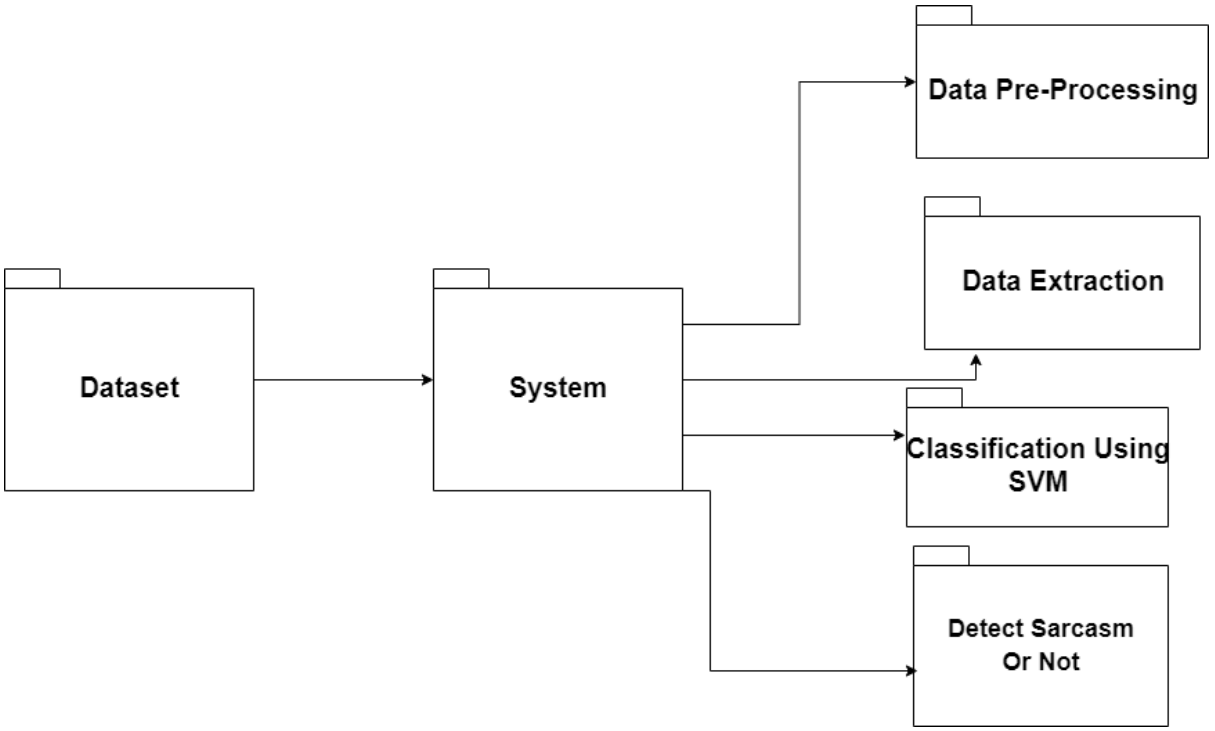


Figure 5.7: Package Diagram

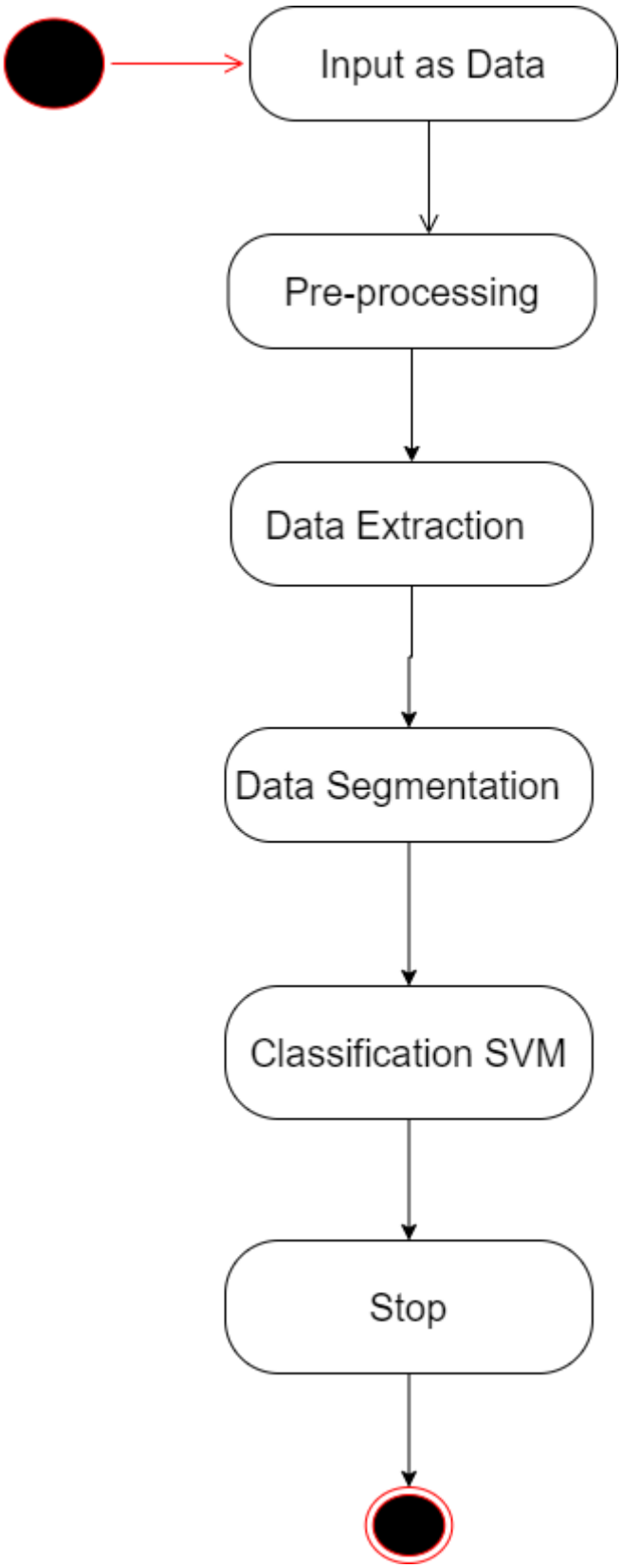


Figure 5.8: State Diagram

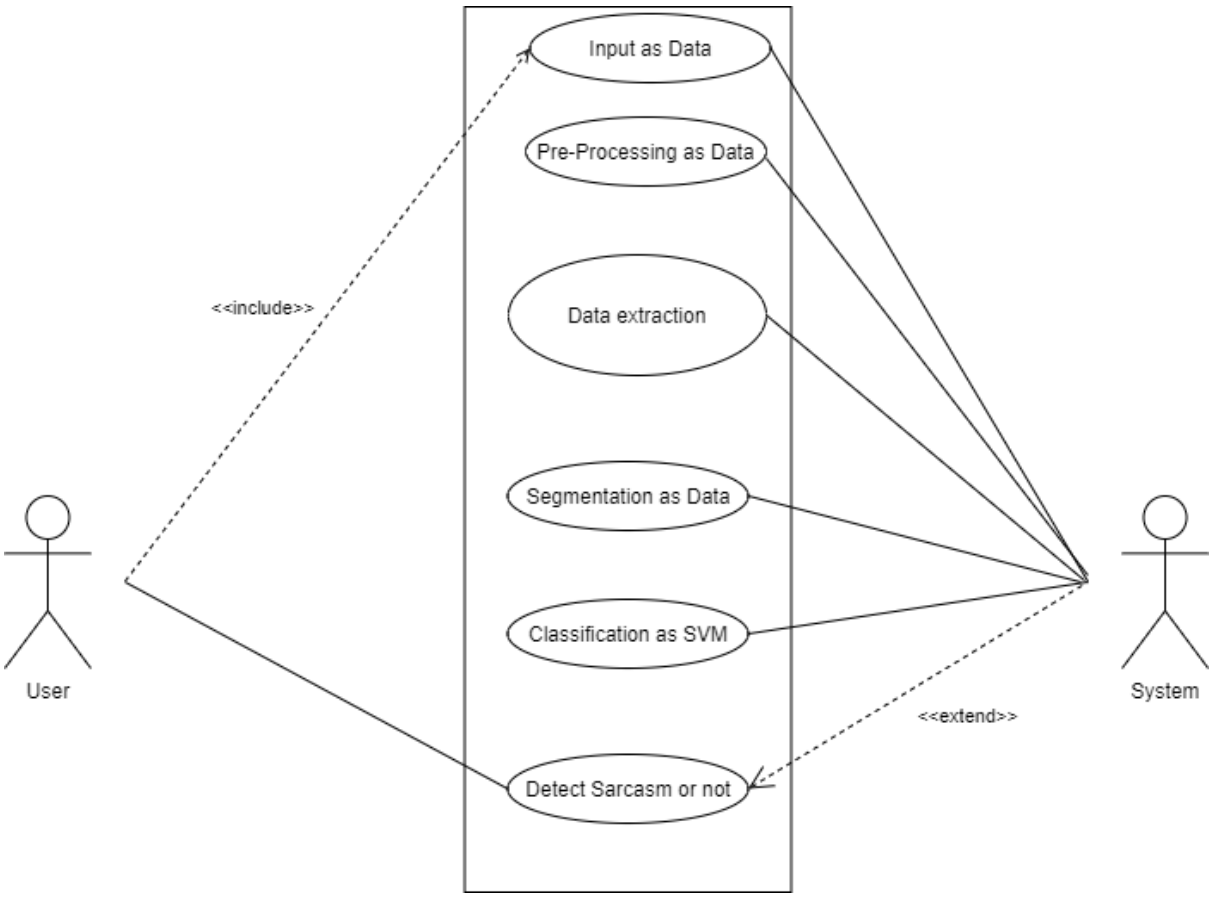


Figure 5.9: Use Case Diagram

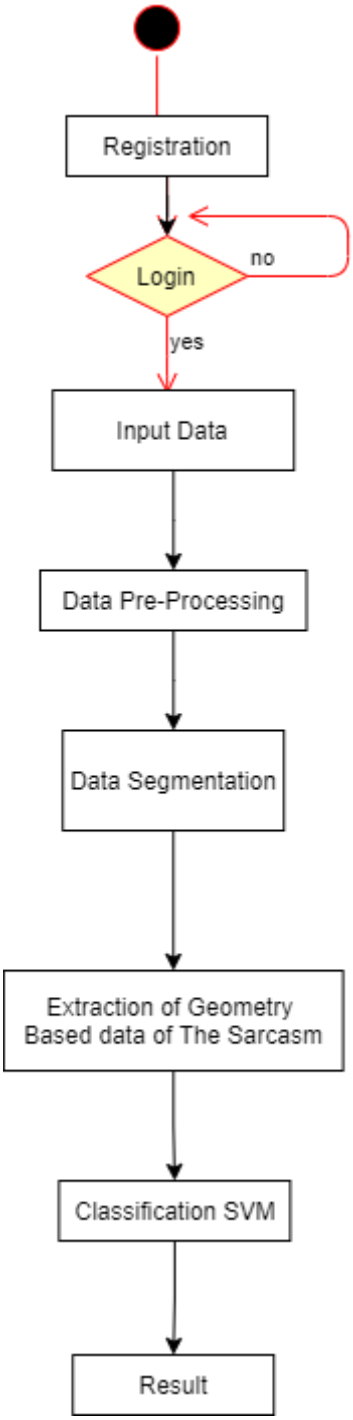


Figure 5.10: Activity Diagram

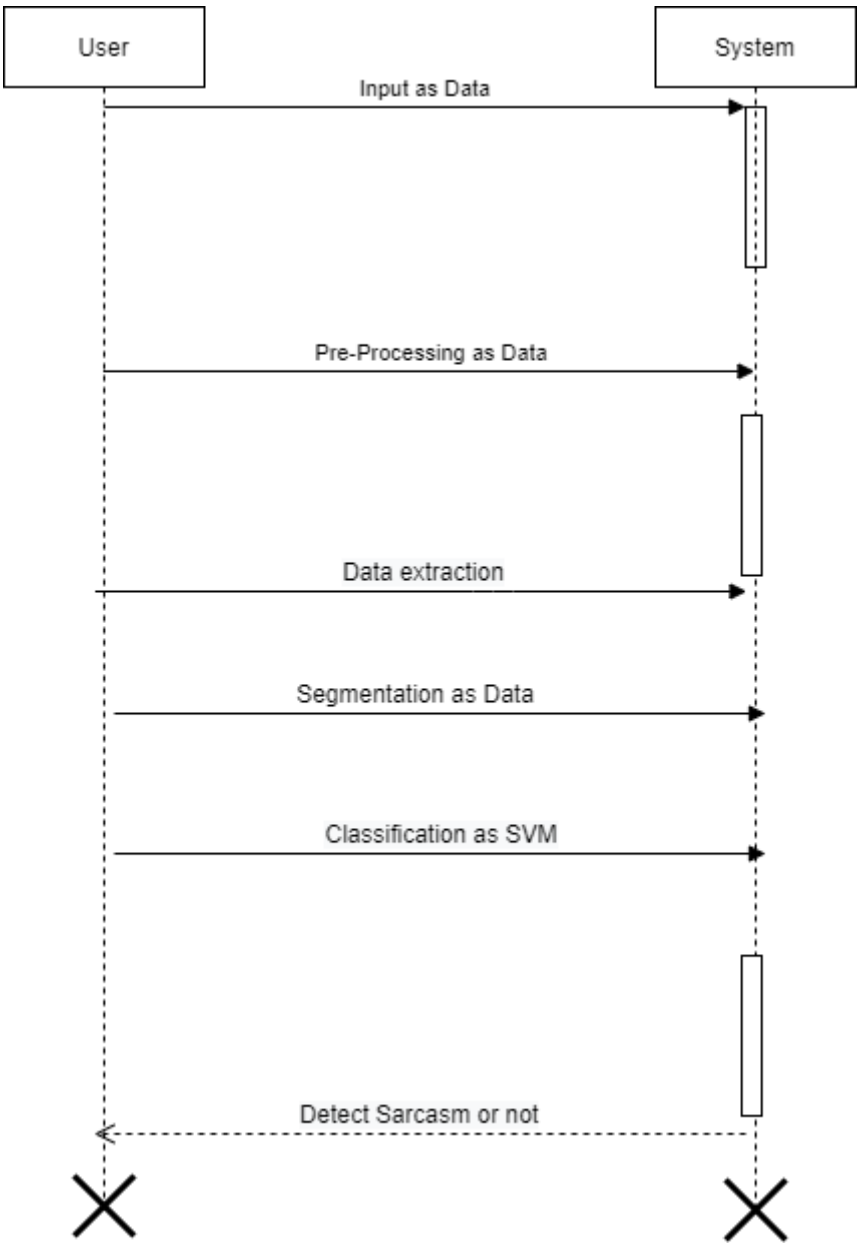


Figure 5.11: Sequence Diagram

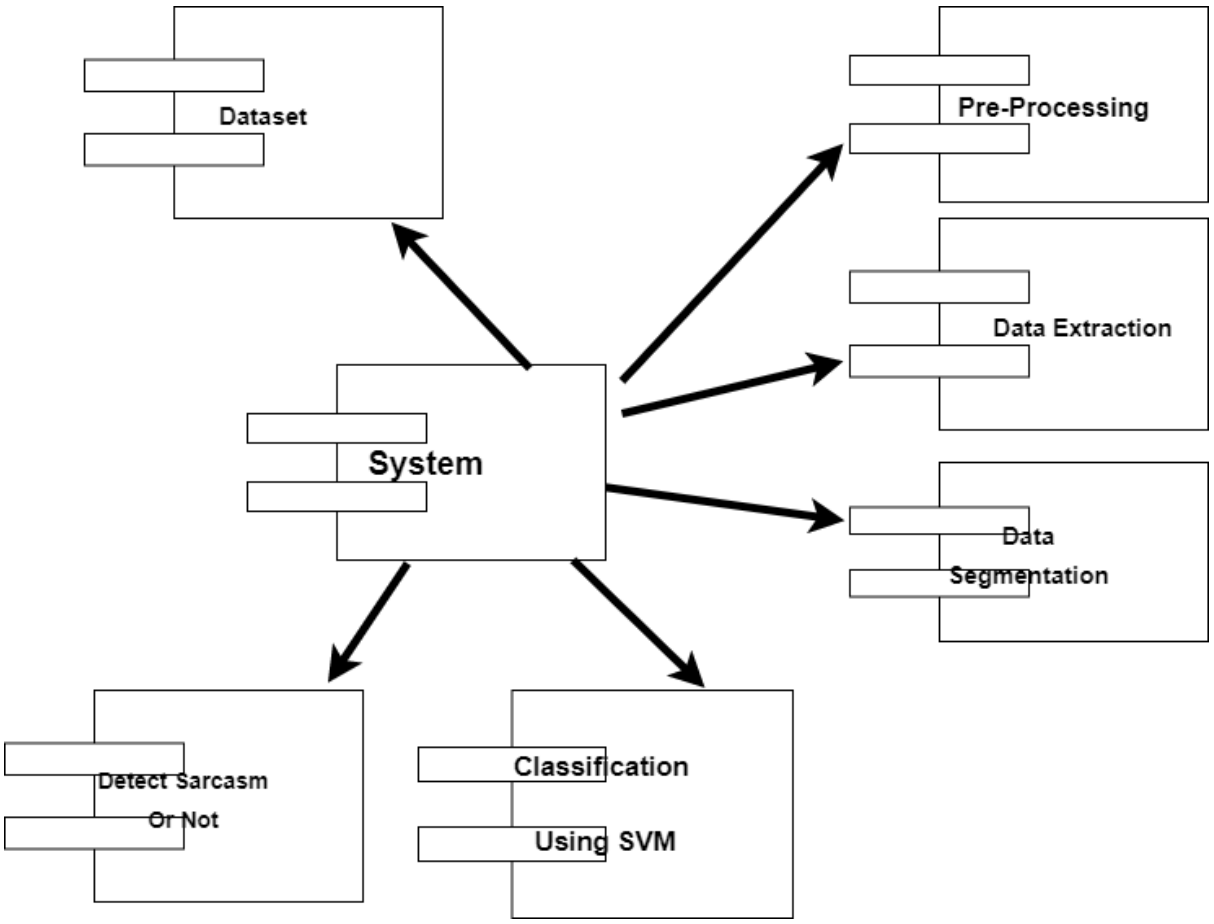


Figure 5.12: Component Diagram

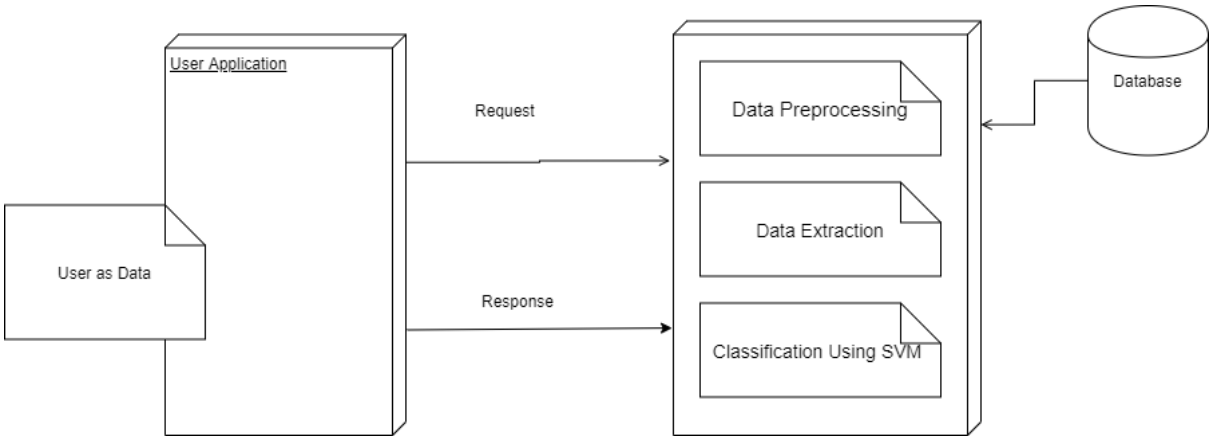


Figure 5.13: Deployment Diagram

CHAPTER 6

MATHEMATICAL MODEL

Let S be the Whole system which consists:

S= Twitter Dataset. Where, Twitter Dataset is the input of the system.

Pro is the procedure applied to the system to classification of the given input.

OP is the output of the project to detect the people to express their opinions

A. Input:

IP = I. Where, I is a set of Twitter Dataset , provided as an input.

B. Procedure:

Step1: Twitter Dataset .

Step2: Verify the information into the database.

Step3: use of Part-of-Speech tags to extract patterns characterizing the level of sarcasm of tweets.

Step4: Show result.

C. Output: to detect the people to express their opinions.

CHAPTER 7

SOFTWARE INFORMATION

Python:

Python is an interpreted, high-level and general-purpose programming language. Created by Guido van Rossum and first released in 1991, Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects. Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly, procedural), object-oriented, and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard library. Python was created in the late 1980s as a successor to the ABC language. Python 2.0, released in 2000, introduced features like list comprehensions and a garbage collection system with reference counting.

Python 3.0, released in 2008, was a major revision of the language that is not completely backward-compatible, and much Python 2 code does not run unmodified on Python 3.

The Python 2 language was officially discontinued in 2020 (first planned for (2015), and "Python 2.7.18 is the last Python 2.7 release and therefore the last Python 2 release." No more security patches or other improvements will be released for it. With Python 2's end-of-life, only Python 3.6.x and later are supported.

Python interpreters are available for many operating systems. A global community of programmers develops and maintains CPython, a free and open-source reference implementation. A non-profit organization, the Python Software Foundation, manages and directs resources for Python and CPython development.

Python was conceived in the late 1980s by Guido van Rossum at Centrum Wiskunde Informatica (CWI) in the Netherlands as a

successor to the ABC language (itself inspired by SETL), capable of exception handling and interfacing with the Amoeba operating system. Its implementation began in December 1989. Van Rossum shouldered sole responsibility for the project, as the lead developer, until 12 July 2018, when he announced his ”permanent vacation” from his responsibilities as Python’s Benevolent Dictator For Life, a title the Python community bestowed upon him to reflect his long-term commitment as the project’s chief decision-maker. He now shares his leadership as a member of a five-person steering council. In January 2019, active Python core developers elected Brett Cannon, Nick Coghlan, Barry Warsaw, CarolWilling and Van Rossum to a five-member ”Steering Council” to lead the project.

Anaconda:

Anaconda is a free and open-source distribution of the Python and R programming languages for scientific computing (data science, machine learning applications, large-scale data processing, predictive analytics, etc., that aims to simplify package management and deployment. The distribution includes data-science packages suitable for Windows, Linux, and macOS. It is developed and maintained by Anaconda, Inc., which was founded by Peter Wang and Travis Oliphant in 2012.

As an Anaconda, Inc. product, it is also known as Anaconda Distribution or Anaconda Individual Edition, while other products from the company are Anaconda Team Edition and Anaconda Enterprise Edition, both of which are not free. Package versions in Anaconda are managed by the package management system conda. This package manager was spun out as a separate open-source package as it ended up being useful on its own and for other things than Python. There is also a small, bootstrap version of Anaconda called Miniconda, which includes only conda, Python, the packages they depend on, and a small number of other packages.

Anaconda distribution comes with over 250 packages automatically installed, and over 7,500 additional open-source packages can be installed from PyPI as well as the conda package and virtual environment manager. It also includes a GUI, Anaconda Navigator, as a graphical alternative to the command line interface (CLI). The big difference between conda and the pip package manager is in how package dependencies are managed, which is a significant challenge for Python data science and the reason conda exists. When pip installs a package, it automatically installs any dependency Python packages without checking if these conflict with previously installed packages. It will install a package and any of its dependencies regardless of the state of the existing installation[citation needed]. Because of this, a user with a working installation of, for example, Google Tensorflow, can find that it stops working having used pip to install a different package that requires a different version of the dependent numpy library than the one used by Tensorflow. In some cases, the package may appear to work but produce different results in detail.

In contrast, conda analyses the current environment including everything currently installed, and, together with any version limitations specified (e.g. the user may wish to have Tensorflow version 2,0 or higher), works out how to install a compatible set of dependencies, and shows a warning if this cannot be done. Open source packages can be individually installed from the Anaconda repository,

Anaconda Cloud (anaconda.org), or the user's own private repository or mirror, using the conda install command. Anaconda, Inc. compiles and builds the packages available in the Anaconda repository itself, and provides binaries forWindows 32/64 bit, Linux 64 bit and MacOS 64-bit. Anything available on PyPI may be installed into a conda environment using pip, and conda will keep track of what it has installed itself and what pip has installed.

Custom packages can be made using the conda build command, and can be shared with others by uploading them to Anaconda Cloud, PyPI or other repositories. The default installation of Anaconda2 includes Python 2.7 and Anaconda3 includes Python 3.7. However, it is possible to create new environments that include any version of the Python package with conda.

PyCharm:

PyCharm is an integrated development environment (IDE) used in computer programming, specifically for the Python language. It is developed by the Czech company JetBrains. It provides code analysis, a graphical debugger, an integrated unit tester, integration with version control systems (VCSes), and supports web development with Django as well as Data Science with Anaconda. PyCharm is cross-platform, with Windows, macOS and Linux versions. The Community Edition is released under the Apache License, and there is also Professional Edition with extra features –

Features

• Coding assistance and analysis, with code completion, syntax and error highlighting, linter integration, and quick fixes

• Project and code navigation: specialized project views, file structure views and quick jumping between files, classes, methods and usages

• Python refactoring: includes rename, extract method, introduce variable, introduce constant, pull up, push down and others

• Support for web frameworks: Django, web2py and Flask [professional edition only]

• Integrated Python debugger

• Integrated unit testing, with line-by-line code coverage Google App Engine Python development [professional edition only]

• Version control integration: unified user interface for Mercurial, Git, Subversion, Perforce and CVS with change lists and merge

• Support for scientific tools like matplotlib, numpy and scipy [professional edition only]

CHAPTER 8

PROJECT PLAN

In this chapter we are going to have an overview about how much time does it took to complete each task like- Preliminary Survey Introduction and Problem Statement, Literature Survey, Project Statement, Software Requirement and Specification, System Design, Partial Report Submission, Architecture Design, Implementation, Deployment, Testing, Paper Publish, Report Submission and etcetera. This chapter also focuses on a stakeholder list which gives information about the project type, user and project member who developed the system.

8.1 SYSTEM IMPLEMENTATION PLAN

The System Implementation plan table:-

Sr No.	Name/Title	Start Date	End Date
1	Preliminary Survey	27/07/20	04/08/20
2	Introduction and Problem Statement	05/08/20	13/08/20
3	Literature Survey	14/08/20	15/08/20
4	Project Statement	16/08/20	17/08/20
5	Software Requirement And Specification	06/09/20	10/09/20
6	System Design	15/11/20	26/11/20
7	Partial Report Submission	29/12/20	29/12/20
8	Architecture Design	27/07/20	27/07/20
9	Implementation	03/01/21	05/01/21
10	Deployment	11/01/21	15/01/21
11	Testing	04/03/21	05/04/21
12	Paper Publish	09/04/21	09/04/21
13	Report Submission	29/04/21	29/04/21

CHAPTER 9

MODULES DESCRIPTION

Prepossessing

As a Machine Learning Engineer, data pre-processing or data cleansing is a crucial step and most of the ML engineers spend a good amount of time in data preprocessing before building the model. Some examples for data pre-processing include outlier detection, missing value treatments and removing the unwanted or noisy data.

Similarly, Image pre-processing is the term for operations on images at the lowest level of abstraction. These operations do not increase image information content but they decrease it if entropy is an information measure. The aim of pre-processing is an improvement of the image data that suppresses undesired distortions or enhances some image features relevant for further processing and analysis tasks.

Feature Extraction

Feature extraction is a part of the dimensionality reduction process, in which an initial set of the raw data is divided and reduced to more manageable groups. So when you want to process it will be easier. The most important characteristic of these large data sets is that they have a large number of variables. These variables require a lot of computing resources to process them. So Feature extraction helps to get the best feature from those big data sets by selecting and combining variables into features, thus, effectively reducing the amount of data. These features are easy to process, but still able to describe the actual data set with the accuracy and originality.

Segmentation

In digital image processing and computer vision, image segmentation is the process of partitioning a digital image into multiple segments (sets of pixels, also known as image objects). The goal of segmentation is to simplify and/or change the representation of an

image into something that is more meaningful and easier to analyze. Image segmentation is typically used to locate objects and boundaries (lines, curves, etc.) in images. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain characteristics.

Classification

This step categorizes detected objects into predefined classes by using a suitable classification technique that compares the image patterns with the target patterns.

Convolutional Neural Network (CNN, or ConvNet) are a special kind of multi-layer neural networks, designed to recognize visual patterns directly from pixel images with minimal pre-processing. It is a special architecture of artificial neural networks. Convolutional neural networks use some of its features of visual cortex and have therefore achieved state of the art results in computer vision tasks. Convolutional neural networks are composed of two very simple elements, namely convolutional layers and pooling layers. Although simple, there are near-infinite ways to arrange these layers for a given computer vision problem. The elements of a convolutional neural network, such as convolutional and pooling layers, are relatively straightforward to understand. The challenging part of using convolutional neural networks in practice is how to design model architectures that best use these simple elements. The reason why convolutional neural networks are hugely popular is because of their architecture, the best thing is there is no need of feature extraction.

The system learns to do feature extraction and the core concept is, it uses convolution of image and filters to generate invariant features which are passed onto the next layer. The features in the next layer are convoluted with different filters to generate more invariant and abstract features and the process continues till it gets final feature/

output which is invariant to occlusions. The most commonly used architectures of convolutional neural networks are LeNet, AlexNet, ZFNet, GoogLeNet, VGGNet, and ResNet.

CHAPTER 10

METHODOLOGY AND ALGORITHMS

10.1 Methodology

In testing tweets we adopted an algorithmic process as, when we are testing tweets in each classifier if any one of the four classifiers declares it as sarcastic tweet then that tweet is further considered as sarcastic without considering other classifiers result. With this process the accuracy level was increased.

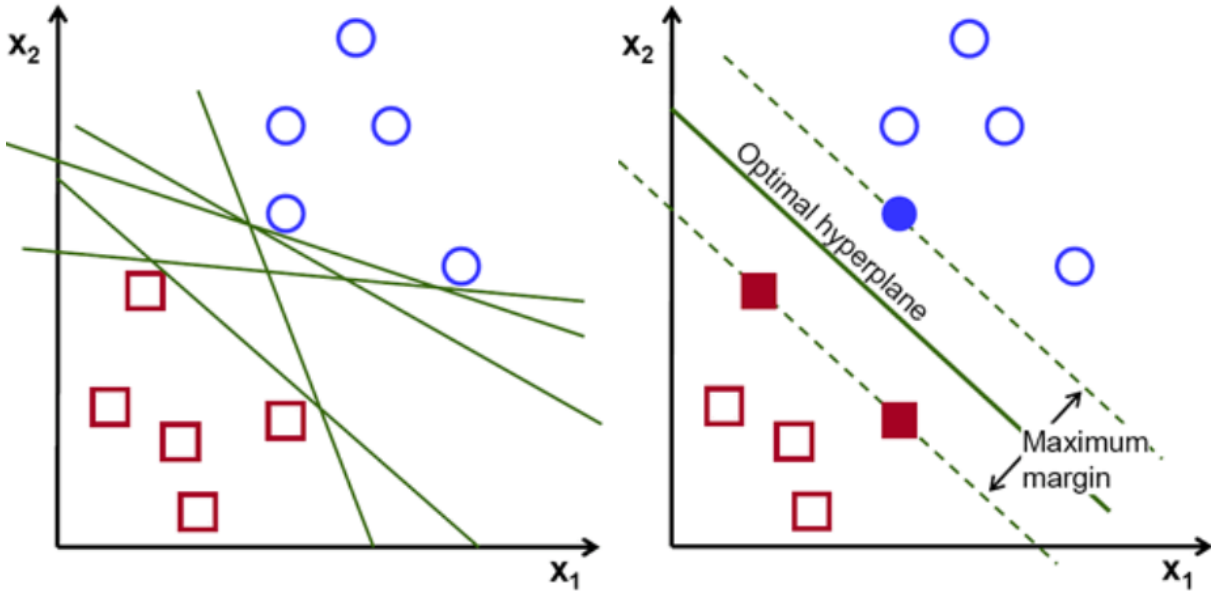
MODELS:

1. Support Vector Machine.
2. Neural Networks.

10.1.1 Algorithm 1 :

Support Vector Machine

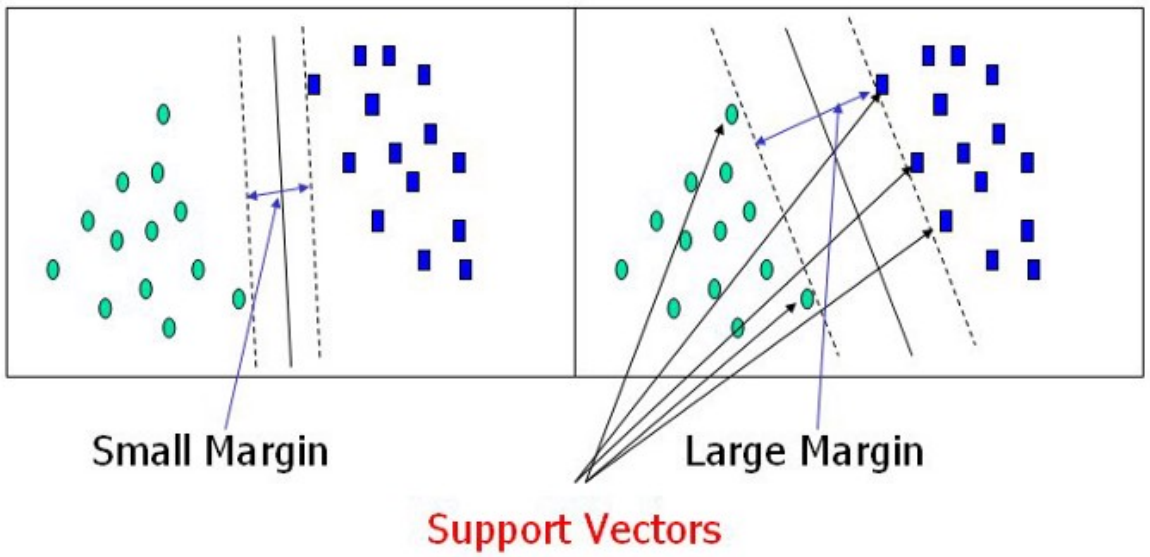
The objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space(N—the number of features) that distinctly classifies the data points.



To separate the two classes of data points, there are many possible hyperplanes that could be chosen. Our objective is to find a plane that has the maximum margin, i.e the maximum distance between data points of both classes. Maximizing the margin distance provides some reinforcement so that future data points can be classified with more confidence.

Hyperplanes and Support Vectors

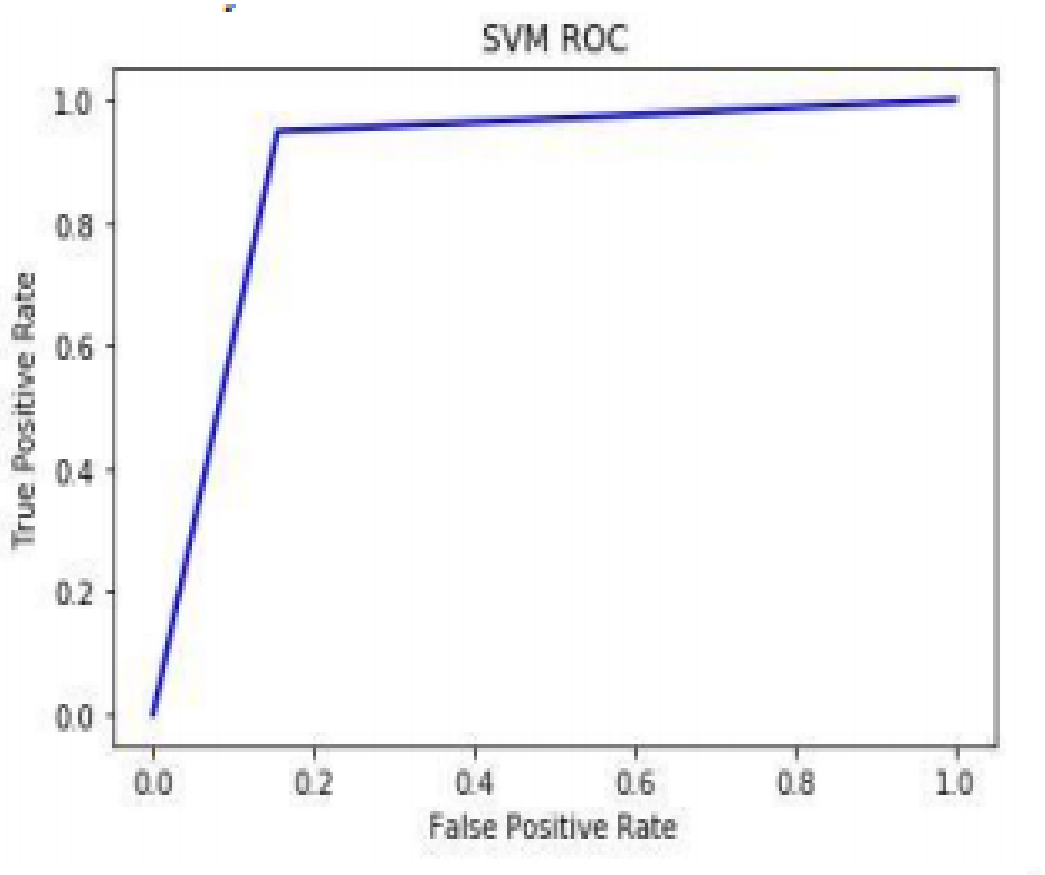
Hyperplanes are decision boundaries that help classify the data points. Data points falling on either side of the hyperplane can be attributed to different classes. Also, the dimension of the hyperplane depends upon the number of features. If the number of input features is 2, then the hyperplane is just a line. If the number of input features is 3, then the hyperplane becomes a two-dimensional plane. It becomes difficult to imagine when the number of features exceeds 3.



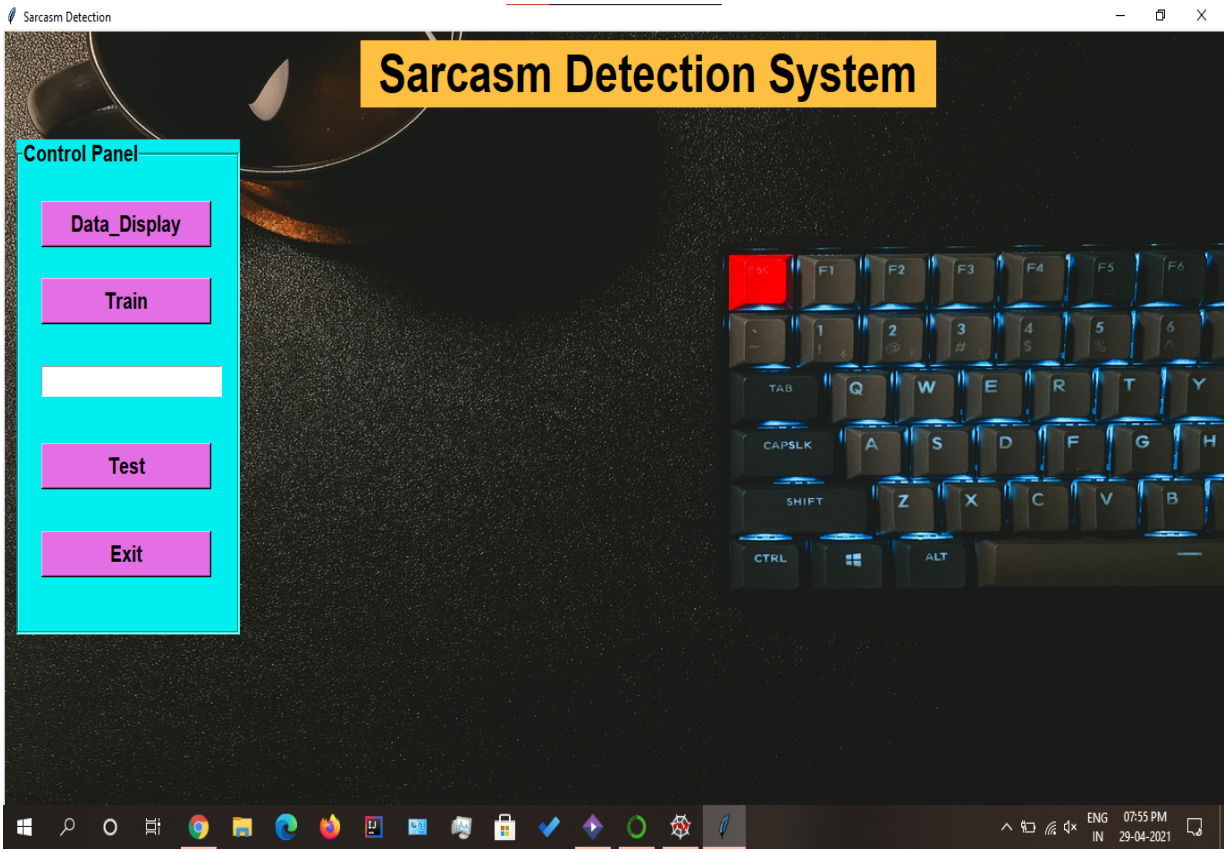
Support vectors are data points that are closer to the hyperplane and influence the position and orientation of the hyperplane. Using these support vectors, we maximize the margin of the classifier. Deleting the support vectors will change the position of the hyperplane. These are the points that help us build our SVM.

Implementation and Output

After preprocessing and Feature engineering we got the well-organized data within that we have trained SVM with 80% data and used 20% for testing purposes and we got 80.02% accuracy.



GUI - SVM Model



ACCURACY - SVM Model

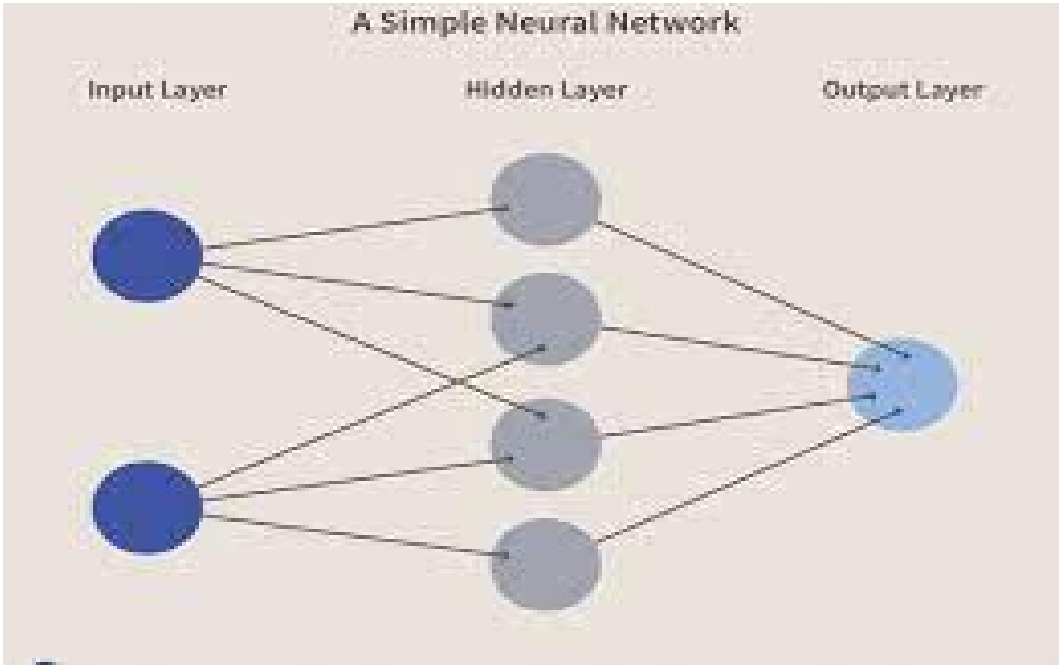
	precision	recall	f1-score	support
0	0.84	0.87	0.85	53
1	0.74	0.69	0.71	29
accuracy			0.80	82
macro avg	0.79	0.78	0.78	82
weighted avg	0.80	0.80	0.80	82

Accuracy : 80.48780487804879%
Model saved as SVM_MODEL.joblib

10.1.2 Algorithm 2 :

Neural Network

A neural network is a series of algorithms that endeavors to recognize underlying relationships in a set of data through a process that mimics the way the human brain operates. In this sense, neural networks refer to systems of neurons, either organic or artificial in nature.
Neural networks can adapt to changing input; so the network generates the best possible result without needing to redesign the output criteria. The concept of neural networks, which has its roots in artificial intelligence, is swiftly gaining popularity in the development of trading systems.

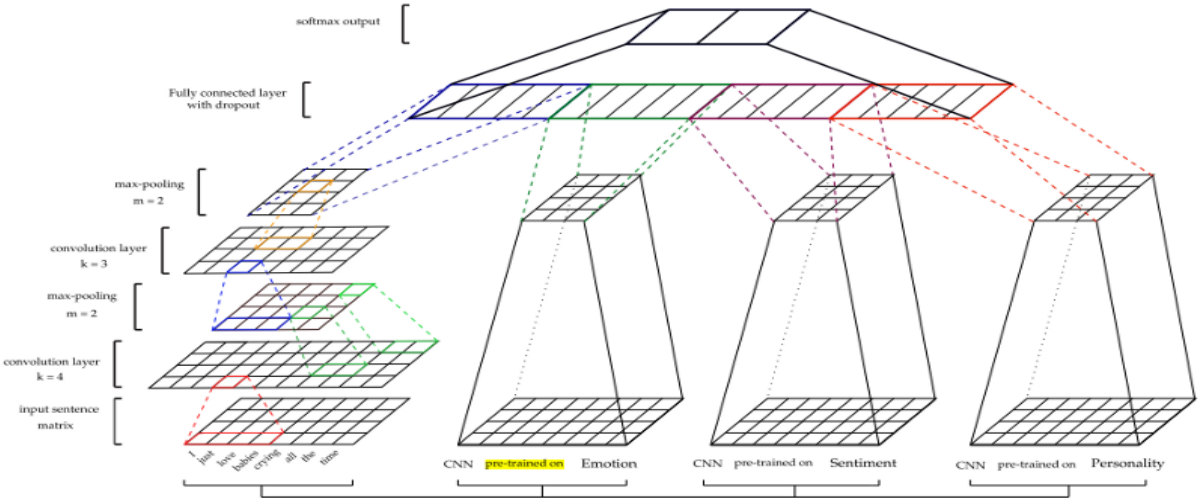


Model

Sentiment shifting is prevalent in sarcasm-related communication; thus, the authors propose to first train a sentiment model (based on a CNN) for learning sentiment-specific feature extraction.

The model learns local features in lower layers which are then converted into global features in the higher layers. The authors observe that sarcastic expressions are user-specific—some users post more sarcasm than others.

In the proposed framework, personality-based features, sentiment features, and emotion-based features are incorporated into the sarcasm detection framework. Each set of features are learned by separate models, becoming pre-trained models used to extract sarcasm-related features from a dataset.



NEURAL NETWORK FRAMEWORK IN SARCASM DETECTION

CONSISTENCY OF NEURAL NETWORK FOR DIFFERENT TRAINING DATASETS

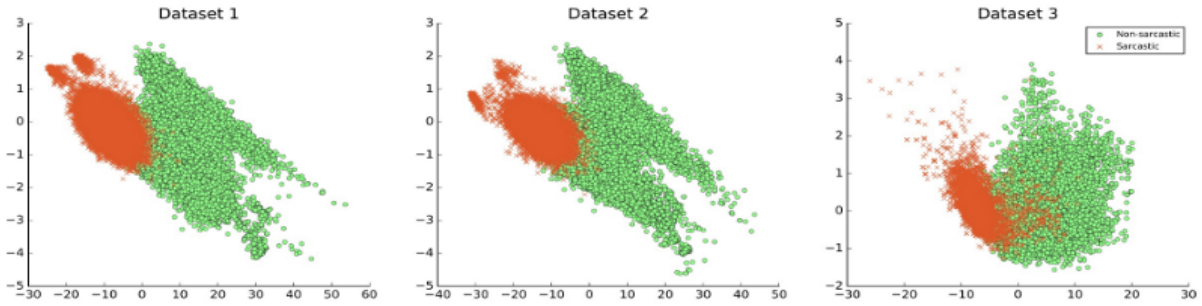


Figure 3: Visualization of the data.

ACCURACY - NN Model

```
Epoch 59 completed out of 80 loss: 836.0741209983826
Epoch 60 completed out of 80 loss: 835.4730886220932
Epoch 61 completed out of 80 loss: 832.2829929292202
Epoch 62 completed out of 80 loss: 827.7675218731165
Epoch 63 completed out of 80 loss: 828.4975971132517
Epoch 64 completed out of 80 loss: 824.667258426547
Epoch 65 completed out of 80 loss: 822.5390749275684
Epoch 66 completed out of 80 loss: 822.3097466230392
Epoch 67 completed out of 80 loss: 819.9452692270279
Epoch 68 completed out of 80 loss: 818.275083899498
Epoch 69 completed out of 80 loss: 815.3631453514099
Epoch 70 completed out of 80 loss: 811.4905333220959
Epoch 71 completed out of 80 loss: 809.5588915944099
Epoch 72 completed out of 80 loss: 809.1157308220863
Epoch 73 completed out of 80 loss: 806.4182283878326
Epoch 74 completed out of 80 loss: 802.3112747520208
Epoch 75 completed out of 80 loss: 803.5293682217598
Epoch 76 completed out of 80 loss: 801.5103746801615
Epoch 77 completed out of 80 loss: 800.2522662580013
Epoch 78 completed out of 80 loss: 799.6876928806305
Epoch 79 completed out of 80 loss: 795.5038438737392
Epoch 80 completed out of 80 loss: 792.6631853878498
F1 Score: 0.6889412634594945
[[2234 5350]
 [1381 7454]]
Accuracy: 0.6400481343269349
Run the command line:
--> tensorboard --logdir=/tmp/logs
Then open http://0.0.0.0:6006/ into your web browser
```

CHAPTER 11

CONCLUSION

11.1 CONCLUSION

Our paper uses a combined approach of different methods like emotion detection, use of emoticons, patterns, etc. identifies whether the social site comment is sarcastic or not. So it is required to use a combined approach which takes different methods and identifies whether the comment is sarcastic or not. The sarcasm identification model is a novel approach based on emotion model. The sarcasm identification model uses different algorithms, libraries And methods in emotion detection phase and its result is used for sarcasm detection.The proposed method makes use of the different components of the tweet. Our approach makes use of Part-of-Speech tags to extract patterns characterizing the level of sarcasm of tweets.

CHAPTER 11

REFERENCES

1. J. M. Soler, F. Cuartero, and M. Roblizo, “Twitter as a tool for predicting election results,” in Proc. IEEE/ACM ASONAM, Aug. 2012, pp. 11941200.

2. U. R. Hodeghatta, “Sentiment analysis of Hollywood movies on Twitter,” in Proc. IEEE/ACM ASONAM, Aug. 2013, pp. 14011404:

3. R:L:J:Brown;“The pragmatics of verbal irony; 00 in Language Use and the Uses of Language: Washington Georgetown University Press; 1980; pp:111127.

4. S. Attardo, “Irony as relevant inappropriateness,” in Irony in Language and Thought. New York, NY, USA: Psychology Press, Jun. 2007, pp. 135174.

5. R.W. Gibbs Jr and J. O’Brien “Psychological Aspects Of irony understanding” J. Pragmatics, vol.

6. H. Grice, “Further notes on logic and conversation,” in Pragmatics: Syntax and Semantics. San Diego, CA, USA: Academic, 1978, pp. 113127.

7. O. Tsur, D. Davidov and A. Rappoport, “ICWSM great catchy name: Semi - supervised recognition uct reviews,” in Proc. AAAI Conf. Weblogs Soc. Media, May2010, pp:162169.

8. A. Joshi, V. Sharma, and P. Bhattacharyya, ”Harnessing context incongruity for sarcasm detection,” ACL-IJCNLP 2015 - 53rd Annu. Meet. Assoc. Comput. Linguist. 7th Int. Jt. Conf. Nat. Lang. Process. Asian Fed. Nat. Lang. Process. Proc. Conf., vol. 2, no. 2003, pp. 757–762, 2015, doi: 10.3115/v1/p15-2124.

9. A. Reyes, P. Rosso, and T. Veale, ”A multidimensional approach for detecting irony in Twitter,” Lang. Resour, Eval., vol.47, no. 1, pp. 239-268, Jul. 2013, doi; 1007/s10579-012-9196-x.

10. M. Bouazizi and T. Otsuki, ”A Pattern-Based Approach for Sarcasm Detection on Twitter,” IEEE Access, vol. 4, pp. 5477–5488, 2016, doi: 10.1109/ACCESS.2016.2594194. College Short Form Name, Department of Computer Engineering 2020 50

11. H. M. K. Kumar and B. S. Harish, ”Sarcasm classification: A novel approach by using Content Based Feature Selection Method,” Procedia Comput. Sci., vol. 143, pp. 378–386, 2018, doi: 10.1016/j.procs.2018.10.409.

12. K. Sundararajan and A. Palanisamy, ”Multi-rule-based ensemble feature selection model for sarcasm type detection in Twitter,” Comput. Intell. Neurosci., vol. 2020, 2020, doi: 10.1155/2020/2860479.

13. Raj, Jennifer S., and J. Vijitha Ananthi, ”Recurrent neural networks and nonlinear prediction in support vector machines,” Journal of Soft Computing Paradigm (JSCP), 1(01), pp.33-40, 2019.

14. S. K. Bharti, K. S. Babu and S. K. Jena, ”Parsing-based sarcasm sentiment recognition in Twitter data,” 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), Paris, 2015, pp. 1373-1380.

15. Shi, Yong, and Aroldis, Japan. ”Multi-dimensional analysis on data sets for information retrieval.” In 2017 International Conference on Intelligent Sustainable Systems (ICISS), pp. 157-162. IEEE, 2017.