

-

An Advanced Hospital Rating System Using Machine Learning and Natural Language Processing

by

Ali Asgar Tamjid

19101363

Gaurab Paul Galpo

19101253

Khadija Begum Urmi

19101261

Fatema Sadeque Chitto

19101592

Sadia Annafi

19101281

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
May 2023

© 2023. Brac University
All rights reserved.

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:



Ali Asgar Tamjid

19101363



Gaurab Paul Galpo

19101253



Khadija Begum Urmi

19101261



Fatema Sadeque Chitto

19101592



Sadia Annafi

19101281

Approval

The thesis/project titled “An Advanced Hospital Rating System Using Machine Learning and Natural Language Processing” submitted by

1. Ali Asgar Tamjid (19101363)
2. Gaurab Paul Galpo (19101253)
3. Khadija Begum Urmi (19101261)
4. Fatema Sadeque Chitto (19101592)
5. Sadia Annaf (19101281)

Of Summer, 2023 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on May 22, 2023.

Examining Committee:

Supervisor:
(Member)



Dr. Amitabha Chakrabarty

Associate Professor
Department of Computer Science and Engineering
Brac University

Co-Supervisor:
(Member)



Mr. Dewan Ziaul Karim

Lecturer
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Md, Golam Rabiul Alam, Ph.D

Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi

Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

Bangladesh is the host of 255 public and 4,452 private hospitals. Unfortunately, there is no reliable metric or resource available online to determine which hospital is better. Patients and their peers often find it difficult to choose the best hospital for their medical attention. The traditional star based rating system can easily be manipulated and they do not take user reviews into count. This is Where this Research and its techniques become useful. Our advanced hospital rating system takes reviews of a hospital and rates it based on the sentiment of the reviews. Our proposed model uses NLP and ML to rate the hospital solely based on the experience of the user shared online. That is why it not only rates the hospital but also identifies the strength and weaknesses of the institution. For this research, 14,443 unstructured reviews were collected from Google Maps of the top 38 hospitals in Dhaka. Additionally, these hospitals were rated based on their review's sentiment and ranked according to the positive percentage. Basically, two types of ranking were introduced where in the general ranking system IBN Sina Specialized Hospital secures the first position and in Class based ranking system Square Hospital secures the first position. Furthermore, a web service is proposed where this trained model predicts the sentiment of the user's reviews and ranks that institution. For future prediction, these reviews were created into multiclass datasets and pre-processed using NLP techniques, and trained into four machine learning models and two deep learning models to predict the sentiment. The most promising model is the Support Vector Machine (SVM) with an accuracy of 85.32%. it's Precision, Recall and F1-score is 86%, 85% and 77% respectively.

Keywords

Hospital Review, Multiclass Dataset, Web Scrapping, Sentiment Analysis, NLP, Machine Learning, Deep Learning, SVM, Logistic Regression, Random Forest, Decision Tree, BERT, CNN, Hospital Ranking, Sentiment Prediction, Web application.

Acknowledgement

All glory to God, who has enabled us to finish our thesis without any significant setbacks. We appreciate the general direction provided by our Supervisor Dr. Amitabha Chakrabarty Sir and co-advisor Dewan Ziaul Karim Sir Without their help, we would be unable to complete our project.

Table of Contents

Declaration	i
Approval	ii
Abstract	iv
Acknowledgment	iv
Table of Contents	vi
List of Figures	viii
List of Tables	ix
Nomenclature	ix
1 Introduction	1
1.1 Introduction	1
1.2 Research Problem	2
1.3 Research objective	2
1.4 Ranking	3
2 Literature review	4
2.1 Overview	4
2.2 Related Works	4
2.3 Sentiment Analysis	7
2.4 Algorithm Used	8
3 Dataset	9
3.1 Dataset Collection	9
3.1.1 Selenium	11
3.1.2 Chromium driver	12
3.2 Data description	12
4 Methodologies	14
4.1 Data preprocessing	14
4.1.1 Removing Emojis	14
4.1.2 Removing Blank rows.	14
4.1.3 Removing Punctuation Mark	14
4.1.4 Tokenization	16

4.1.5	Removing Duplicate Reviews	16
4.1.6	Lemmatization	16
4.1.7	Stemming	16
4.1.8	Oversampling	16
4.1.9	Text Vectorization	16
4.1.10	Under sampling	17
4.1.11	Lower Casing	17
4.2	Word Cloud	17
4.3	Decision Tree Classifier	19
4.4	SVM	20
4.5	CNN	21
4.6	Logistic regression	23
4.7	BERT(Bidirectional Encoder Representations from Transformers) . .	24
4.8	Random Forest	26
5	Result Analysis and Evaluation	28
5.1	Overview	28
5.2	Accuracy	28
5.3	Precision	28
5.4	Recall	29
5.5	F1 Score	29
5.6	Confusion Matrix	29
5.7	Machine Learning Approach	30
5.7.1	Support Vector Machine	30
5.7.2	Random Forest Classifier	31
5.7.3	Logistic Regression	32
5.7.4	Decision Tree Classifier	33
5.8	Deep Learning Approach	34
5.8.1	Bidirectional Encoder Representations from Transformers (BERT)	34
5.8.2	Convolutional Neural Network(CNN)	35
5.9	Ranking	37
5.9.1	General Ranking	38
5.9.2	Class Based Ranking	38
5.10	Comparison of the Accuracy of Algorithms	39
5.11	Rating system generated by SVM	40
5.12	Hospital Ranking web application	42
6	Conclusion and Future Works	43
6.1	Conclusion	43
6.2	Future Works	43
	Bibliography	46

List of Figures

4.1	Work Flow diagram	15
4.2	Positive Word Cloud	18
4.3	Negative Word Cloud	18
4.4	Mixed Word Cloud	19
4.5	Decision Tree Classifier	20
4.6	Performance Chart of Logistic Regression	21
4.7	Standard-CNN-on-text-classification	22
4.8	Logistic Regression	24
4.9	Bert diagram	25
4.10	Random Forest	27
5.1	Confusion Matrix of Support Vector Machine	30
5.2	Performance Chart of Support Vector Machine	30
5.3	Confusion Matrix of Random Forest	31
5.4	Performance Chart of Random Forest	31
5.5	Confusion Matrix of Logistic Regression	32
5.6	Performance Chart of Logistic Regression	32
5.7	Confusion Matrix of Decision Tree Classifier	33
5.8	Performance Chart of Decision Tree Classifier	33
5.9	Confusion Matrix of BERT	34
5.10	Performance Chart of BERT	35
5.11	Validation Accuracy of BERT	35
5.12	Validation Loss of BERT	35
5.13	Confusion Matrix of CNN	36
5.14	Performance Matrix of CNN	36
5.15	Validation Loss of CNN	37
5.16	Validation Accuracy of CNN	37
5.17	Model Accuracy Comparison	39
5.18	Unlabelled Reviews	40
5.19	Predictive hospital reviews	41
5.20	Predictive hospital rating	41
5.21	Review page of web application	42
5.22	Ranking page of web application	42

List of Tables

3.1	Table to test captions and labels.	13
5.1	Confusion Matrix	29
5.2	General Hospital Ranking	38
5.3	Class Based Hospital Ranking	39
5.4	Model Evaluation Table	40

Chapter 1

Introduction

1.1 Introduction

Considering [1] both the urban and rural areas, Bangladesh has around 975 private healthcare facilities and 3976 public hospitals. If we compare with other Asian countries, it indeed has a good healthcare network with so many hospitals and clinics yet the consumption of healthcare is very low here in Bangladesh. According to a Canadian survey conducted by the CIET (Centre for International Epidemiological Training); 13 out of 100 people in our country, who seek treatment, often go to government hospitals aka public hospitals. 27 of 100 choose private hospitals or NGO[24] health care services and 60 out of 100 people consume unqualified health care services [17]. We can see the number of users for unqualified services is very high whereas the number of users for public services is very[23] low. Another Government study[22] that was conducted by HEU (Health Economics Unit) of the MOHFW Bangladesh (Ministry of Health and Family Welfare), that claims- the lack of management or long waiting times, travel time, scarcity of medicine, unavailability of nurses, lack of empathy inexperienced doctors, lack of professionalism in staff etc. play a vital role when it comes to the low number of users of public health care services. Due to the poor quality of health care services, Bangladeshi patients are slowly growing a tendency of seeking foreign medical care. Not only this, people seldom know if there is any Bangladeshi hospital that can provide up-to-the-mark medical services. There is no system of even finding the same. During our research, we figured out that around 500 million USD is spent by Bangladeshi people on foreign medical purposes and when we tried to investigate the reasons behind this we find out that there is no structured system that can identify the feedback of patients and identify this problem. The most resourceful platform that contains user reviews of hospitals is Google Maps where the traditional star based rating system is implemented and it does not take user sentiment of remarks into consideration for rating. But from our research, we have realized the institution with more reviews has more trustworthiness but these reviews have been ignored in the google map's rating system. The current traditional scale-based rating system is not constructive and it does not rank a hospital. rating is cardinal it can not compare between two hospitals. There is where our research comes from, we are building a model so that it minimizes the inconvenience of patients. Our proposed model Rates the hospital based on the emotions and sentiments of the reviews. This emotion-based rating system is created for this research and this rating is used to compare and rank the

hospitals which are ordinal. These ordinal values are very important in terms of emergencies. It helps people to take decisions quickly.

1.2 Research Problem

We started our journey with the most[26] renowned hospitals of Dhaka city. for example Square Hospitals Ltd, LABAID Specialized Hospital etc. Initially, the thing which bothered us most is that the medical field is very very multidimensional, some hospitals provide better treatment for gynecology, some hospital is well known for their neuroscience department, some hospitals have got better physicians in the case of oncology, and so on. But when it comes to the reviews there were no constructive reviews based on different fields and it's unavailable on the internet moreover, there is no synchronized data set based on our topic and we could not find enough relevant data in emr(electronic medical record) and ehr(electronic health record), so our first challenge was data collection and decorating a constructive dataset. To collect the data set we built a web scraper using the Selenium test automation tool of Python. Basically, we started our thesis during the period of covid. As a result, we might get poor reviews of those hospitals that didn't provide quality services to covid affected patients. Still, we were able to manage more than 14443 of data initially. Later on, some comments were eliminated because some of them were just emojis, repeated comments, integer values, blank values and interpretations using different languages like typing Bengali words in English which really creates a confused medley.

1.3 Research objective

Every hospital has been known to use these rankings to establish objectives and evaluate performance[2]. Artificial intelligence and machine learning are crucial in the case of rating systems. AI can anticipate scores on surveys that go unanswered as well as after any response, review, or interaction if it is trained on recent and historical events. Machine learning will essentially assist us in analyzing the components required for constructive rating because, as we know, it does some specific tasks based on statistics and data without explicit interactions. Additionally, (ML and NLP) will assist us in discovering hidden patterns to increase the precision of our predictions. Here, we will use text mining algorithms and natural language processing to analyze customer and user reviews left in comments on various websites, including social media.

- Design a system for better patient care: Statistics and data about hospitals must be included in the design so that patients can easily find the desired hospital.
- Reduce hospital operating costs: Embrace the adaptation of technology and reduce duplication of tests and other services. This may reduce operating costs.
- Hospital Review: We will employ our natural language processing knowledge in order to get the people's opinions, sentiments, and emotions from their text for different hospitals.

- Improving the quality of healthcare: It's difficult to know where to seek treatment[3] when you have a health emergency. They might start their investigation of hospitals and doctors by comparing them to other medical institutes using our research. The top hospital may not always be the best option for all whether it's too far away to justify the trip expenses or it's not in their insurance network. It's a good idea to check for the highest-ranked hospital that satisfies their requirements. Patients with special requirements or interests may look for hospitals that are well-ranked in those areas. For illnesses such as heart disease, cancer, and Parkinson's disease, the specialized rankings can serve as a starting point.
- Spontaneous Improvement by Hospital Authorities: Since we will be ranking the hospitals, so the ranks of hospitals won't be static, it will fluctuate accordingly. One hospital can surpass the other depending on the parameters of ranking. So let's imagine that the hospital which was on 3rd position in the list now stands 1st due to their enhancement which enabled that particular hospital to fulfill the necessary criterias embedded in the system. As a result those hospitals which lagged behind will spontaneously point out their deficiencies, flaws etc as a result they will work to improve their current condition try their level best to get back to their previous position. It might led to a competitive environment between the existing hospitals but it will make the authorities worried about the quality of their services as well.

1.4 Ranking

Our ranking process consists of two categories:

- (1)General Ranking
- (2)Class-Based Ranking

In the case of class-based ranking, the hospitals were divided into two classes based on the number of reviews. The superior class range was (2000-1000), the inferior class range was (999-390) and the superior class stood higher in rank. Amongst the superior-class hospitals, the hospital which had the highest percentage of positive reviews achieved the first position. In class-based ranking Square Hospital obtained first place. In the case of general ranking the hospitals were not differentiated based on their class. The ranking was dependent on the percentage of positive reviews. In the case of general ranking IBN Sina Hospital acquired the first position. Our methodology includes two types of ranking so that the hospitals with lower no of reviews don't face discrimination because hospitals with fewer no of reviews can provide the same quality services as those which have higher no of reviews comparatively.

Chapter 2

Literature review

2.1 Overview

Treatment is the fundamental[15] right of any human being. The health industry's major purpose is to provide proper treatment to the general public, and hospitals are the most important place to acquire these services. So, it is very important for a patient to be admitted to a good hospital, and the hospital rating system helps to find the best hospitals in the city. But the rating system we know today mainly uses Stars and SERPs rating systems. Star ratings on Google and other social media are powered by consumer reviews, where they use an algorithm and an average to determine how many stars are displayed. So these stars and comments do not actually provide honest, constructive feedback. But in a constructive review system, we will be analyzing the comments and feedback of the patients from various sites and social media about the hospitals so that there will be no problem or dilemma in choosing the exact hospital they need, and the patients' complacency will determine the rank. of the hospitals. The rank will not be static, it will be dynamic, and the rank of those hospitals may inflate or deflate based on the service they provide. In this section, we have explored summaries of various articles that relate to our research work. We will write up a summary of these research papers. We also observe how other papers implement the different types of algorithms in their works. We will describe how the other research paper uses the Decision Tree, CNN, BERT,Random Forest,SVC(Support Vector Classifier) and logistic regression algorithms in sections 2.3

2.2 Related Works

In[7] this paper the researcher used predictive and prescriptive analysis[31] of health care. Machine learning underpins predictive analytic, which entails developing algorithms[29] to extract knowledge from existing data, combine it, and assume the future. These algorithms can be divided into two categories. One of these algorithms is called supervised learning[35] and another is called unsupervised[28] learning. For both breast cancer and non-breast cancer diagnoses, their report was 100% accurate. Moreover, To estimate the survival rate of patients with oral cancer, researchers[27] used data mining techniques[32] based on logistic regression with an accuracy of 95.7%. Furthermore, on a heart disease prediction, they have an accuracy of 93.02%. Here[8] The researchers demonstrated how artificial intelligence and

machine learning can be used to improve the healthcare industry. They stated that machine learning may be used to predict the pharmacological qualities of chemical molecules and therapeutic targets. On medical images, pattern recognition, and segmentation algorithms are used to enable faster diagnoses and illness tracking. The accuracy of their claim is undetermined in the paper. On the other hand, in[9] another paper the author talked about the elements of artificial intelligence and how the collaboration between AI and humans can help us improve the health industry. Artificial intelligence, voice technology and assistants, natural language processing, and machine learning all help the healthcare industry in a positive way. The author also talked about medical robotics where he predicts in the future the robots will take care of the patients. However, the accuracy of their claim is undetermined. In [33] they have used EMR (electronic medical records for data processing. They linked with VASQIP cases in the instance of Database, which is an administrative database that contains records on all discharged veterans from VA hospitals. In this type of database, the linkage is linked/ based on the patient's identifier code. For additional measures, they implemented Patient Safety Indicators. In the case of accuracy, 7 the rate is 95%. In[11] and[12] both researchers used NLP. In the case of (Comparative Study of Machine Learning Algorithms for Hospital Rating System) the study was related to SVM- based on machine learning k-means cluster learning whereas the (Data processing and text mining technologies on electronic medical records: a review. Journal of Healthcare engineering, 2018) paper was regarding data processing text mining. In the case of (Data processing and text mining technologies on electronic medical records: a review. Journal of healthcare[25] engineering, 2018) the researchers not only used the NLP but also, used SVM algorithm linear SVM which provides them a method of decision-making function. In Data processing and text mining technologies on electronic medical records: a review. Journal of healthcare engineering, 2018) the factors were precision, recall, support etc. Both papers have an accuracy rate of 100%. In[13] and both researchers[14] used predictive models for their data. In (Harnessing the Power of artificial intelligence) a predictive model was used for the generation of alerts from a large amount of patient population. In (Hospital Facebook Reviews Analysis Using a Machine Learning Sentiment Analyzer and Quality Classifier. In Healthcare (Vol. 9, No. 12, p. 1679)) this model was used for accuracy rating. Besides in (Hospital Facebook Reviews Analysis Using a Machine Learning Sentiment Analyzer and Quality Classifier. In Healthcare (Vol. 9, No. 12, p. 1679)) SERVQUAL scale was used which consists of 5 dimensions; which are used to determine the accuracy of feedback on the websites. In the paper (Harnessing the power of artificial intelligence) Ehr (Electronic health records] was used as a data source. In the case of Ehr, the accuracy of predictive analysis is 82% and in the case of SERVQUAL scale, the accuracy rate is 83.5%. Here in[16] [34] [30] the both researchers used the linear regression model in order to model a targeted/ predicted value based on independent variables. In the research (Measuring patient-perceived quality of care in US hospitals using Twitter. BMJ quality safety, 25(6), 404-413.) HCAPS model was used to measure the perspective quality of patients and they also used Pearson correlation to assess the linear relationship for statistical analysis. In the paper (Predictive and prescriptive analytics in healthcare: a survey. Procedia Computer Science, 170, 1029-1034) they used supervised and unsupervised models to train their data. In both the researches the rate of accuracy cannot be calculated or neither be dragged to any conclusion.

Here[17] in the research paper the researchers have Used patient feedback to improve healthcare service quality. They have gathered data sets on different departments of medical discipline such as Cardiologists, Oncologists, Neurologists, etc. They applied Stanford CoreNLP tools which have helped them to filter out text-based input In an attempt to give these raw data structures, transforming it to the appropriate format for further analysis. They've also covered the Deep Learning algorithms that were used to handle sequential data. They also have adopted Multimodal features fusion so that text and visual features can be extracted. Additionally, they implemented a standardized BOWs method by pre-processing data by deleting stop words, tokenizing, and stemming. However, their LSTM models have an accuracy 8 of 95.12% which is better compared to others. For image-only data set DL-CNN models achieve 83.22% accuracy. Finally, their multimodal features fusion gives 97.75% accuracy. Overall, their reports were well-defined and understandable. In this[18] study the researchers Estimated hospital admissions via text mining. Their source for the data set was past medical information from the emergency department. They created a Pre-Processing Module that contains six phases, including acquiring informative text, normalization, tokenization, feature selection, conversion to set-of-words, and splitting the data between training and testing sets. They also used text mining to analyze hospital consulting records. Finally, modules such as Random Forest, Decision Tree, AdaBoost, Logistic Regression, Extremely Randomized Tree, Multinomial Naive Bayes, and Support Vector Machines have been developed for predicting. The most effective text mining approach was the Nu-Support Vector machine. With a 0.66 standard deviation, It scored 77.70 percent on the F1 scale in predicting hospitalization. Here[19] the researchers Identified patient satisfaction with healthcare services received by An approach to natural language processing. They have used qualitative methods for sentiment analysis. They have also used natural language processing to manage the data. Moreover, they developed an annotation schema for topics and sentiment recorded in free-text satisfaction responses. Moreover, they have trained a Naïve Bayes (NB) classifier using the 300 adjudicated documents. However, the overall F-score of this system was 84%. In this[20] report the researchers Used of Sentiment Analysis of patients online. They have sources of data from websites and social media. They have used Weka data-mining software on 6412 online comments. Here the machine learning approach mainly used two components such as pre-processing, and classification. For classification, Naïve Bayes multinomial, Decision trees, Bagging, and Support vector machines were used. Their accuracy is respectively 88.6% , 80.8%, 82.5%, 84.6%. Here[21] the research is all about improving care/treatment in hospitals by using artificial intelligence. They have used EHR like all the previous papers. A hierarchical Bayesian model and the time series topic model were used to analyze the time series from the child's initial few hours after birth., and the time series topic model, Ms.Suchi Saria, and her colleagues created physiological evaluation ratings for the newborn. The above-mentioned strategy aids healthcare workers in determining the precise risk factors associated with newborns. Though their accuracy cannot be counted in percentage, it's undoubtedly an accurate method to improve the quality.

2.3 Sentiment Analysis

Sentiment analysis is the technique used in natural language processing (NLP) to determine the emotional undertone of text. A piece of writing can be analyzed by sentiment analysis software to discover the author's viewpoint on a particular issue, product, or service. It is also known as opinion mining. The mining of data, deep learning, AI, and computational linguistics are all applied in this process. Sentiment analysis is a highly effective way to very rapidly and simply derive understandings from a huge quantity of text data. By using the AL tool, the text can detect whether it is positive, negative, or neutral. Nowadays, human interacts most of the time with Social Media because anyone can share their emotions, and idea publicly. If we want to take service to any hospital, restaurant, or shop, we check the review from the website or social media. Facebook, YouTube, Google Maps, Amazon customer reviews, Daraz, Shajgoj, Zomato, etc are popular sites where the customer can give their opinion. Organizations can obtain information about how consumers feel, interactions with clients, and the image of the brand through sentiment analysis technologies. By using the technique of sentiment analysis, it is possible to identify the subject[10], the perspective of the applicant, and polarity the degree of passivity and negativity in a text. Marketers can configure the software to give an alert when a negative word is detected. The computer program can be initially configured by the marketer that notifies them when a negative word has been identified. Based on actual and detailed client feedback, Businesses upgrade their products or services according to the satisfaction of a client. Sentiment has been employed in a wide variety of fields. Based on online customer reviews, sentiment analysis in the healthcare industry can help clients choose the most trustworthy hospitals. Machine learning algorithms are used in sentiment analysis to analyze literature written using conventional languages. Collecting the data, cleansing the data, and feature extraction is often done before sentiment analysis. The piece of writing is then scored according to the specific emotion it demonstrates using a machine learning system.

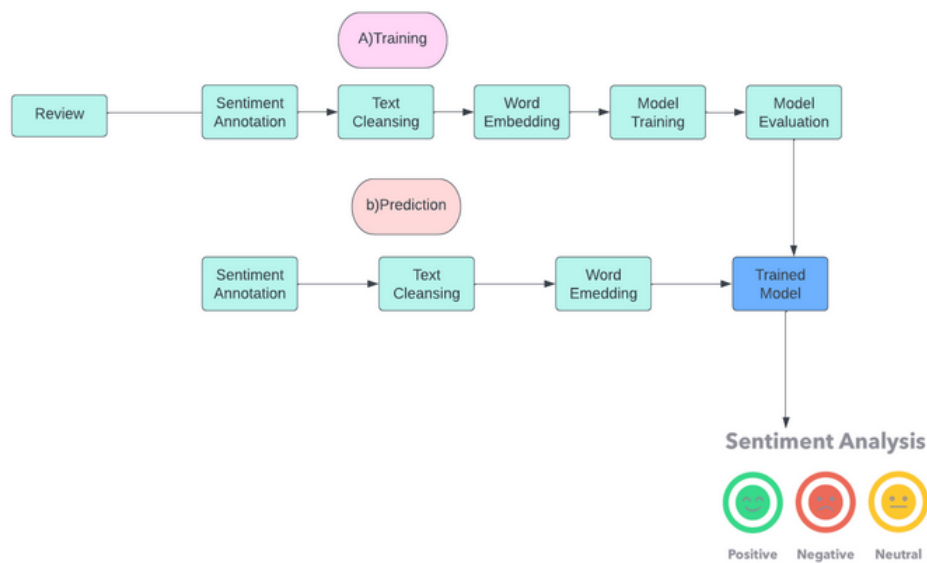


Figure 2.3: Sentiment analysis diagram

2.4 Algorithm Used

BERT:

BERT is capable of understanding a word's context from either direction of its word neighbors[6]. This tremendous advancement in machine learning opens the door to a number of potential sentiment analysis applications. By concurrent conditioning on both left and right context in all layers, BERT is able to produce "deep bidirectional representations from unlabeled text

CNN:

Convolutional neural networks are used in text analysis. CNN has a convolutional layer to extract information from a larger chunk of text. We develop a straight-forward convolutional neural network model and evaluate it using benchmark data. The outcomes show that it outperforms more established techniques, such as the SVM and LSTM, in classifying any kind of sentiment.

Logistic Regression:

The logistic regression classification approach solves the binary classification problem. The outcome in models with a double situation is often either 0 or 1. Estimation is performed using binary classification and logistic regression on the training data.

Decision Tree:

A generic, predictive modeling approach[4] with applications in many different fields is decision tree analysis. In general, decision trees are built using an algorithmic method that determines how to divide a data set based on several criteria. It is among the most popular and useful techniques for supervised learning.

SVM:

SVM means Support Vector Machine. Due to its capacity for high accuracy while maintaining computational efficiency, SVM's are a preferred option for many machine learning applications. Finding the ideal boundary between the various classes is how SVM's identify data points in a high-dimensional space.

Random Forest:

A widely prominent algorithm employed in the machine learning methodology is called Random Forest. In the training stage of the random forests or random choice forests ensemble learning approach, which is used for classification, regression, and other tasks, several decision trees are constructed.

Chapter 3

Dataset

In this section we have thoroughly discussed our dataset how we collected the raw data, how we preprocessed it, labeled it based on emotions along with our proposed model structure and code implementation part.

3.1 Dataset Collection

Dataset collection is the core part of the research work[5]. For our study, we acquired second-party quantitative data conveyed in words, which we then analyzed with interpretations and categorizations. The idea we had was to develop a methodology that would identify the best hospital based on factors including public satisfaction, experience, evaluation, and more. Because of this, we placed a special emphasis on gathering data and devoted the majority of our time to it.

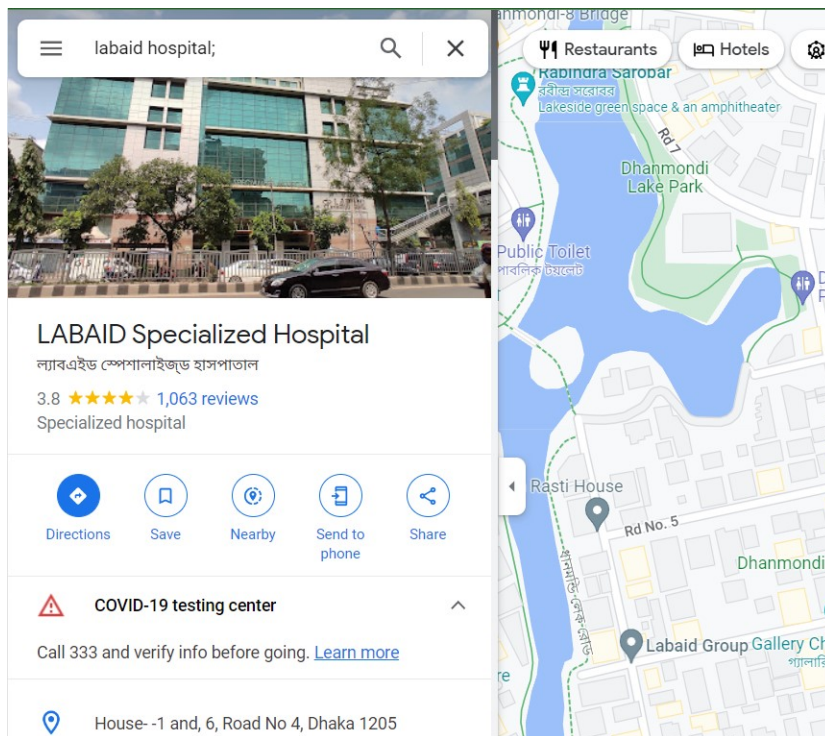


Figure 3.1: The Google Map Review of LABAID Hospital (1063 reviews)

First of all, we created a customized dataset named "Dhaka Private Hospitals Review." We aim to produce a trustworthy and authentic dataset that will be accessible to everyone. The only profound sources were the Facebook comment section of different hospital pages, and Google Maps reviews of other social platforms including the websites of different hospitals. Out of all these sources, scraping the patient or customer review from Google Maps is the most feasible and resourceful one because, from our country's perspective, people tend to leave hospital reviews on Google Maps. For example, LABAID Hospital and Square Hospital have respectively 1063 and 1945 reviews on Google Maps. To be precise, the reason for using the Google map reviews as our data was simply because it has comparatively more patient or customer reviews than other Internet sites. Manually converting the last three types of reviews into English is time-consuming. That's why, during the extraction time, with the help of Google Translator on Google Map pages of different hospitals, we translated all reviews into English. As a result, the scraping tool only allowed us to extract English comments, and we created the actual dataset. There were four different types of patient or customer review that we extracted. They are-

- Solid English- Good health services provider.
- Solid Bengali-ডাক্তারের সেবা আলহামদুলিল্লাহ ভালো।
- Bengali and English Mixed- ব্যায়বহুল চিকিৎসা কেন্দ্র, Expensive medical center.
- Banglish- Valo. Tobe kichu kichu doctors onektay arrogant.

Figure 3.2: Review Type

Manually converting the last three types of reviews is time-consuming. That's why during the extraction time, with the help of Google Translator of Google Map pages of different hospitals, we translated all reviews into English. As a result, the scraping tool extracted only English comments for us. It kept the translation information of reviews as well so we were able to check the translated version in case any changes were needed there. In Figure 3.1 we can notice four types of reviews. Solid English, solid Bangla, Bangla and English and Banglish. These are translated by the Google Maps Translate method to turn into solid English Reviews. These Reviews are then stored for Further pre-processing works.

Any method used to collect data from the world wide web is referred to as "web scraping". Numerous uses for this data are possible, including market research and the gathering of product reviews for analysis. Users can quickly and easily obtain product reviews, pricing details, and other data from websites by employing web scraping. Researchers that need to get information from large websites might save time and effort by using web scraping. Python is the most popular language used to create web scrapers because it includes a number of libraries created especially for the task. We collected private hospital reviews in our dataset, it will be very time consuming if we collect reviews by copying and pasting from google maps. For our convenience, a web scraper was developed a web scrapping tool by using Python's Selenium package and Chromium driver for collecting reviews. Our web scraper can

collect reviews from the google maps website. In our Python code for web scraping, when we run our code chromium driver leads to that hospital's review page. Then the selenium package was used to extract the data. By using the "find element by class name" method of selenium to review was accessed. In "wi17pd" class all the reviews of google maps are contained. We have extracted the ".text" file from that class and stored it in a .csv file. in figure 3.2 the code is illustrated. We can divide this task into 8 stages:

1. Hospital Selection
2. Review Translation
3. URL Generation
4. Inspection of HTML page
5. HTML class name identification
6. Coding and implementation
7. Reviews Collection
8. Data saving in a . CSV file

We used the above mentioned Python libraries to implement our web scraping. The Selenium library is utilized for scraping comments from websites. It has the capability of automating the browser operations. The Python programming language's BeautifulSoup module can be used to parse HTML documents and produce parse trees. When trying to retrieve the data, these parse trees are helpful.

We first tried to scrape the data using the API of Google Maps, but we were only able to scrape five reviews. We discovered some expensive paid web scrapers. Later, we created a web scraper tool to extract raw data by using the Selenium and Pandas packages of Python. These are used to collect the HTML unstructured and structured text data from Google Maps. So far, we have extracted customer reviews of 38 Bangladeshi hospitals and have plans to collect reviews of more hospitals in the future. The total review combining all 38 hospitals is 15379.

Manually converting the last three types of reviews into English is time-consuming. That's why, during the extraction time, with the help of Google Translator on Google Map pages of different hospitals, we translated all reviews into English. As a result, the scraping tool only allowed us to extract English comments, and we created the actual dataset.

3.1.1 Selenium

For scraping Google Maps reviews, the Selenium package, a well-liked Python web automation and testing framework, can be quite helpful. Google Maps' dynamic nature and substantial use of JavaScript make it difficult to scrape because of these factors. But by enabling us to automate browser interactions and retrieve the needed data, Selenium offers a potent option. Selenium can be used in conjunction with a web driver, such as the Chrome driver, which acts as a bridge between the browser and Selenium, to scrape reviews from Google Maps. We can access Google Maps, look for a certain address, and read reviews by starting an automated Chrome browser session with Selenium.

We may use Selenium's several techniques to find and extract the review elements once we are on the desired page. We may, for instance, use techniques like "find elements by class name," "find elements by XPath," or "find elements by CSS selector" to locate and obtain the review elements on the page. Additionally, Selenium

gives us the ability to interact with the page and dynamically load more reviews as needed. In order to replicate user behavior and make sure that all reviews are loaded and ready for extraction, we can automate scrolling activities and clicking on load more buttons. We can further process the data to recover the exact review text, reviewer data, scores, and any other pertinent information after extracting the review elements. Selenium’s adaptability enables us to efficiently explore the HTML structure and extract the needed data.

In conclusion, the Selenium package offers a potent option for scraping evaluations from Google Maps when used in conjunction with an appropriate web driver like the Chrome driver. It is a useful tool for obtaining reviews and utilizing them for analysis, study, or any other purpose due to its capacity to automate browser operations, find elements, and get data.

3.1.2 Chromium driver

A crucial element that enables easy Selenium integration with the Chromium browser and effective web automation and scraping activities is the Chromium driver. The Chromium driver serves as a link between the Selenium WebDriver API and the Chromium browser when using Selenium in Python. The Python automation script and the Chromium browser instance are connected through a communication channel that the Chromium driver creates. It makes it easier for instructions and information to be sent, allowing Selenium to programmatically control the browser.

Developers can automate a variety of browser tasks, including going to URLs, interacting with web elements, submitting forms, and retrieving data from online pages, by using the Chromium driver. The driver efficiently converts Selenium commands into particular Chromium actions, ensuring smooth automation task execution. Developers must obtain and install the correct version of the Chromium driver that corresponds to their Chromium browser in order to utilize it with Selenium in Python. When the driver is ready, it may be created within the Python script to connect to the browser and make it possible to run automation instructions.

By utilizing the properties of the underlying Chromium browser, such as its quick rendering engine, support for JavaScript execution, and compatibility with contemporary web technologies, the Chromium driver expands Selenium’s functionalities. A reliable and effective solution for web scraping, testing, and other automated activities is provided by this combo. In conclusion, the Chromium driver acts as an essential middle-man component, enabling flawless Chromium browser control via Selenium. Developers can harness the power of automation and carry out a wide range of operations proficiently and effectively by utilizing the advantages of Chromium through the driver.

3.2 Data description

Data Labelling

To train ML or NLP models, data labeling is the most valuable step. It helps algorithms to build accurate environmental understanding. The term Data labeling refers to the procedure of assigning labels to data so that it can have context as well as interpretation. In our research case, we need to preprocess these hospital reviews

of customers according to the level of emotions to prepare the dataset for sentiment analysis. So that the model doesn't lack confidence because of the highly valued dataset. we manually labeled it.

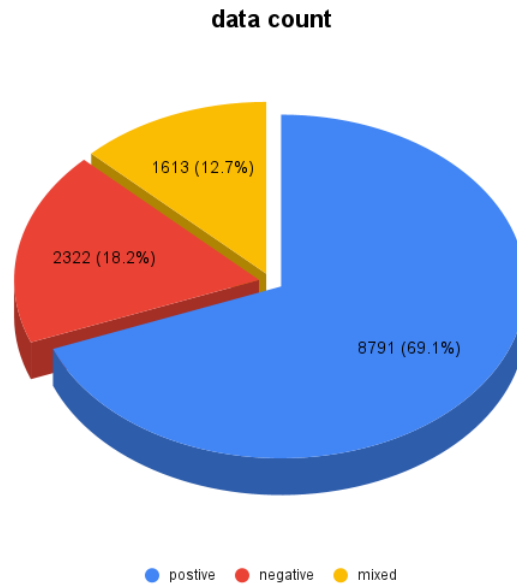


Figure 3.5: Pie chart of Sentiments

From our observation, We have noticed that a patient or client's review may seem positive to someone and may seem negative to others. To avoid that we labeled the dataset with 3 sentiments. We classified those into 3 types of emotions as positive, negative, and mixed. Out of 12726 labeled reviews, there are 8791 positive reviews, 2322 negative reviews and 1613 mixed reviews which is illustrated in the table 3.1.

total	target	data count	percentage
	positive	8791	69.08%
12726	negative	2322	18.25%
	mixed	1613	12.67%

Table 3.1: Table to test captions and labels.

After collecting data, there are found distorted and unequal datasets. In This type of dataset, one type of category contains a relatively larger number of cases than the others. Machine learning algorithms often struggle to work with imbalanced datasets since they are frequently programmed to perform best when there are nearly equal numbers of samples for each class. It is possible to use numerous approaches for balancing a dataset such as undersampling, oversampling, Hybrid methods, etc. Oversampling techniques were applied to equalize the categories data in our collection by increasing the number of the minority of categories. To balance our data, we have raised the proportion of positive and negative reviews and randomly paraphrased the data. Then we marge our newest negative and mixed review in the dataset. We received 21075 reviews after oversampling, which we used to train our models.

Chapter 4

Methodologies

4.1 Data preprocessing

Only collecting the data is never enough to build an ML model because there is inconsistency, punctuation, emotions, noise etc. in real-world data. If we use data without pre-processing then these factors work prevent to the predictive models from being used directly. In fact, if we want to improve the performance and accuracy of the trained model then data pre-processing is required for data cleaning and preparation for a classification model. A model's performance depends highly on how the data preprocessing was done. For example, when we run the Decision tree algorithm on our dataset without cleaning any data the accuracy came as 10% whereas after cleaning and preprocessing, the accuracy increased to 80% . The preprocessing stages that we performed for our dataset are discussed below.

4.1.1 Removing Emojis

People tend to use emojis to express their emotions. Since we used the first hand user review, there were a significant amount of emojis. We found reviews where people used “heart” emoji to express their positive experience for a hospital. We also encountered reviews where people used “thumbs down” emoji to describe how they didn't like a hospital's service. We removed all types of emojis from our dataset.

4.1.2 Removing Blank rows.

After removing emojis or repeated comments there were blank rows in between the horizontal columns. To remove the blank space we used the code `Reviewdata = Reviewdata[Reviewdata['Review'].notna()]` which removes the unnecessary blank space and makes the dataset look better.

4.1.3 Removing Punctuation Mark

Punctuation removal is an important NLP preprocessing step. For better results, we acknowledged 32 types of punctuation which are `'!'"()*+,-./:;|=i?@[‘— ’`. Since 17 these punctuation marks have an impact over text processing approach, we have eliminated these from our dataset.

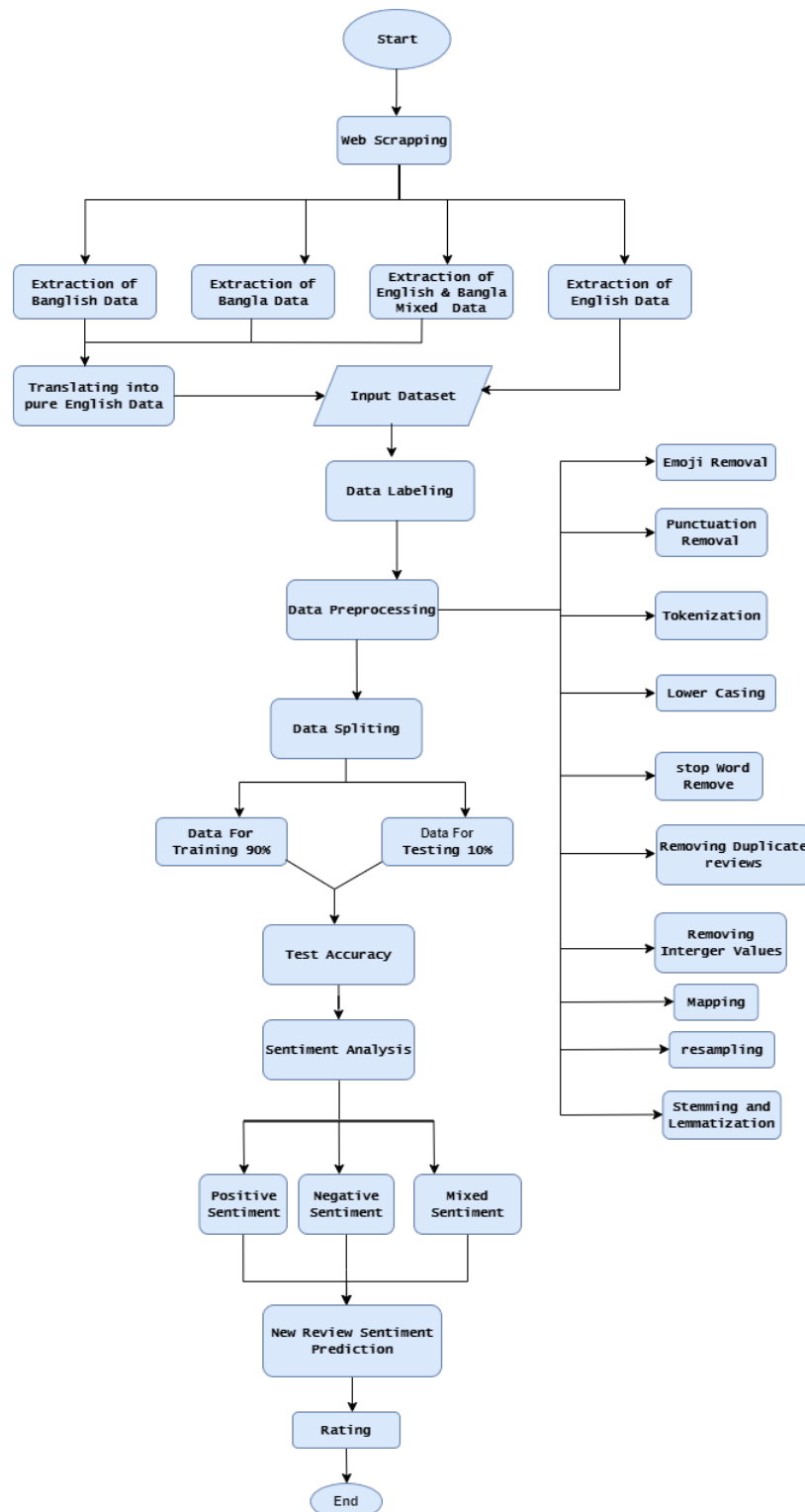


Figure 4.1: Work Flow diagram

4.1.4 Tokenization

Tokenization is the process of breaking down a series of phrases into smaller chunks, such as words. It's basically the process of splitting up words and phrases from paragraphs. Each word, each component is considered as an individual token in this concept. For example, we have a sentence in our dataset that goes like "This hospital is best" After tokenization it should be 'This', 'hospital', 'is', 'best'.

4.1.5 Removing Duplicate Reviews

Since we extracted reviews from the web, there was duplicate data. We removed those for better results.

4.1.6 Lemmatization

By studying word morphology and vocabulary, we realized that lemmatization leads to normalization. By removing just inflectional endings, lemmatization aims to preserve the lemma word's original form. Although slower than stemming, it is a much more advanced and potent text analysis technique. It seeks to keep the words' structural links intact. Deep learning models have been employed with this technique. The deep learning models develop the ability to take the input data's useful information and transform it into a format that can be applied to the task at hand.

4.1.7 Stemming

Stemming is a technique used in natural language processing (NLP) that involves breaking down words into their "stem," or root, form. Stemming is the process of normalizing word variants and condensing them into a basic structure, which can enhance text analysis, search precision, and information retrieval. Affixes like prefixes and suffixes are taken off of words during stemming to reveal the root meaning.

4.1.8 Oversampling

In order to balance the dataset, oversampling entails artificially boosting the number of cases in the minority class or classes. This is often accomplished by reproducing or creating fresh synthetic samples using the minority class data that is already available. The intention is to give the classifier a more balanced representation of the classes so that it can draw knowledge from a larger variety of examples.

4.1.9 Text Vectorization

The process of transforming textual input into a numerical representation that machine learning algorithms can comprehend is known as text vectorization. Text vectorization is an essential stage in natural language processing (NLP) applications like text classification, sentiment analysis, and document clustering because the majority of machine learning methods require numerical input. It uses techniques such as BoW, TF-IDF, Word embedding etc.

4.1.10 Under sampling

In machine learning and data analysis, under sampling is a method for addressing the problem of unbalanced data sets. The quantity of samples or instances belonging to one class is significantly more or fewer than the other classes in an unbalanced data set, which might result in biased model performance. In order to obtain a better balance across classes, under sampling involves lowering the number of instances in the dominant class or classes. Typically, to do this, instances from the majority class are randomly removed until the size of the minority class is achieved. The intention is to give the classifier a more balanced representation of the classes so that it can draw knowledge from a wider variety of examples.

4.1.11 Lower Casing

For the experiment purpose, initially, we tried to train our proposed model with data that had both the upper and lower case. The result wasn't satisfactory as the algorithms we used categorized these two types of data separately.

For example, positives are of the same meaning but in vector space models they are considered as different terms. Since we can't let the dimensions increase, after tokenizing we converted all our data into lowercase.

4.2 Word Cloud

In multi-class sentiment analysis, a word cloud can be a helpful visual representation to understand the most prevalent and noticeable words connected to various sentiment classes. It starts by gathering or setting up the dataset, which should include examples of text that have been classified according to various sentiments. It's crucial to preprocess the text data in order to get rid of extraneous information and noise before making the word cloud. Usually, this entails actions like eliminating punctuation, lowercasing the text, removing stop words, and conducting stemming or lemmatization. A list of words were made that express each sort of sentiment. These lists can be created manually or automatically using methods like term frequency-inverse document frequency (TF-IDF) or other sentiment analysis-specific approaches. Then word frequency was determined. A word cloud generator, such as Word-Cloud in Python was used to create the visual representation. It uses different colors, font sizes, or forms to alter the word cloud's appearance according to the sentiment class. Thus word clouds were created by adding the words that correspond to each sentiment class's respective frequency. From this dataset the review was categorized into three categories which are positive, negative and mixed respectively, a cloud visual representation from the dataset is shown in the figure above which includes different color size and contains all three sorts of sentiment.

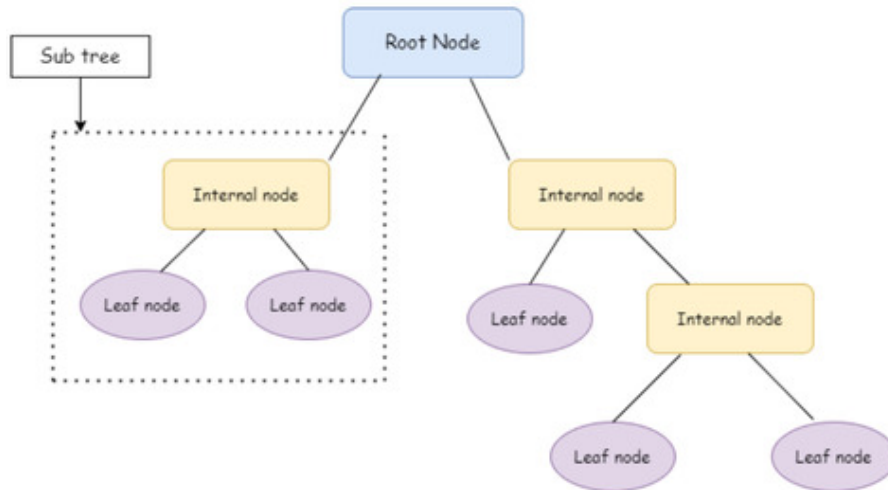


Figure 4.5: Decision Tree Classifier

4.4 SVM

The machine learning technique known as SVM, or Support Vector Machine, is frequently employed in sentiment analysis applications. The technique of figuring out the sentiment or emotion expressed in a piece of text, whether a sentence or a document, is called sentiment analysis. Our data set is comprised of multi-class statements which are divided into categories like positive, negative, and mixed. SVM can effectively handle high-dimensional feature spaces and can capture complex relationships between data points. An overview of the sentiment analysis process of SVM is shown below:

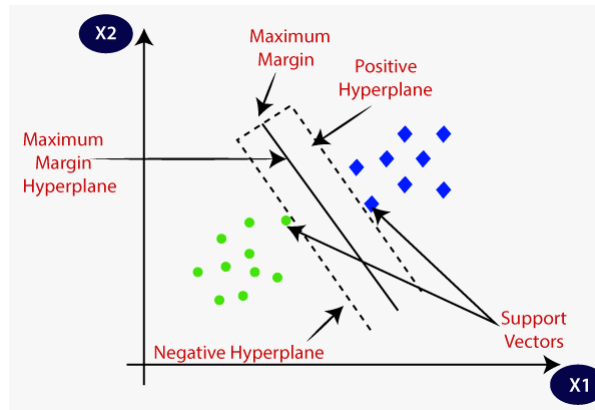


Figure 4.6: Performance Chart of Logistic Regression

Preparing Data

Gathering and preprocessing text data for sentiment analysis is the initial stage. Tokenization (the process of dividing a text into individual words or tokens), the elimination of stop words (frequent words like "the" or "and" that don't convey much sentiment), and stemming or lemmatization (the process of breaking down words into their simplest forms) are duties involved in this process. Tokenization and lemmatization were implemented in our dataset.

Extraction of features

The next step is to extract numerical features from the text data that can be utilized as inputs for the SVM algorithm. Using this method known as "bag-of-words," we represented each text as a vector of word frequencies. In order to capture more complex aspects, we also used other methods like TF-IDF (Term Frequency-Inverse Document Frequency) or word embedding.

Training and Prediction

After the features have been extracted our dataset is ready for training. On this dataset, the SVM algorithm is trained to identify patterns and connections between the text features and sentiment using the input features and related sentiment labels. Following training, the performance of the model is assessed using assessment measures like accuracy, precision, recall, or F1 score. The SVM model's ability to generalize to fresh, untested data is evaluated in this step. The SVM model can foretell the sentiment of fresh, unlabeled text input after being trained and assessed. The model outputs the anticipated sentiment label based on the new text's retrieved features as input.

4.5 CNN

Tasks requiring sentiment analysis can be successfully completed using Convolutional neural networks (CNN). In our dataset, we have plenty of texts to classify which is directly related to sentiment.

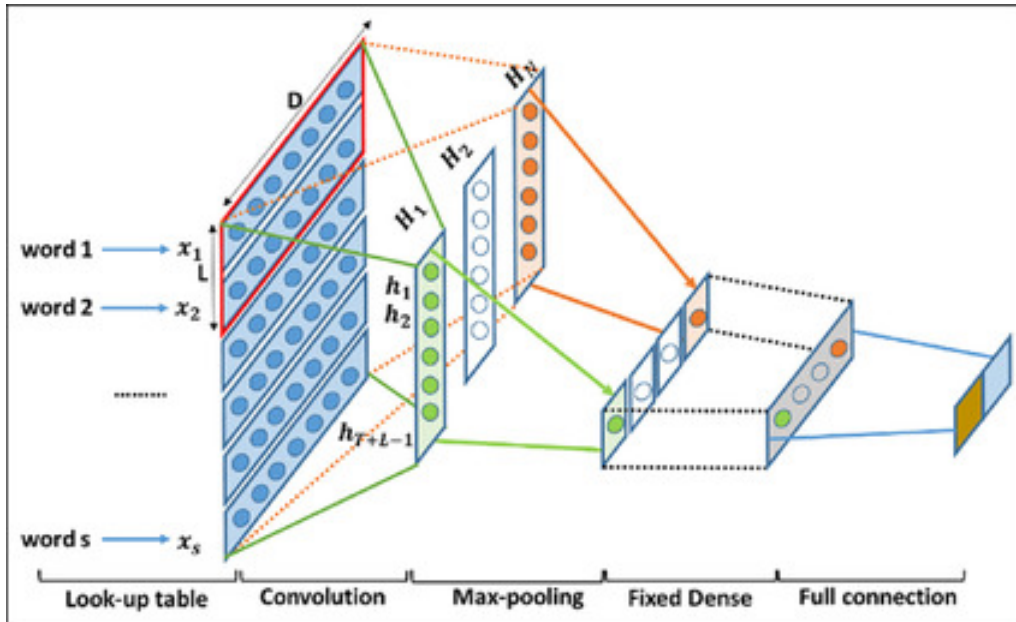


Figure 4.7: Standard-CNN-on-text-classification

Finding the sentiment or emotional polarity of a text, like a social media post to be more precisely the reviews of hospitals in our dataset, is the aim of sentiment analysis. The text's emotions can be categorized as either positive, negative, or mixed.

CNN can be used for sentiment analysis by the following parameters:

Text Preprocess

The text of our dataset was preprocessed before feeding it into the CNN. This typically involves tokenization (splitting text into words or characters), removing stop words, and applying techniques such as stemming or lemmatization to normalize the text'

Word Embedding

Word embeddings are voluminous vector representations of words. The word embeddings Word2Vec, GloVe, and fastText are frequently utilized. These embeddings depict the text in a more relevant way by capturing the semantic links between words.

Convolutional layers

Word embeddings are regarded as input channels in convolutional layers, much like picture channels are in conventional CNNs. To identify various local properties or patterns in the text, convolutional layers with several filters of various sizes (n-grams) are used.

Pooling Layers

The dimensionality of the feature maps produced by the convolutional layers is reduced using max pooling or average pooling. By lowering the sensitivity to the precise position of the features, pooling aids in capturing the most crucial features.

Output

The merged features are flattened before being transferred through one or more layers that are fully connected. These layers blend the characteristics retrieved by the convolutional layers and learn higher-level representations. The output layer, which typically comprises of one or more units with softmax activation to provide the sentiment probabilities for each class (positive, negative, and neutral), is connected to the final fully connected layer.

Training and Evaluation

A collection of texts that have been labeled with the corresponding sentiment labels is used to train the model. Typically, categorical cross-entropy is utilized as the loss function, and optimization methods such as stochastic gradient descent (SGD) or Adam are frequently used. After the model has been trained, its performance can be assessed on a different test set. The trained model may be used to predict fresh texts, and it will output the projected sentiment probabilities for each text as it is predicted.

4.6 Logistic regression

Logistic regression is basically a type of linear model that denotes that by default it assumes that the relationship between the input and output is linear. Even though it is a rather straightforward algorithm, it can still be useful for sentiment analysis jobs, especially when used in conjunction with the right feature engineering methods. It's significant to highlight that logistic regression can be applied to problems involving many classes of sentiment analysis in which there are more than two sentiment labels (such as positive, negative, or neutral). Techniques like one-vs-rest or softmax regression can be applied in these circumstances to manage numerous classes. In our thesis we had multiple classes of review in our dataset and thus one vs rest softmax regression was implemented.

Just like all other algorithms this algorithm initially preprocesses our data for the analysis of sentiment. To process the data tasks like tokenization lemmatization were performed. Secondly, the data was transformed into numerical factors. Techniques like word embedding was implemented to represent the data. We already had a labeled dataset with sentiment labels . The labels are typically positive, negative and mixed. On this dataset, the logistic regression algorithm is trained to understand the relationship between the input features and the corresponding sentiment labels. After training, metrics like accuracy, precision, recall, or F1 score are used to assess the logistic regression model's performance. This process aids in determining how effectively the model generalizes to fresh, untested data.

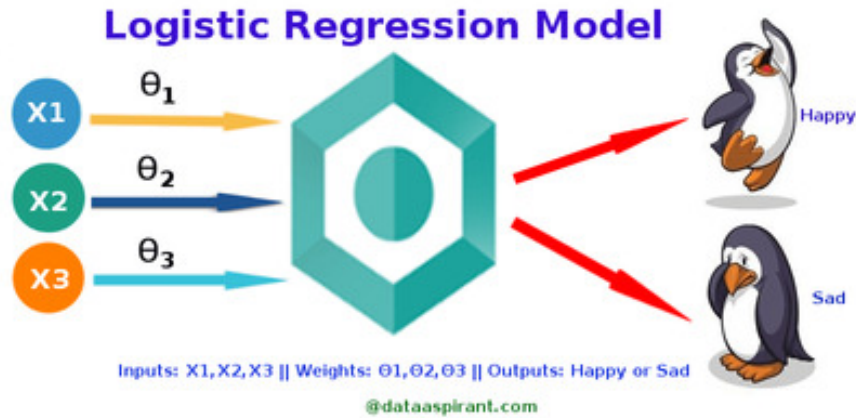


Figure 4.8: Logistic Regression

4.7 BERT(Bidirectional Encoder Representations from Transformers)

In the context of Transformers, Bert is also referred to as the Bidirectional Encoder Representations. The primary focus of Bert's work is the implementation of bidirectional training to model languages. This is a departure from the conventional deep learning models, which analyze text sequences from left to right or right to left. It operates in the form of an encoder-decoder, with the encoder portion reading text input and the decoder part making predictions about the text data. We have made advantage of BERT's already-pretrained models in order to expeditiously categorize the data that makes up our dataset. By doing this, we hope to be able to categorize the evaluations using the Multiclass text as Positive, Negative, and Mixed. Because we need the labels to be in a numeric format, we map them correspondingly to the values 0,1 and 2. In addition, we make use of a function applied to a column that has filtering criteria so that we may examine some random samples. It shows the outcome together with the appropriate conditions, which enables us to comprehend our strategy in the appropriate method. We used the Train Test split as 75:25. Here, the pretrained models were implemented using TensorFlow Hub. For the model, we have chosen universal-sentence-encoder-cmlm/multilingual-base. With this encoder, we are able to accommodate more than one hundred different languages. We hope that by using this approach, we will be able to transform textual input into vectors that reflect higher dimensions and get sentence level semantics. The encoder and preprocessor layers are imported from TensorFlow Hub on our end. Following that, we construct a function in order to obtain embeddings from the textual data that is included inside our dataset. In order to validate the model, we compared the cosine similarity across words that were conceptually similar and observed the outcome.

For example: Sentence 1: This hospital is very good, Sentence 2: This medical institute is best; should show similar high cosine similarity. On the other hand, Sentence 3: Our journey to the hospital was very lengthy should show lower cosine similarity than the actual sentence. Later on, we determined the precision, recall and F1 score to get a better understanding of the model. The layers are firstly encoder layer with followed by dropout layer. The next layer used is dense layer with SoftMax as activation function. To train the model, we used 20 epochs and validation loss

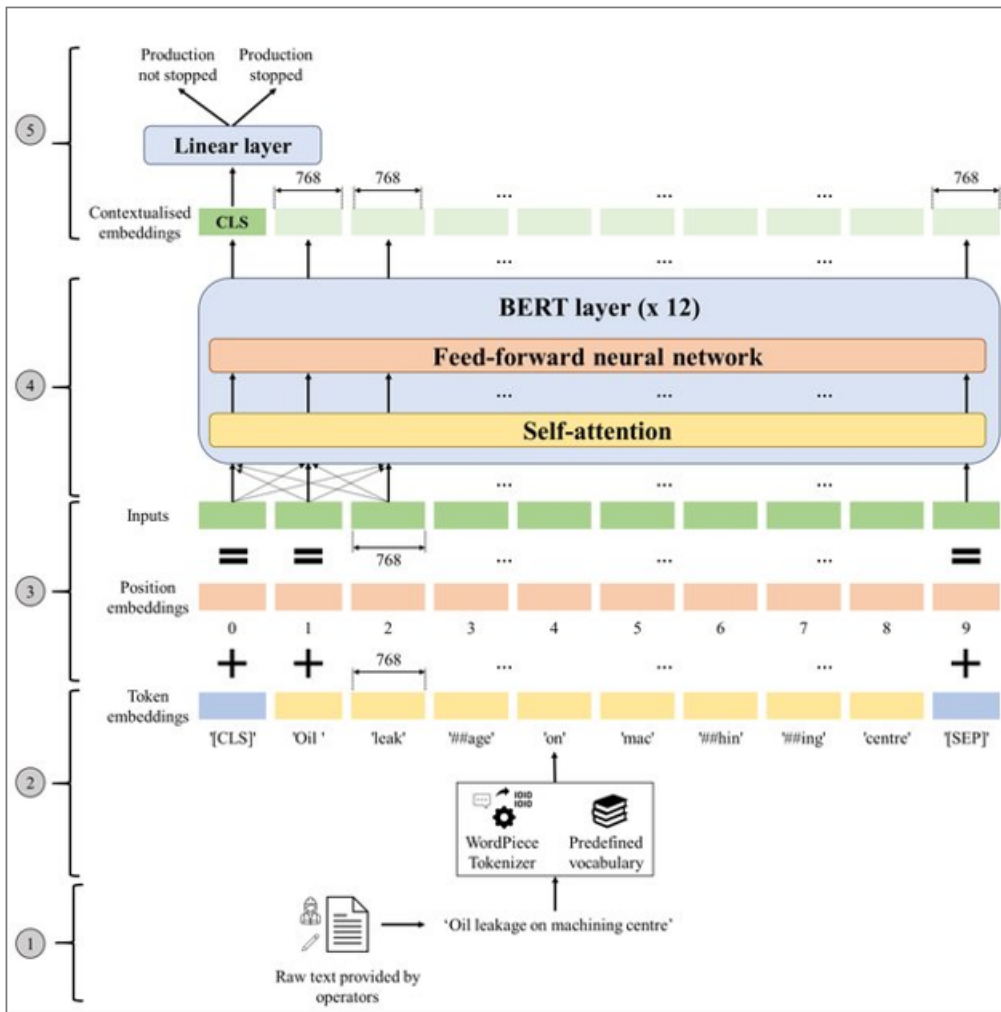


Figure 4.9: Bert diagram

was monitored using EarlyStopping. The metrics were used to generate training and validation curves. The losses measured were very minimal which determines the model is working optimally. Lastly, we use the model to predict sentences and get desired outputs.

4.8 Random Forest

Popular machine learning method Random Forest can be utilized successfully for sentiment analysis jobs. The goal of sentiment analysis is to identify the sentiment or emotional tone—whether positive, negative, or neutral—expressed in a particular text. In our dataset, the categories of sentiment were positive, negative and mixed respectively.

Just like all the other algorithms random forest initially does the preprocessing of the data. In preprocessing any irrelevant text is cleaned which includes special characters, emojis, numbers etc. After processing our data it was converted into numerical representation which can be used as input. Common techniques used are BOW, TF-IDF as well as word embedding like Word2Vec. Then the dataset was split into two parts; the train set and the test set. Utilizing the test set, we evaluated the trained Random Forest model. To evaluate its performance, compute measures like accuracy, precision, recall, or F1-score. To enhance the model's performance, we changed hyperparameters like the number of trees in the forest or the maximum depth of the trees. You can use the Random Forest model to foretell the sentiment of fresh, unexplored text data once it has been trained and assessed. We used the same feature extraction method as during training to transform the fresh text input, then feed it to the sentiment prediction model.

It's important to note that the caliber and representativeness of the training data are crucial for the success of sentiment analysis using Random Forest or any other machine learning method. To get the best results for particular sentiment analysis work, you might also need to tweak the algorithm and test out various preprocessing strategies, feature representations, and hyperparameter values.

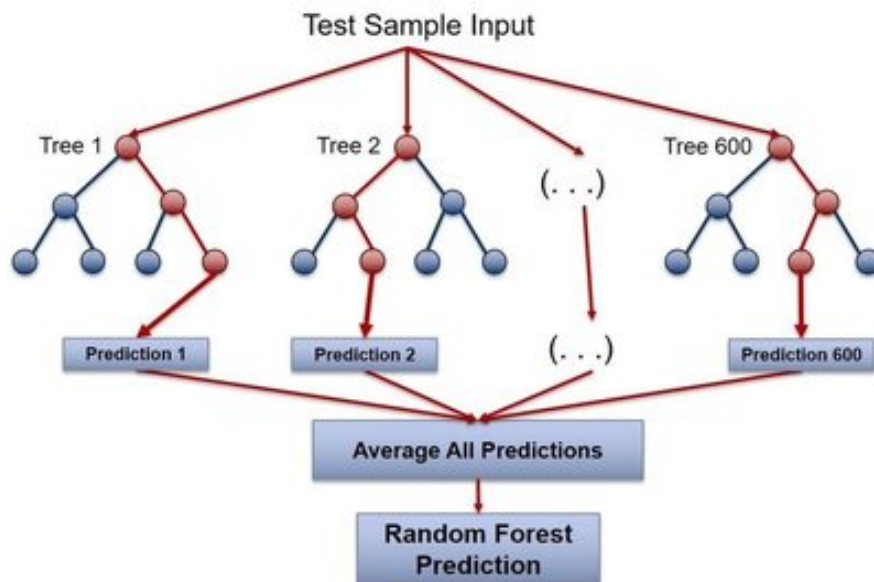


Figure 4.10: Random Forest

Chapter 5

Result Analysis and Evaluation

5.1 Overview

This chapter focuses on assessing the trained model's performance. To evaluate the model's efficiency, we created key metrics such as accuracy, F1 score, precision, and recall. A confusion matrix was used to provide a clear depiction of the classification results. Furthermore, we conducted a thorough side-by-side comparison of the model's six various variations. Depending on the algorithm we have two types of approach. (1) machine learning and (2) deep learning. For machine learning 4 algorithms were used which are Decision Tree Classifier, Random Forest, Logistic Regression and SVM respectively. For deep learning 2 algorithms were used which were BERT and CNN respectively. We illustrated the accuracy of every algorithm by developing a confusion matrix for each. Moreover, a performance matrix was also developed which demonstrates F1 Score, Precision and Recall.

5.2 Accuracy

The accuracy of the classification problem indicates the percentage of correct predictions. It is calculated by dividing the total amount of forecasted data by the total amount of estimated data that was accurate. Accuracy contrasts the ratio of accurate forecasts to all other predictions and is used to assess a model's performance. It is typically used as a concluding statistic to assess the general efficiency of a classification model.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (5.1)$$

5.3 Precision

Precision is the proportion of correctly predicted predictions (i.e., the number of right positive predictions) among all positive predictions made by the model (true positives plus false positives). It is a measure of how well the model can identify instances of positivity and lower the number of false positives. With low false positives and high accuracy, a model is less likely to incorrectly categorize a counter example as positive.

$$Precision = \frac{TP}{TP + FP} \quad (5.2)$$

5.4 Recall

Recall, also known as sensitivity, is the proportion of correctly predicted events (i.e., the number of positive predictions that materialized) among all real-world occurrences in the data (true positives plus false negatives). It assesses how well the model can distinguish each positive case with accuracy and minimize the number of false negatives. Since there are fewer false negatives, high recall makes it more likely that the model will properly identify all positive cases.

$$Recall = \frac{TP}{TP + FN} \quad (5.3)$$

5.5 F1 Score

The F1-score provides a complete view of Precision and Recall because it is the mean of these two numbers. It is at its finest if Precision and Recall are equal. A model's accuracy can be measured using this useful metric, but if the distribution of the classes is uneven (one class having much more examples than the other), it may not be accurate. A model that regularly predicts the majority class in these situations may be exceedingly accurate, but it would not be a desirable model. In this case, other metrics like accuracy, recall, and F1-score are more illuminating.

$$F1score = \frac{2PrecisionRecall}{Precision + Recall} = \frac{2TP}{2TP + FP + FN} \quad (5.4)$$

5.6 Confusion Matrix

The prediction summary is shown on a confusion matrix. It displays the number of accurate and wrong predictions made for each class. It is essential to plot a confusion matrix for a biased dataset matrix because it clarifies the classes that the model confuses for another class.

		Predicted		
		Positive	Negative	Mixed
Actual	Positive	0.55	0.16	0.04
	Negative	0.13	0.75	0.02
	Mixed	0.03	0.01	0.91

Table 5.1: Confusion Matrix

5.7 Machine Learning Approach

In this case, we used four algorithms in our machine learning approach: SVM, Random forest, Logistic regression, and Decision tree. Support Vector Machine has an accuracy of 85.32%, Random Forest has an accuracy of 82.96%, Logistic Regression has an accuracy of 82.81%, and Decision Tree has an accuracy of 74.67%.

5.7.1 Support Vector Machine

The Support Vector Machine has an accuracy of 0.8532, or 85.32%. The accuracy of this model can be evaluated using precision, recall, f1 score etc. The performance matrix in the figure below illustrates that it has got precision of 86%, recall of 85%, and f1 score of 77%. To further enhance performance, hyperparameter modification may be required for the regularization parameter (C) and kernel selection (linear, polynomial, radial basis function)The Support Vector Classifier's confusion matrix and Performance Chart are shown below:

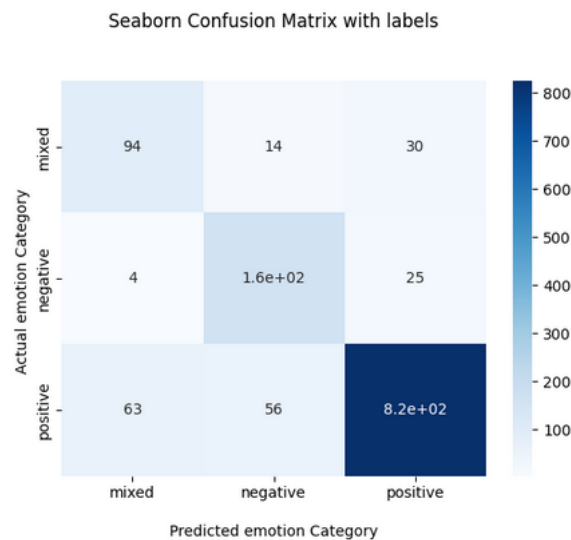


Figure 5.1: Confusion Matrix of Support Vector Machine

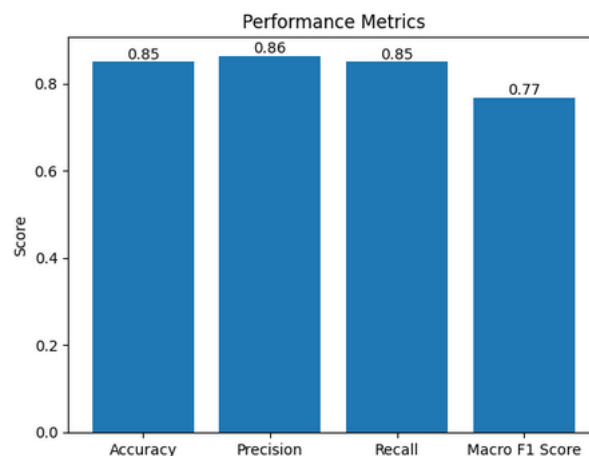


Figure 5.2: Performance Chart of Support Vector Machine

5.7.2 Random Forest Classifier

The Random Forest Classifier has an accuracy of 0.8296, or 82.96%. We can assess the precision, recall, F1-score, or area under the receiver operating characteristic curve (AUC-ROC) of the Random Forest classifier in multi-label text classification. For further improvement of performance, hyperparameter adjustment may be required, such as adjusting the number of decision trees and the maximum depth of each tree. From its performance matrix we can notice that it has precision of 86%, recall 83% and f1 score of 74%. The Random Forest Classifier's confusion matrix and Performance Chart are shown below:

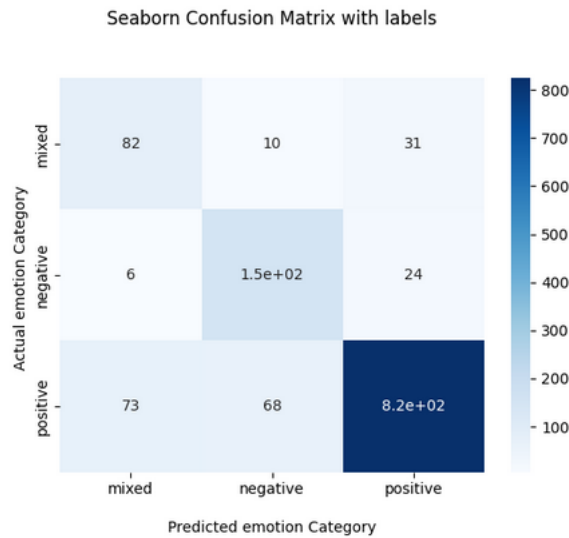


Figure 5.3: Confusion Matrix of Random Forest



Figure 5.4: Performance Chart of Random Forest

5.7.3 Logistic Regression

The Logistic Regression Classifier has an accuracy of 0.8281, or 82.81%. The precision, recall, F1-score, or area under the receiver operating characteristic curve (AUC-ROC) are common assessment metrics used to assess the quality of the logistic regression model in multi-label text categorization. These metrics offer a more thorough evaluation of the model's performance because they take into account both the true positive and false positive rates for each label. The performance chart below illustrates a precision of 85%, recall 82% and f1 score 70%. The Random Forest Classifier's confusion matrix and Performance Chart are shown below:

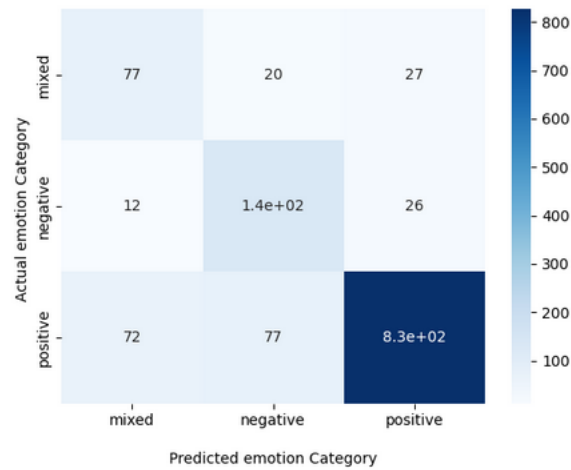


Figure 5.5: Confusion Matrix of Logistic Regression

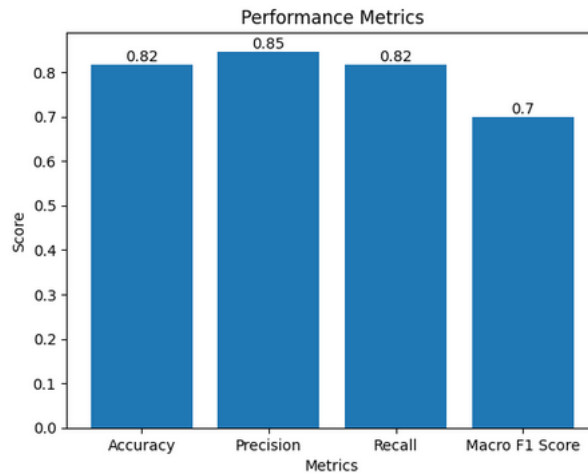


Figure 5.6: Performance Chart of Logistic Regression

5.7.4 Decision Tree Classifier

The decision Tree Classifier has an accuracy of 0.7467, or 74.67%. One popular assessment metric is the "accuracy score," which is used to evaluate the decision tree classifier's accuracy in multi-label text categorization. Out of all the labels in the test set, it calculates the proportion of labels that were accurately predicted. In performance, it has achieved a precision of 81%, recall 75%, f1 score 77%. The Decision Tree Classifier's. Figure 5.7 shows the confusion matrix of the Decision Tree Classifier.

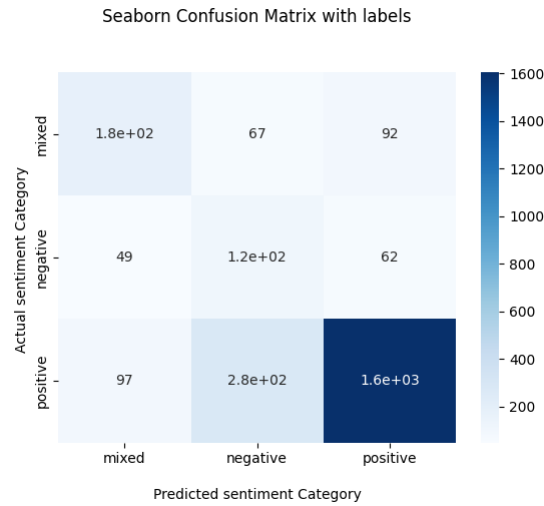


Figure 5.7: Confusion Matrix of Decision Tree Classifier

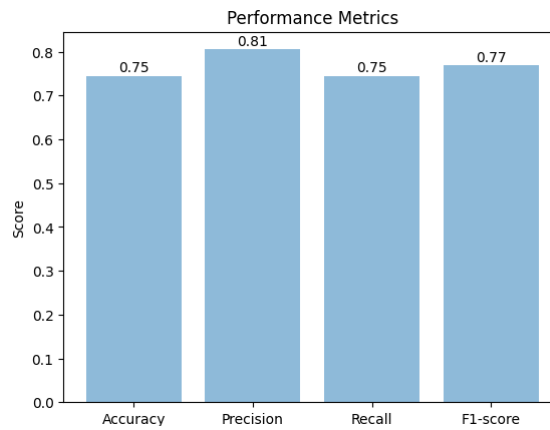


Figure 5.8: Performance Chart of Decision Tree Classifier

5.8 Deep Learning Approach

In this scenario, we employed two deep learning algorithms: BERT and CNN. BERT has an accuracy rate of 84.56%, whereas CNN has a rate of 79.20 %.

5.8.1 Bidirectional Encoder Representations from Transformers (BERT)

BERT's accuracy is 0.8456, or 84.56%. The suggested technique correctly predicts 2463 positive labels, 493 negative labels, and 244 mixed labels. To test the model, we examined the cosine similarity between similar sentences and observed the result. For example: Sentence 1: This hospital is very good, Sentence 2: This medical institute is best; should show similar high cosine similarity. On the other hand, Sentence 3: Our journey to the hospital was very lengthy should show lower cosine similarity than the actual sentence. Later on, we determined the precision, recall and F1 score to get a better understanding of the model. The layers are firstly an encoder layer followed by a dropout layer. The next layer used is a dense layer with SoftMax as activation function. To train the model, we used 20 epochs and validation loss was monitored using EarlyStopping. From performance matrix figure it can be realized that f1, recall and precision of this model fluctuates. By calculating the average this model has got precision of 76%, recall 71.5% and f1 score of 73%. It' The Bert's confusion matrix and performance report are provided below:

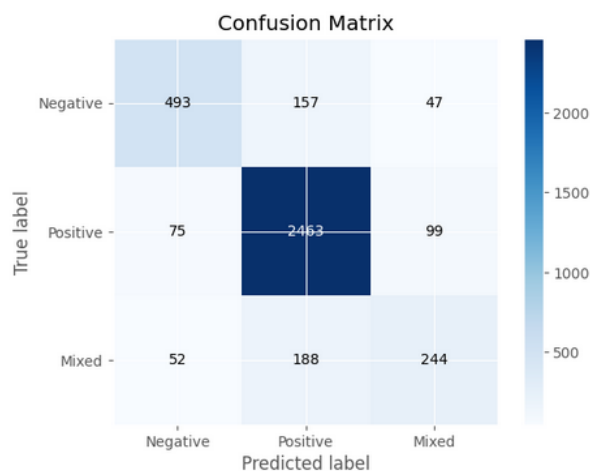


Figure 5.9: Confusion Matrix of BERT

if we look at the in figure 5.12 We can determine that with the increase of epochs the training and validation accuracy gets to an optimal point with a lesser difference. For loss, in figure 5.13 we can notice the vice versa which is decreasing. As the accuracy does not increase after a particular number of epochs, early stopping is implemented for avoiding overfitting. Here we have used 14 epochs because the dataset was too large, if we run more epochs then it will be time more time consuming, additionally the gpu won't be able to handle the load of this epochs.

```

297279 [.....] - 142 510m/step - loss: 0.5553 - accuracy: 0.7554 - balanced_recall: 0.4756 - balanced_precision: 0.5461 - balanced_f1_score: 0.5239 - val_loss: 0.4357 - val_accuracy: 0.8217 - val_balanced_recall: 0.5580 - val_balanced_precision: 0.7575 - val_balanced_f1_score: 0.6517
297279 [.....] - 143 507m/step - loss: 0.4482 - accuracy: 0.8224 - balanced_recall: 0.4869 - balanced_precision: 0.7389 - balanced_f1_score: 0.6036 - val_loss: 0.4387 - val_accuracy: 0.8337 - val_balanced_recall: 0.6080 - val_balanced_precision: 0.7597 - val_balanced_f1_score: 0.6873
297279 [.....] - 144 499m/step - loss: 0.4455 - accuracy: 0.8251 - balanced_recall: 0.5413 - balanced_precision: 0.7591 - balanced_f1_score: 0.6579 - val_loss: 0.4445 - val_accuracy: 0.8428 - val_balanced_recall: 0.6564 - val_balanced_precision: 0.7665 - val_balanced_f1_score: 0.7211
297279 [.....] - 145 492m/step - loss: 0.4275 - accuracy: 0.8237 - balanced_recall: 0.4959 - balanced_precision: 0.7650 - balanced_f1_score: 0.7014 - val_loss: 0.4200 - val_accuracy: 0.8506 - val_balanced_recall: 0.6559 - val_balanced_precision: 0.7645 - val_balanced_f1_score: 0.6905
297279 [.....] - 146 484m/step - loss: 0.4239 - accuracy: 0.8262 - balanced_recall: 0.4986 - balanced_precision: 0.7691 - balanced_f1_score: 0.7007 - val_loss: 0.4248 - val_accuracy: 0.8423 - val_balanced_recall: 0.6454 - val_balanced_precision: 0.7618 - val_balanced_f1_score: 0.7158
297279 [.....] - 147 476m/step - loss: 0.4238 - accuracy: 0.8258 - balanced_recall: 0.4975 - balanced_precision: 0.7670 - balanced_f1_score: 0.7077 - val_loss: 0.4228 - val_accuracy: 0.8421 - val_balanced_recall: 0.6500 - val_balanced_precision: 0.7713 - val_balanced_f1_score: 0.7211
297279 [.....] - 148 468m/step - loss: 0.4281 - accuracy: 0.8380 - balanced_recall: 0.4687 - balanced_precision: 0.7681 - balanced_f1_score: 0.7138 - val_loss: 0.4225 - val_accuracy: 0.8418 - val_balanced_recall: 0.6668 - val_balanced_precision: 0.7642 - val_balanced_f1_score: 0.7068
297279 [.....] - 149 460m/step - loss: 0.4219 - accuracy: 0.8481 - balanced_recall: 0.4754 - balanced_precision: 0.7756 - balanced_f1_score: 0.7179 - val_loss: 0.4218 - val_accuracy: 0.8418 - val_balanced_recall: 0.6886 - val_balanced_precision: 0.7756 - val_balanced_f1_score: 0.7268
297279 [.....] - 150 452m/step - loss: 0.4129 - accuracy: 0.8509 - balanced_recall: 0.4750 - balanced_precision: 0.7898 - balanced_f1_score: 0.7155 - val_loss: 0.4158 - val_accuracy: 0.8408 - val_balanced_recall: 0.6792 - val_balanced_precision: 0.7754 - val_balanced_f1_score: 0.7302
297279 [.....] - 151 444m/step - loss: 0.4137 - accuracy: 0.8482 - balanced_recall: 0.4768 - balanced_precision: 0.7788 - balanced_f1_score: 0.7187 - val_loss: 0.4178 - val_accuracy: 0.8408 - val_balanced_recall: 0.6870 - val_balanced_precision: 0.7716 - val_balanced_f1_score: 0.7268
297279 [.....] - 152 436m/step - loss: 0.4123 - accuracy: 0.8465 - balanced_recall: 0.4987 - balanced_precision: 0.7741 - balanced_f1_score: 0.7111 - val_loss: 0.4140 - val_accuracy: 0.8415 - val_balanced_recall: 0.6792 - val_balanced_precision: 0.7705 - val_balanced_f1_score: 0.7211
297279 [.....] - 153 428m/step - loss: 0.4120 - accuracy: 0.8509 - balanced_recall: 0.4775 - balanced_precision: 0.7807 - balanced_f1_score: 0.7154 - val_loss: 0.4170 - val_accuracy: 0.8415 - val_balanced_recall: 0.6925 - val_balanced_precision: 0.7705 - val_balanced_f1_score: 0.7211
297279 [.....] - 154 420m/step - loss: 0.4138 - accuracy: 0.8384 - balanced_recall: 0.4990 - balanced_precision: 0.7750 - balanced_f1_score: 0.7133 - val_loss: 0.4140 - val_accuracy: 0.8413 - val_balanced_recall: 0.6864 - val_balanced_precision: 0.7709 - val_balanced_f1_score: 0.7267
297279 [.....] - 155 412m/step - loss: 0.4123 - accuracy: 0.8452 - balanced_recall: 0.4993 - balanced_precision: 0.7750 - balanced_f1_score: 0.7200 - val_loss: 0.4120 - val_accuracy: 0.8409 - val_balanced_recall: 0.6900 - val_balanced_precision: 0.7700 - val_balanced_f1_score: 0.7267
297279 [.....] - 156 404m/step - loss: 0.4125 - accuracy: 0.8418 - balanced_recall: 0.4921 - balanced_precision: 0.7721 - balanced_f1_score: 0.7200 - val_loss: 0.4168 - val_accuracy: 0.8394 - val_balanced_recall: 0.6780 - val_balanced_precision: 0.7679 - val_balanced_f1_score: 0.7127
297279 [.....] - 157 396m/step - loss: 0.4101 - accuracy: 0.8407 - balanced_recall: 0.4904 - balanced_precision: 0.7750 - balanced_f1_score: 0.7200 - val_loss: 0.4122 - val_accuracy: 0.8409 - val_balanced_recall: 0.7120 - val_balanced_precision: 0.7677 - val_balanced_f1_score: 0.7277

```

Figure 5.10: Performance Chart of BERT

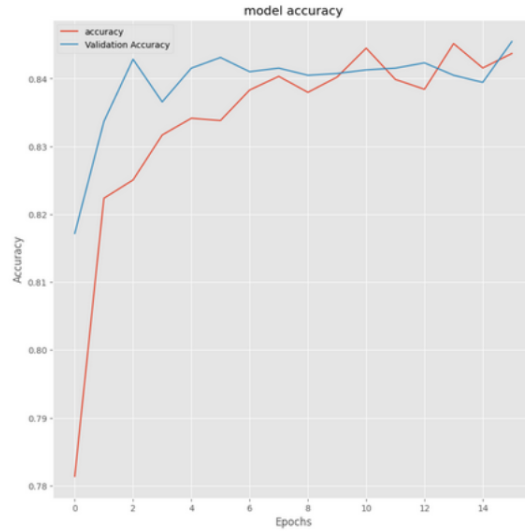


Figure 5.11: Validation Accuracy of BERT

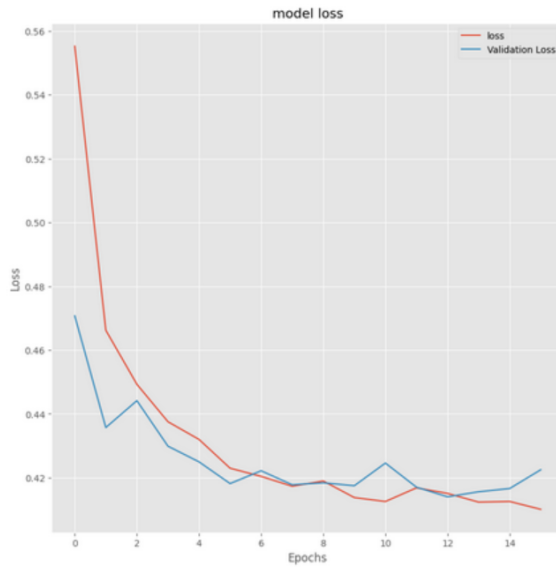


Figure 5.12: Validation Loss of BERT

5.8.2 Convolutional Neural Network(CNN)

The accuracy of this model is 79.2%. This model can successfully predict 1545 positive, 286 negative and 191 mixed labeled sentiments. We have trained around

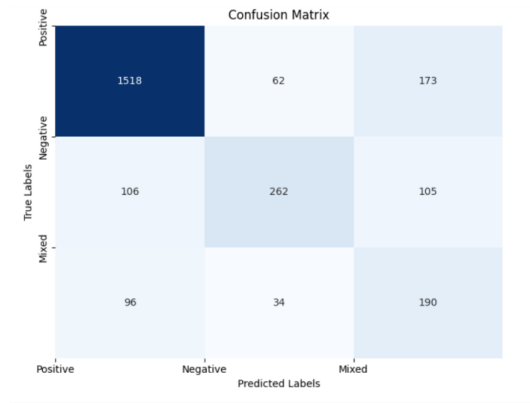


Figure 5.13: Confusion Matrix of CNN

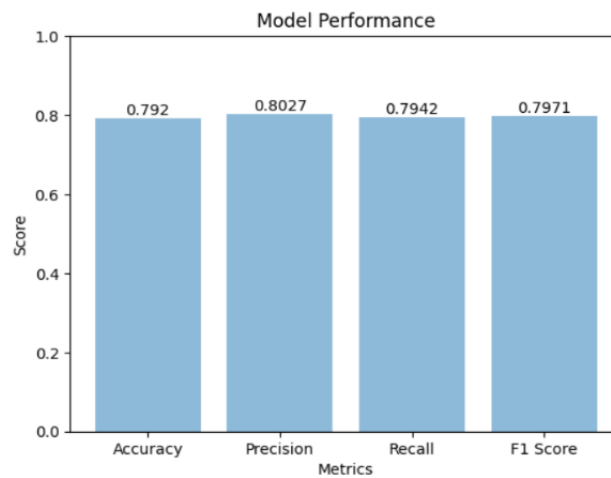


Figure 5.14: Performance Matrix of CNN

10-15 epochs in this model. Validation loss and accuracy fluctuates with in the increase numbers of epochs. Before training epochs the text of the dataset goes through two layers, one is the convolutional layer, another is the pooling layer, we can convolutional/ pooling layers as per our convenience, the more convolutional layer the more dimensionality is obtained. In contrast, the pooling layer reduces the dimensionality created by the convolutional layer. It has got precision of 80.27%, recall 79.42%, f1 score of 79.71% . Here in fig 5.15 we have used 11 epochs depending on the size of the data, it's complexity and our device capacity. The confusion and the precision matrix is given below:



Figure 5.15: Validation Loss of CNN

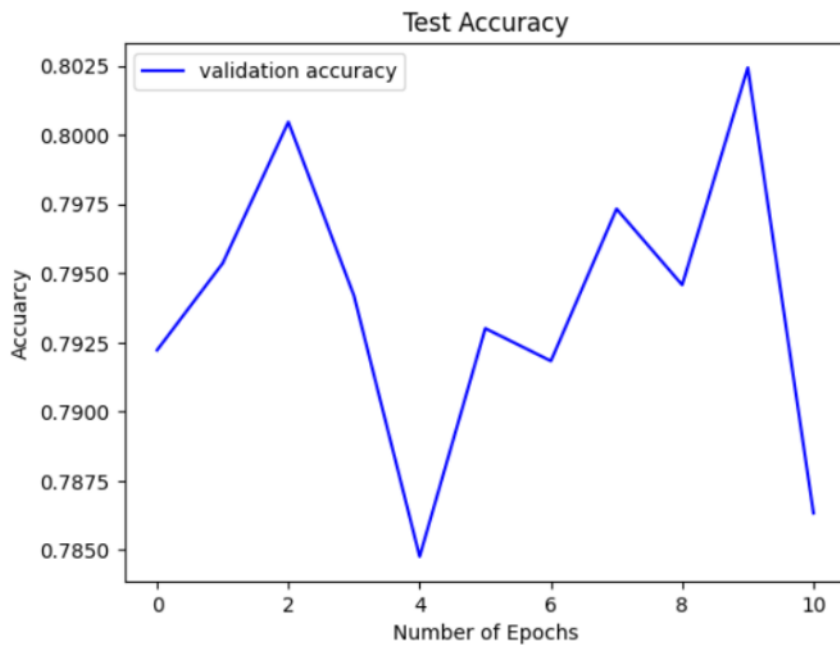


Figure 5.16: Validation Accuracy of CNN

5.9 Ranking

The current rating system in Google Maps is a scale based rating system. In the traditional rating system of Google it uses the average value of stars to rate an institution, but in our advanced hospital rating system we have rated the hospital based on sentiment of the reviews. Our rating is based on the emotion of the text that the user provided in the Google Maps review section. We divided the sentiment

into three categories which are positive, negative and mixed respectively. After that, we have ranked the hospital based on the positive percentage of reviews. For this research, we have ranked those hospitals which have got more than 390 reviews and illustrated the top 10 hospitals in Dhaka. We have divided the ranking system into two categories:

- (1) General Ranking and
- (2) Class Based Ranking.

$$\text{rating} = \left(\frac{\text{positive reviews}}{\text{total reviews}} \right) \times 100 \quad (5.5)$$

5.9.1 General Ranking

The following chart illustrates our general ranking. We have ranked the following hospitals on the basis of positive review percentage. In this category the hospital that has the most positive review percentage secures the first position. Here some hospitals have higher number of reviews, some have comparatively lower. The formula we applied here is $((\text{positive} / \text{total}) * 100)$. We have IBN Sina Specialized Hospital standing in the first position which acquired 85.2% positive reviews. Here we split our dataset based on the type of reviews.

Rank	Hospital Name	Positive	Negative	Mixed	Positive (%)
1st	IBN Sina Specialized Hospital	367	40	24	85.15%
2nd	Ad-Din Women's Medical College Hospital	320	79	24	75.65%
3rd	Green Life Medical College Hospital	417	104	39	74.46%
4th	Asgar Ali Hospital	288	63	40	73.66%
5th	Square Hospital	1325	304	310	68.33%
6th	BRB Hospitals Limited	283	95	45	66.90%
7th	United Hospital Limited	1116	289	307	65.19%
8th	LABAID Specialized Hospital	913	270	242	64.07%
9th	Bangladesh Specialized Hospital	272	166	53	55.40%
10th	Evercare Hospital Dhaka	530	308	226	49.81%

Table 5.2: General Hospital Ranking

5.9.2 Class Based Ranking

The following picture shows our class based ranking. In the case of class based ranking the hospitals were divided into a specific category of class considering the number of reviews. Each class has a specific numerical range. The range of the higher class is 1000-2000 and the range of the lower class is 400-999. The higher class range was given the highest priority, apart from the number of reviews the positive percentage was considered as a significant parameter. So the hospital with a higher no of reviews as well as higher percentage of positive reviews will remain at a peak level of ranking. In the following table, we see that Square Hospital stands at the first position. since it falls in the higher class, it has total reviews of more than 1900 and among the higher class hospitals, it has the highest percentage of positive reviews. By adding the positive, negative and mixed reviews the total number of reviews is determined. So here two parameters are taken into consideration; one is their class based on number of reviews and the second one is the positive percentage. If a hospital has a higher positive percentage but a lower class range then it will not surpass those hospitals enrolled in the higher class range.

Here we split our dataset based on the number of reviews.

Class	Rank	Hospital Name	Positive	Negative	Mixed	Positive (%)
2000-1000	1st	Square Hospital	1325	304	310	68.33%
	2nd	United Hospital Limited	1116	289	307	65.19%
	3rd	LABAID Specialized Hospital	913	270	242	64.07%
	4th	Evercare Hospital Dhaka	530	308	226	49.81%
999-390	5th	IBN Sina Specialized Hospital	367	40	24	85.15%
	6th	Ad-Din Women's Medical College Hospital	320	79	24	75.65%
	7th	Green Life Medical College Hospital	417	104	39	74.46%
	8th	Asgar Ali Hospital	288	63	40	73.66%
	9th	BRB HOSPITALS LIMITED	283	95	45	66.90%
	10th	Bangladesh Specialized Hospital	272	166	53	55.40%

Table 5.3: Class Based Hospital Ranking

$$\text{total reviews} = \text{positive} + \text{negative} + \text{mixed} \quad (5.6)$$

$$\text{total reviews} \geq 1000 = \text{higherclass}(2000 - 1000) \quad (5.7)$$

$$\text{total reviews} < 1000 = \text{lowerclass}(999 - 390) \quad (5.8)$$

5.10 Comparison of the Accuracy of Algorithms

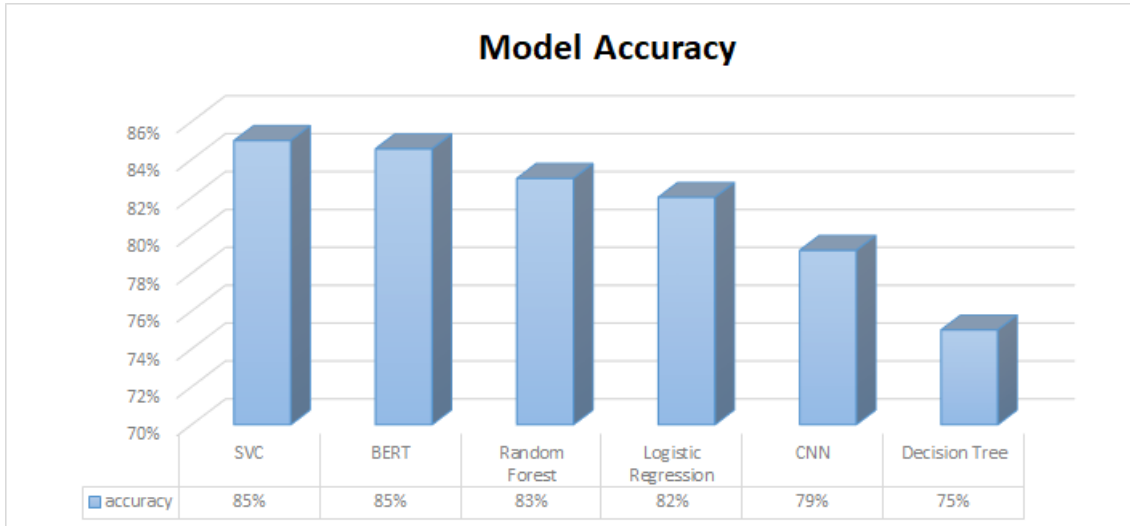


Figure 5.17: Model Accuracy Comparison

So far we have implemented 6 algorithms to deploy our proposed model, they are; Decision Tree, BERT, SVM, Random Forest, Logistic Regression and CNN algorithms. The above mentioned algorithms contain both deep learning and machine learning approach. For machine learning svm, logistic regression, decision tree and random forest were used. For deep learning BERT and CNN were used. These algorithms calculate the accuracy score and precision score respectively. Here we can see from the bar graph that SVM model showed an accuracy of 85.32%, BERT model showed an accuracy of 84.56%, Random Forest model showed an accuracy of

82.96%, Logistic Regression model showed 82.81%, CNN model showed 79.2%, Decision tree model showed 75% accuracy. Among these six, SVM showed the highest accuracy, meaning this algorithm can accurately identify positive, negative or mixed labeled reviews, and it can predict the emotions from text opinions correctly.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
SVC	85.32%	86.23%	85.41%	77.80%
BERT	84.56%	76.16%	71.39%	73.40%
Random Forest	82.96%	86.77%	83.30%	74.80%
Logistic Regression	82.81%	85.00%	82.00%	70.00%
CNN	79.20%	80.10%	79.40%	70.29%
Decision Tree	74.67%	81.20%	75.23%	77.32%

Table 5.4: Model Evaluation Table

5.11 Rating system generated by SVM

A trained Support Vector Machine (SVM) model for sentiment analysis of hospital reviews is used in the rating system that is produced. Positive, negative, and mixed emotions are all taken into account. Future unlabeled reviews of any particular hospital may be anticipated using this algorithm, and scores depending on the fraction of favorable reviews can be calculated. A dataset of labeled hospital reviews with sentiments classified as positive, negative or mixed is used to train the SVM model. The model gains the ability to categorize emotions across various labels by employing the One-vs-Rest method.

reviews	
0	very good
1	best hospital
2	bad hospital
3	good hospital but dirty

Figure 5.18: Unlabelled Reviews

The SVM model is used to anticipate the sentiment stated in each review in order to rate unlabeled reviews. The proportion of positive sentiment can be determined by looking at the probability or decision scores that the model provides. This percentage is used to calculate the overall score for each hospital review. This proposed rating created has a number of advantages. It offers a scalable, automated method for processing a lot of reviews quickly. It provides a more sophisticated assessment of patient experiences by taking sentiment analysis into account rather than relying just on conventional star ratings. The training dataset's quality and representativeness must be guaranteed for subsequent improvement. For precise

predictions, it is essential to have a diverse, well-annotated dataset with a range of attitudes.

	reviews	predictions
0	very good	positive
1	best hospital	positive
2	bad hospital	negative
3	good hospital but dirty	mixed

Figure 5.19: Predictive hospital reviews

in the Figure 5.17 the unlabelled reviews of a hospital are viewed in the data frame. From this reviews column the proposed SVM model has predicted the sentiment which is illustrated in the 5.18 figure. Finally, the rating is calculated by the positive percentage of the total reviews which is shown in the given figure.

For rating a hospital positive percentage of the total reviews has been taken into account. For counting the Positive percentage we simply divide the positive reviews by the total amount of reviews and multiply by 100. The rating counts from 1 to 100. A higher value determines a better rating.

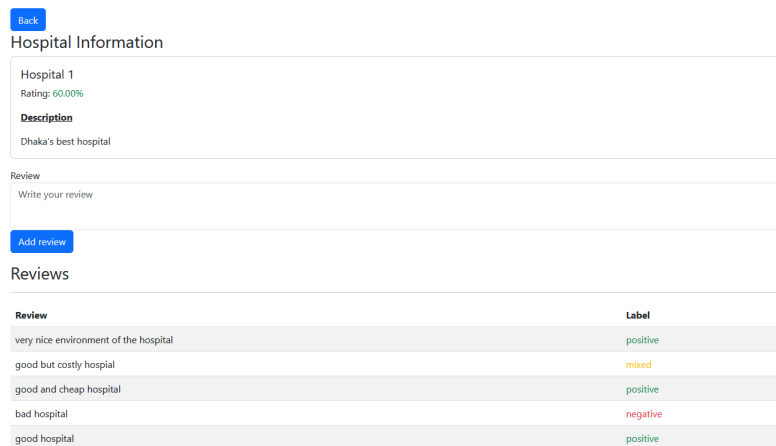
$$\text{Rating} = \left(\frac{\text{positive reviews}}{\text{total reviews}} \right) \times 100 \quad (5.9)$$

rating or Positive Percentage: 50.000%

Figure 5.20: Predictive hospital rating

5.12 Hospital Ranking web application

This web application can rank the hospitals based on analysis of the sentiments of the reviews. It has got two user interfaces, which are hospital ranking and hospital review. In the hospital review UI, a user can share their experience of a particular hospital. In that review input field, this review is then fed to the trained SVM model in the background. our model then predicts the sentiment between three classes. These classes can be either positive, negative, or mixed. the prediction result is shown in the sentiment column of the review page which is shown in Figure 5.19. After that, the calculation of the positive percentage is shown in the rating output field.



Review	Label
very nice environment of the hospital	positive
good but costly hospital	mixed
good and cheap hospital	positive
bad hospital	negative
good hospital	positive

Figure 5.21: Review page of web application

finally, on the 'ranking' page, these ratings of hospitals are compared and sorted with higher priority. In the Figure 5.20 page, the hospitals are sorted in ascending order of their rating. For this web application joblib library of Python is used to load the SVM model. For storing the hospital reviews sqlite us used as a DBMS. Django is used as a backend. and plain HTML is used for frontend.

General rating

#rank	Hospital name	Number of reviews	Rating
1	Hospital 1	5	60.00%
2	Square	4	50.00%
3	Hospital 2	6	33.33%

Figure 5.22: Ranking page of web application

Chapter 6

Conclusion and Future Works

6.1 Conclusion

Each sector of healthcare requires its own set of technical software or systems. Developing an innovative hospital rating system is a fantastic way if we think about improving a unique and effective healthcare strategy. The implementation of an advanced hospital rating system project aids in the storage of all kinds of documents, user communication, and legislative reform. This model can fulfil the demands of patients, employees, and hospital administrators while also easing interaction. So far with the implementation of 3 algorithms (BERT, CNN Decision Tree), we tried to train our model on our classified dataset that we created. Our purpose was to compare between the accuracy of algorithms whether it can predict positive or negative reviews. In the future we plan to rank different hospitals based on different ML algorithms.

6.2 Future Works

We wish to expand our research by incorporating hybrid models such as ensemble models, multi-modal models, transfer learning models, hybrid deep learning models, semantic-based models, etc. These models can employ a variety of learning approaches, such as deep learning or traditional machine learning, and can use many different types of data, such as text, images, or audio. In future the ranking system will be more enhanced; if people wants to search ranking by a specific field like cardiology, radiology etc they will avail the ranking based on that category. In future we can utilize the above mentioned model to achieve more accuracy.

Bibliography

- [1] R. M. B. Ricardo K. Hussmann and S. Zaman, “Dhaka: Health economics unit, ministry of health and family welfare, government of bangladesh;,” 2004.
- [2] D. M. A. Cokroft and N. Anderson, “?bangladesh health and population sector programme: Third service delivery survey 2003: Final report,” 2003.
- [3] IHE, “Cross border health care: A study of determinants for patients in kolkata from bangladesh,” 2002.
- [4] B. Charbuty and A. Abdulazeez, “Classification based on decision tree algorithm for machine learning,,” pp. 20–28, 2021.
- [5] S. S. M. A. -U.-Z. Ashik and S. Haque, “Data set for sentiment analysis on bengali news comments and its baseline evaluation, 2019 international conference on bangla speech and language processing (icbslp) ieee,” pp. 1–5, 2019.
- [6] K. L. J. Devlin M. W. Chang and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding, arxiv preprint arxiv:1810.04805,” 2018.
- [7] S. K. a. P. Shah F. Kendall, “Artificial intelligence and machine learning in clinical development: A translational perspective, npj digital medicine, jourvol 2, number 1,” pp. 1–5, 2019.
- [8] M. Chen and M. Decary, “Artificial intelligence in healthcare: An essential guide for health leaders, healthcare management forum sage publications sage ca: Los angeles, ca, volume 33,” pp. 10–18, 2020.
- [9] M. I. Ibrahim, A. Rahim, K. Musa, S.-L. Chua, and N. Yaacob, “Assessing patient-perceived hospital service quality and sentiment in malaysian public hospitals using machine learning and facebook reviews,” *International Journal of Environmental Research and Public Health*, vol. 18, no. 18, p. 9912, 2021.
- [10] M. E. M. a. H. J. Murff F. FitzHenry, “Automated identification of postoperative complications within an electronic medical record using natural language processing, jama, journal 306, number 8,” pp. 848–855, 2011.
- [11] S. N. V. Kalyan T. J. Reddy and A. M. Prakash, “Comparative study of machine learning algorithms for hospital rating system, hospital, jourvol 1, page 2,”
- [12] W. Sun, Z. Cai, Y. Li, F. Liu, S. Fang, and G. Wang, “Data processing and text mining technologies on electronic medical records: A review,” *Journal of Healthcare Engineering*,
- [13] J. Sensmeier, “Harnessing the power of artificial intelligence, nursing management, jourvol 48, number 11, pages 14–19, 2017,” pp. 14–19, 2017.

- [14] S.-L. C. A.I.A.Rahim M. I. Ibrahim and K. I. Musa, “Hospital facebook reviews analysis using a machine learning sentiment analyzer and quality classifier, in *healthcare mdpi*, volume 9,” p. 1679, 2021.
- [15] J. B. Hawkins, J. S. Brownstein, G. Tuli, *et al.*, “Hospital facebook reviews analysis using a machine learning sentiment analyzer and quality classifier,” *Healthcare MDPI*, vol. 9, p. 1679, 2021.
- [16] T. G. J.Lopes and M. F. Santos, “Predictive and prescriptive analytics in healthcare: A survey, *procedia computer science*, jourvol 170,” pp. 1029–1034, 2020.
- [17] S. A. A. S. A.M.Shah X. Yan and G. Mamirkulova, “Mining patient opinion to evaluate the service quality in healthcare: A deep-learning approach, *journal of ambient intelligence and humanized computing*, jourvol 11, number 7,” pp. 2925–2942, 2020.
- [18] G. J. d. S. F.R.Lucini F. S. Fogliatto *et al.*, “Text mining approach to predict hospital admissions using early medical records from the emergency department, *international journal of medical informatics*, jourvol 100,” pp. 1–8, 2017.
- [19] K. Doing-Harris, D. L. Mowery, C. Daniels, W. W. Chapman, and M. Conway, “Understanding patient satisfaction with received healthcare services: A natural language processing approach,” *AMIA Annual Symposium Proceedings*, vol. 2016, p. 524, 2016.
- [20] F. Greaves, D. Ramirez-Cano, C. Millett, A. Darzi, L. Donaldson, *et al.*, “Use of sentiment analysis for capturing patient experience from free-text comments posted online,” *Journal of Medical Internet Research*, vol. 15, no. 11, e2721, 2013.
- [21] D.B.Neill, “Using artificial intelligence to improve hospital inpatient care, *iee intelligent systems*, jourvol 28, number 2,” pp. 92–95, 2013.
- [22] S. Gohil, S. Vuik, and A. Darzi, “Sentiment analysis of health care tweets: Review of the methods used,” *JMIR Public Health Surveill*, vol. 4, no. 2, e43, 2018. DOI: 10.2196/publichealth.5789.
- [23] M.Sivakumar and U.S.Reddy, “Aspect based sentiment analysis of students opinion using machine learning techniques,” pp. 726–731, 2017.
- [24] K. Denecke and D. Reichenpfader, “Sentiment analysis of clinical narratives: A scoping review,” *Journal of Biomedical Informatics*, vol. 140, p. 104336, 2023, ISSN: 1532-0464. DOI: <https://doi.org/10.1016/j.jbi.2023.104336>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1532046423000576>.
- [25] S. Ahmed and A. Danti, “Effective sentimental analysis and opinion mining of web reviews using rule based classifiers,” in *Computational Intelligence in Data Mining—Volume 1*, 2015. DOI: 10.1007/978-81-322-2734-2_18.
- [26] T. Liu, Y. Ma, and X. Yang, “Service quality improvement of hospital reservation system based on text sentiment analysis,” in *2018 9th International Conference on Information Technology in Medicine and Education (ITME)*, 2018, pp. 289–293. DOI: 10.1109/ITME.2018.00071.

- [27] “Reviews and price on online platforms: Evidence from sentiment analysis of airbnb reviews in boston,” *Regional Science and Urban Economics*, vol. 75, pp. 22–34, 2019, ISSN: 0166-0462. DOI: <https://doi.org/10.1016/j.regsciurbeco.2018.11.003>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S016604621730340X>.
- [28] D. S. Panchal, S. S. Kawathekar, and S. N. Deshmukh, “Sentiment analysis of healthcare quality,” *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, vol. 9, no. 3, pp. 1–5, 2020, ISSN: 2278-3075.
- [29] A. I. A. Rahim, M. I. Ibrahim, K. I. Musa, S.-L. Chua, and N. M. Yaacob, “Assessing patient-perceived hospital service quality and sentiment in malaysian public hospitals using machine learning and facebook reviews,” *International Journal of Environmental Research and Public Health*, vol. 18, no. 18, p. 9912, 2021. DOI: 10.3390/ijerph18189912.
- [30] A. I. A. Rahim, M. I. Ibrahim, K. I. Musa, S.-L. Chua, and N. M. Yaacob, “Assessing patient-perceived hospital service quality and sentiment in malaysian public hospitals using machine learning and facebook reviews,” *International Journal of Environmental Research and Public Health*, vol. 18, p. 9912, 18 2021. DOI: 10.3390/ijerph18189912. [Online]. Available: <https://doi.org/10.3390/ijerph18189912>.
- [31] N. Zaman, D. M. Goldberg, S. Kelly, R. S. Russell, and S. L. Drye, “The relationship between nurses’ training and perceptions of electronic documentation systems,” *Nursing Reports*, vol. 11, no. 1, pp. 12–27, 2021. DOI: 10.3390/nursrep11010002.
- [32] G. Alexander, M. Bahja, and G. F. Butt, “Automating large-scale health care service feedback analysis: Sentiment analysis and topic modeling study,” *JMIR Med Inform*, vol. 10, no. 4, e29385, Apr. 2022, ISSN: 2291-9694. DOI: 10.2196/29385. [Online]. Available: <http://www.ncbi.nlm.nih.gov/pubmed/35404254>.
- [33] A. Jain, A. Bansal, and S. Tomar, “Aspect-based sentiment analysis of online reviews for business intelligence,” *International Journal of Information Technologies and Systems Approach (IJITSA)*, vol. 15, no. 3, pp. 1–21, 2022. DOI: 10.4018/IJITSA.307029. [Online]. Available: <http://doi.org/10.4018/IJITSA.307029>.
- [34] A. Hamdi, K. Shaban, and A. Zainal, “Authors info & claims,” *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 17, no. 4, pp. 1–28, DOI: 10.1145/3209885. [Online]. Available: <https://doi.org/10.1145/3209885>.
- [35] A. Shah, X. B. Yan, and S. A. Shah, “Tracking patients healthcare experiences during the covid-19 outbreak: Topic modeling and sentiment analysis of doctor reviews,” *Journal of Entrepreneurship and Rural Development*, vol. 9, no. 3A, DOI: 10.36909/jer.v9i3A.8703.