SABRE – Sentiment analysis-based recommendation system for electronic products

A Project Report

Submitted by

ADITYA NAIR, JANYA PANDYA, CHRISTOPHER PARALKAR, YASH CHOPRA

Under the Guidance of

PROF. DEEPA KRISHNAN

in partial fulfillment for the award of the degree of

BACHELORS OF TECHNOLOGY

COMPUTER ENGINEERING

At

MUKESH PATEL SCHOOL OF TECHNOLOGY MANAGEMENT AND ENGINEERING

April, 2021

DECLARATION

We, Aditya Nair, Janya Pandya, Christopher Paralkar, Yash Chopra, Roll No.s E003, E010, E013, E071, B.Tech (Computer Engineering), VII semester understand that plagiarism is defined as anyone or combination of the following:

1. Un-credited verbatim copying of individual sentences, paragraphs or illustration (such as graphs, diagrams, etc.) from any source, published or unpublished, including the internet.

2. Un-credited improper paraphrasing of pages paragraphs (changing a few words phrases, or rearranging the original sentence order)

3. Credited verbatim copying of a major portion of a paper (or thesis chapter) without clear delineation of who did wrote what. (Source: IEEE, The institute, Dec. 2004)

4. I have made sure that all the ideas, expressions, graphs, diagrams, etc., that are not a result of my work, are properly credited. Long phrases or sentences that had to be used verbatim from published literature have been clearly identified using quotation marks.

5. I affirm that no portion of my work can be considered as plagiarism and I take full responsibility if such a complaint occurs. I understand fully well that the guide of the seminar/ project report may not be in a position to check for the possibility of such incidences of plagiarism in this body of work.

Signature of the Student:

Name: Aditya Nair, Janya Pandya, Christopher Paralkar, Yash Chopra

Roll No: E003, E010, E013, E071

Place: Mumbai, Maharashtra, India

Date: 15/04/2021

CERTIFICATE

This is to certify that the project entitled "SABRE – Sentiment Analysis Based Recommendation system for Electronics" is the bonafide work carried out by Aditya Nair, Janya Pandya, Christopher Paralkar, Yash Chopra of B.Tech (Computer Engineering), MPSTME (NMIMS), Mumbai, during the VII semester of the academic year 2020-21, in partial fulfillment of the requirements for the award of the Degree of Bachelors of Engineering as per the norms prescribed by NMIMS. The project work has been assessed and found to be satisfactory.

_____

Prof. Deepa Krishnan

Internal Mentor

_____ _____

Examiner 1 Examiner 2

_____

Dean

Table of contents

List of Figures

List of Tables

****

Abbreviations

Abbreviation	Description
ML	Machine Learning
NLP	Natural Language Processing
AI	Artificial Intelligence
LDA	Linear Discriminant Analysis
KNN	K Nearest Neighbor
HTML	Hypertext Markup Language
DOM	Document Object Model
RAPIER	Robust Automated Production of Information Extraction Rules
SRV	Sequence Rules with validation
JSP	JavaScript Object notation
CSV	Comma Separated Values
MAE	Mean Absolute Error
RMSE	Root Mean Square Error
SVM	Support Vector Machine

Abstract

Recommendation systems are ubiquitous these days and are used in nearly every domain; from learning which videos could be recommended to users on streaming websites, to products that can be sold on e-commerce platforms. These systems are driven by the copious amount of data that is scraped and collected from sources such as review platforms and social media websites. On this collected data, sentiment analysis can be performed to recommend products to users based on an overall analysis of sentiments conveyed using reviews, comments or opinions. The information thus obtained is provided to already existing machine learning based filtering techniques which include content-based, collaborative and hybrid filtering.

The preliminary step to initiate any process always involves the gathering of relevant data. There are several techniques out of which we will be summarizing ones which have been used for collection of data from various websites. We have collected data from two data sets, one for the headphone and another for the laptop which we have collected using web scraping techniques reviewed by us. This is followed by various pre - processing techniques where widely available libraries are used to make the dataset ready for further steps. Next there is sentiment analysis. Sentiment analysis also known as emotion detection is a process which uses NLP (Natural Language Processing) and classification models to determine the sentiments behind verbal or textual data. It has various applications such as mining social data, determining product reputation, and understanding customer requirements. It does so by detecting the polarity of the data that can be in the form of a document, paragraph or a sentence. Sentiment analysis has been performed by us using classifiers like Logistic Regression, Naive Bayes, Random Forest, XGBoost and Catboost. We have further compared the result, displaying the best algorithm. Finally, there is a recommendation system. Recommendation systems are algorithms aimed at suggesting relevant items to users.

Firstly, this report showcases the literature survey conducted by us where we are comparing and contrasting results out of twenty-one papers reviewed by us. Post this, we have explained the analysis and design part using various software engineering principles to explain our work. This would include various design analysis done by us throughout the two semesters. We have then shared information about the implementation done where we highlighted how we solved the problem of data sparsity. We finally conclude our work along with mentioning the scope of improvement and future work which can be done.

Chapter1

Introduction

1.1 Project Overview

1.1.1 Why SABRE?

It has been proven that starting anything by answering WHY? brings more attention therefore we would start our report by answering WHY just as Apple does. This era of growing technology is rightly coined as the "Digital Age", which is often characterized by the abundance and availability of information. Online shopping is becoming popular day by day because of the low cost, effective logistic systems and variety. However, this abundant diversity leads to uncertainty in quality and indecisiveness which is the source of confusion for many customers. To find an answer to this question customers generally look up reviews and opinions on websites and analyse them manually. This is not proportionate to the energy and time which it requires.

1.1.2 Problem and Motivation

In recent years lots of research has been done for providing useful recommendations to the customers. Many of these recommendation systems are based on the sentiment analysis of reviews and social media content. During this period, the researchers have come across problems such as data sparsity. Realization of the problems being factor dependent has led to the development of techniques that are application specific or problem specific. We have come up with one such solution to this problem.

`

1.2 Aim

1. *To develop a Recommender system based on sentiment analysis of textual reviews of electronic products from various sources for those customers who shop on e-commerce websites.*

2. *To solve the problem of data sparsity by replacing the missing values with the most immediate value.*

1.2.1 Purpose of the project

The purpose of this Project is to improve upon the pre-existing ratings-based product recommendation systems through embedding sentiment analysis of text reviews by collection of users' sentiment data from multiple platforms. User ratings will then be assigned to improve the quality of recommendations for the customers who are looking for similar products on an e-commerce website.

1.2.2 Scope

The scope of this project lies within providing appropriate recommendations based on the users' sentiments about electronic products which would be analysed using a trained machine learning. We would bridge the gap between popular opinion on social media and reviews given by users on websites. A proper recommendation system can be applied on various platforms. It can be used in many sectors from the entertainment sector to the health sector. A proper domain specific system with good accuracy is what our product would achieve.

1.2.3 Purpose of the report

The purpose of this report is to highlight our work in the field on sentiment analysis and filtering. We would like to demonstrate our work done during the year 2020-2021, the major obstacles encountered by us and how we tackled them. This will also create a foundation for anyone who wishes to start research in sentiment analysis/ filtering. It will help a new researcher to be pre-aware about the issues they might encounter and help them to save a lot of time.

1.3 Hardware Specification

The project makes use of high end computers that must have these specifications:

- Ram: 8 GB
- Processor: Intel i7
- Hard disk: 100GB
- Speed: Up to 4.1 GHz

1.4 Software Specification

- Operating System: Windows, Linux or Mac
- Anaconda: Jupyter Lab, Google Colab
- Python interpreter: python 3.7 or higher
- Octoparse: Latest version

`

Chapter 2

Review of Literature

2.1 Literature Review

The following is a summary of every paper that we researched upon, along with our key findings and drawbacks.

Table 2.1 : A tabulated literature review

Paper citation	Key findings	Drawbacks
1. Amel Ziani, Nabiha Azizi, Didier Schwab, Monther Aldwairi, Nassira Chekkai, et al.. "Recommender System Through Sentiment Analysis. 2nd International Conference on Automatic Control, Telecommunications and Signals", Dec 2017, Annaba, Algeria. Ffhal-01683511	Model used: Semi-supervised support vector machine for opinion analysis. Techniques: Spearman similarity (k nearest neighbors) for recommendation, feature extraction and emotionalism for sentiment analysis, mean absolute error (precision and recall) for result analysis Pros: Multilingual so it is flexible between Arabic, English and French datasets, excellent hybrid of collaborative filtering and social filtering to eliminate any weaknesses. Precision from result analysis is almost in the range of 0.90 to 0.96 for Arabic and English dataset. For the French dataset the precision was 1.0	Social media opinion not taken into consideration. No solution to cold start problem or data sparsity problem given
2. Alia Karim Abdul Hassan, Ahmed Bahaa aldeen abdulwahhab "Reviews Sentiment analysis for collaborative recommender system", Kurdistan Journal of	Model: Naïve Bayes, logistic regression and decision tree on3 datasets. Techniques: Discussion about various NLP and probabilistic classifiers, usage of the NLTK library in anaconda for	Suffers from cold start problem Only collaborative filtering and no other type has been used for recommendation. Overall accuracy among the three datasets while using three

`

Applied Research, August 2017, Volume 2 Issue 3	sentiment analysis, computation of the confusion matrix, precision and recall based result analysis. Pros: Bilingual as it supports Arabic and English datasets, logistic regression for IMDB reviews yielded healthy accuracy of 0.89.	techniques which we calculated was an average of 0.77 which is mediocre and insufficient. Social media opinion not taken
3. Xiaojiang Lei, Xueming Qian, Member, IEEE, and Guoshuai Zhao "Rating Prediction Based on Social Sentiment From Textual Reviews"	Techniques: Linear Discriminant Analysis (LDA) used to extract product features from textual reviews. Root mean square error (RMSE), Mean absolute error (MAE) for result analysis Pros: Analysis of social friend circle done in building a better recommender system. Sentiment analysis done at Review level, sentence level and phrase level. Concepts like stop word, noise word used in creating a more efficient lexon.	Domain specific analysis not done. No solution for data sparsity given. No clear approach about segregation of ambiguous words.
4. N. A. Osman, S. A. M. Noah, and M. Darwich "Contextual Sentiment Based Recommender System to Provide Recommendation in the Electronic Products Domain"	Models: Three type of matrix (ratings-based CF, sentiment CF contextual sentiment CF) Techniques: Root mean square error (RMSE), Mean absolute error (MAE) for result analysis Pros: Minimized the level of data sparsity	None Noticed
5. Nurul Aida Osman and Shahrul Azman Mohd Noah "SENTIMENT-BASED MODEL FOR RECOMMENDER SYSTEMS"	Models: Integration of textual review into CF recommendation. Techniques: integration of rating items and textual reviews.	Inefficient approach for sentimental analysis. No clear approach about segregation of ambiguous words.

`

	Result measured by Root mean square error (RMSE) Pros: Detailed information about type of recommendation system. Data Sparsity problem approached efficiently.	Domain specific analysis not done.
6. Nyein Ei Ei Kyaw, Thinn Thinn Wai "Inferring User Preferences Using Reviews for Rating Prediction"	Models: Memory-based CF (User-User CF), Model-based CF (Matrix Factorization) Technique: User-User Collaborative filtering, Matrix Factorization, lexicon-based Sentiment analysis Pros: Predicts unknown ratings using user's reviews	None Noticed
7. R. Lydia Priyadharsini, M. Lovelin Ponn Felciah" Recommendation System in E-Commerce using Sentiment Analysis" International Journal of Engineering Trends and Technology (IJETT) – Volume 49 Number 7 July 2017	Models: Hybrid model (content + collaborative filtering) Techniques: Collaborative filtering and content-based filtering techniques where implemented individually and together, precision and recall were used as performance metrics Pros: The results were evaluated on real time user data Mac Address based filtering was used to remove fake reviews Emoticons were also considered Cold start problem was approached efficiently	Dataset was constrained to 1-2 products Other approaches were not considered
8. Xing Fang* and Justin Zhan" Sentiment analysis using product review data" Fang and Zhan Journal of Big Data (2015) 2:5	Models: Naïve Bayesian, Random Forest, Support vector machines for sentence level categorization- manually labeled sentences and machine labeled sentences and review level categorization. Techniques: Sentiment sentences extraction and pos tagging, negation phase identification, sentiment score computation, feature vector formation Pros:	Assumed that amazon reviews are spam free and considered almost all of them for sentiment analysis

`

	Addressed problems of adverbs before the featured words example not worth not working etc. Combined ratings with reviews to generate more accurate sentiment scores.	
9. R. Diouf, E. N. Sarr, O. Sall, B. Birregah, M. Bousso and S. N. Mbaye, "Web Scraping: State-of-the-Art and Areas of Application," 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, 2019, pp. 6040-6042, doi: 10.1109/BigData47090.2019.9005594.	Models: Software platforms like import.io, easy web extract, Fminer, Weboob etc. Techniques: Mimicry, Weight measurement, Differential approach, Machine learning approach, browser extensions, programming language libraries Pros: Comprehensive review of different methods for web scraping, nearly all details mentioned for various modes of usage depending upon programming expertise, areas of application mentioned help in deciding which technique is best suited	Exact approach for data extraction step is not given
10. D. M. Thomas and S. Mathur, "Data Analysis by Web Scraping using Python," 2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 2019, pp. 450-454, doi: 10.1109/ICECA.2019.8822022.	Models: Python 3.6 along with Scrapy, Techniques: Use a python script to extract data from source URL then use another one to analyze data. Pros: Simple script language used, good visualization of data content, easy to code and implement	Restricted to a single website. Possibly hard for mass collection of data scraping across multiple websites and domains. No technique used to filter unwanted data after collection, during analysis``
11. M. S. Parvez, K. S. A. Tasneem, S. S. Rajendra and K. R. Bodke, "Analysis Of Different Web Data Extraction Techniques," 2018 International Conference on Smart City and Emerging Technology (ICSCET),	Models: WIEN, WHISK, Rapier (Robust Automated Production of Information Extraction Rules), SRV (Sequence Rules with validation) all machine learning approaches for data extraction Techniques: Human Copy paste, HTML Parser, Semantic annotation, tree-based	Practical implementation approach is not given

`

Mumbai, 2018, pp. 1-7, doi: 10.1109/ICSCET.2018.8537333.	technique, Web wrappers Pros: Gives view of both web crawling and web data extraction, explains data extraction and conversion of unstructured data into structured data, gives a generalized view of data extraction independent of programming language constraints, different approaches for ML, concludes that web wrapping is the best technique for extraction	
12. S. Sharma, A. Sharma, Y. Sharma and M. Bhatia, "Recommender system using hybrid approach," 2016 International Conference on Computing, Communication and Automation (ICCCA), Noida, 2016, pp. 219-223, doi: 10.1109/CCAA.2016.7813722.	Techniques: Content based Approach Collaborative based approach: - User based, item based, hybrid based Pros: The proposed Composite search algorithm uses the hybrid-based approach. The disadvantage of one filtering approach is overcome by applying the other filtering technique on the result of the former technique. Reduces the exposure of users' personal information.	The Composite Search Algorithm does not take user's search history into consideration.
13. P. Venil, G. Vinodhini and R. Suban, "Performance Evaluation of Ensemble based Collaborative Filtering Recommender System," 2019 IEEE International Conference on System, Computation, Automation and Networking (ICSCAN), Pondicherry, India, 2019, pp. 1-5, doi: 10.1109/ICSCAN.2019.8878777.	Techniques: Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) used for result analysis Pros: Addresses the problem of data sparsity and problem of scalability in conventional collaborative filtering algorithms.	Only a small dataset is used.
14. Zeenia Singla, Suk Chandan Randhawa, and Sushma Jain,	Techniques: Statistical analysis of dataset which had parameters such as rating distribution by brand, review counts, review	Whilst they have taken a good amount of data for their predictive model, they have

`

Statistical and sentiment analysis of consumer product reviews, IEEE – 40222	length, positive and negative review distribution, word clouds, Context based sentiment analysis based upon parameters such as fear, joy, anger, anticipation etc. Pros: Generally sentiment analysis is done by dividing the sentiments into positive and negative in this they have also considered other sentiments like trust, joy, anticipation along with positive and negative.	worked on just SVM which may or may not be the best model for the prediction
15. Kim Schouten and Flavius Frasincar, Survey on Aspect-Level Sentiment Analysis, IEEE transactions on knowledge and data engineering, vol. 28, no. 3, march 2016	Techniques: A survey paper which had a lot many techniques for aspect detection and sentiment analysis which were broadly based on supervised machine learning models, unsupervised machine learning models, dictionary-based models, frequency models and hybrid models. Pros: It compares a lot of techniques with credible performance metrics such as precision recall, ranking score accuracy	Since this was a survey aper which discussed about different techniques pertaining to sentiment analysis and combining aspect detection with sentiment analysis we could not find any substantial shortcomings
16. Recommendation System using Lexicon Based Sentimental Analysis with collaborative filtering, 2020 International Conference on Power Electronics & IoT Applications in Renewable Energy	Naïve-Bayes classification model. Techniques: Sentiment sentences extraction and pos tagging, negation phase identification.	Hu and Liu Opinion Lexicon, a list of 6800 positive and negative words used to score the text of the tweets based on how many of the "bad" and "good" words show up.
17. Shipra Goel, Muskan Banthia, Adwitiya Sinha "Modeling Recommendation System for Real Time Analysis of Social Media	Model used: Naïve-Bayes classification model. Techniques: d: Tweepy is a twitter API used in Python to provide access to the entire twitter data.	Drawbacks: Accuracy of only 77% achieved with Naïve-Bayes algorithm. Only twitter data used for analysis.

`

Dynamics"	Hu and Liu Opinion Lexicon, a list of 6800 positive and negative words used to score the text of the tweets compared to how many of the "bad" and "good" words show up in each. Visualization technique in the form of word cloud used to display the analysis. Pros: Detailed information about web scrapping and analysis techniques used given. 95% accuracy received for sentiment-based score	
18. R.M. Gomathi P.Ajitha G. Hari Satya Krishna I. Harsha Pranay "Restaurant Recommendation System for User Preference and Services Based on Rating and Amenities"	Model used: Natural Language Processing based model Techniques: Data collection done manually. User reviews segregated as positive, negative and neutral using a specific lexicon. Recommendation against various other existing approaches such as SVM, PNN and BPN. The NLP (traditional algorithm) used for mining. To implement rating and discover methods, metrics such Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) Pros: Analysis using various algorithms done to select the best algorithm.	Drawbacks: Analysis done only based on reviews on the website and amenities. Collaborative filtering would increase the accuracy by a good margin.
19. Recommendation System using Lexicon Based Sentimental Analysis with collaborative filtering" Rahul Pradhan, Vedant Khandelwal, Ankur Chaturvedi, Dilip Kumar Sharma"	Model used: Hadoop framework Techniques: Collaborative filtering is used. User based and item-based filtering done. User based filtering uses Pearson correlation and prediction function. Item based filtering uses similarity metrics and prediction function. Pros: Collaborative filtering reviewed well. Effective visualization for better recommendation support done.	Drawbacks: Segmentation of users not done. Sentimental analysis not done using predefined effective lexicons.

`

2.2 Review Summary

Data Extraction:

There are browser extensions such as Spider, data scrapers like Weboob and import.io which do not require much knowledge in programming and have a flexible interface for converting them into various formats such as JSON (JavaScript Object notation) or CSV (Comma Separated Values). Alternatively, there are programming language libraries such as those of NodeJS and Java, a minor drawback of them being that they are not suitable for a layman and only a seasoned specialist may be able to use them to their full potential. [9] Data analysis using python [10] is also an efficient way not just to scrape data, using software like Scrapy but also using a snippet of code to analyze the stored data. A minor problem is the lack of uniformity and the dynamic nature of web pages from which data needs to be extracted, which makes extraction a hard process.

There are several different ways to further refine data extraction, by understanding the parsing structure of HTML (Hypertext Markup Language) pages. This involves the DOM (Document Object Model) based tree structure as well as Beautiful Soup, a python library that can extract certain parts of the content from the web that can eliminate the HTML tags from them. Another method is wrapper classes, which allows the user to specify a particular algorithm, the wrapper finds the relevant web pages and can convert unstructured to structured data.

Extraction tools	Techniques used
Software platforms like import.io, easy web extract, Weboob etc [11]	• Mimicry • Weight measurement • Differential approach
Machine learning based tools Rapier and SRV [9]	• HTML Parser • Semantic annotation • tree-based technique
Python based Web scrapers/crawlers [10]	• Scrapy • Beautiful Soup

Table 2.2: Data Extraction Techniques

`

Sentiment Analysis:

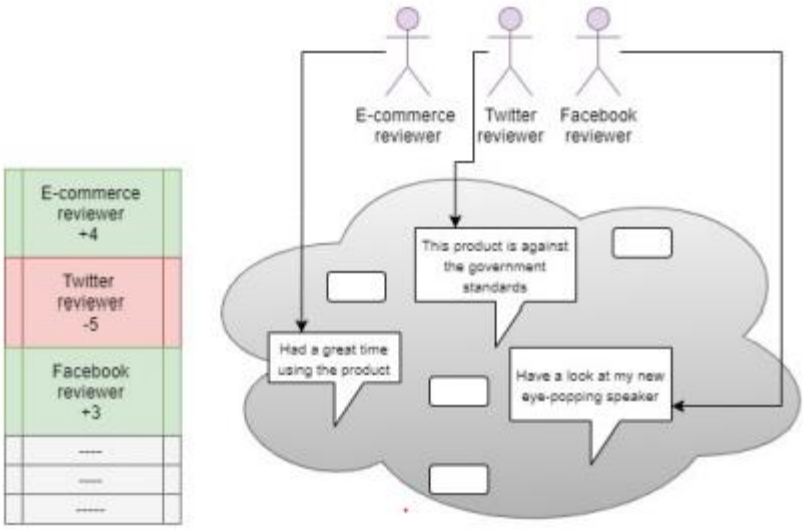


Figure 2.1: Sentiment Analysis framework

Types of Sentiment analysis:

1) **Fine grained:** It is performed at a sentence or subsistence level,it concerns the polarity of the data which can be categorized into strongly negative/positive, weakly positive/negative, neutral, positive, negative. For example, product, review ratings which range from 1-5(negative-positive). [12] [13]

2) **Emotion detection:** This is used to classify basic and complex natural human emotions involving happiness, sadness and anger. It is performed from textual or verbal data using lexicons or classification algorithms.

3) **Aspect-based Sentiment Analysis :** Also known as context based sentiment analysis, it is mainly used in analyzing sentiment of texts, for example, this model aims to determine which aspect or feature of a product in product review is being mentioned in a positive, neutral or a negative way. [2] [18]

4) **NLP methods and algorithms**

- **Rule Based: Rule Based**: Use a set of human-crafted rules such as stemming, lexicons, counts and so on, that help in identifying the contextuality, polarity, or emotion. [6] [14]

`

- **Automatic Approaches**: These are machine learning based approaches that use classification algorithms to generate rules which help in identifying the sentiment behind a review or a speech. Some of the classification algorithms used are

    - **Naive Bayes:** These classifiers are a simple family of probabilistic classifiers, that predict the sentiment of the text using Bayes's theorem. [1] [15]

    - **Logistic Regression:** It is a statistical supervised machine learning model that uses a logistic function to classify a binary dependent variable [16]

    - **SVM (Support Vector Machine):** A similarity driven model that, it takes the input data as points and creates a Ndimensional space (N-Number of features), which are used to find a hyperplane that precisely classifies all the data points, for example, different sentiments are mapped to different regions and new texts, are assigned classes based on its similarity to a particular region. [4] [15] [16]

    - **Decision Tree:** Use top-down tree like structure which classifiy the data points based on "if-then" rules, these rules are generated sequentially from input data. The features at the top greatest impact on the decision. [21]

Naïve Bayes	• Sentence level categorization and feature vector formation [21] • Segregation based on how many "bad" and "good" words show up [15]
Logistic Regression	• Used for probabilistic classification [8]
Decision tree, Random Forest	• Division by appearance or absence of a word to classify a document [8] • Random forest is an ensemble method that generates a multitude of decision trees classifies based on the aggregated decision of those trees. [21]

`

Algorithms	Techniques used
Rule based	• Hu and Liu Opinion Lexicon (Dictionary of good and bad words) based frequency model [6]
SVM	• Sentence level categorization and feature vector formation [21] • Feature extraction and emotionalism [16] • Aspect/context-based sentiment analysis [2]

Table 2.3: Sentiment analysis techniques

Table below presents the comparison of classification algorithms based on qualitative factors such as type, quality, quantity of data and time complexity.

Algorithm]	Qualitative Analysis
Naïve Bayes	Supports effective and highly scalable model building along with scoring, scales linearly with the number of predictors and rows.
Support Vector Machine	SVM is found effective in high dimensional spaces, it is not suitable for large data sets. Conversely it doesn't work well when dataset has a lot of noise.
Decision tree	Scaling of data is not required. Decision trees requires less effort for data preparation during pre-processing compared to other algorithms
Logistic regression	It makes no assumptions about distributions of classes in feature space. It is very fast at classifying unknown records. Conversely it is tough to obtain complex relationships using logistic regression.

Table 2.4: qualitative comparison of classification models

`

Fig below presents the quantitative analysis involving precision, recall and accuracy of these classification algorithms when applied on the IMDB data set consisting 1568200 data values.

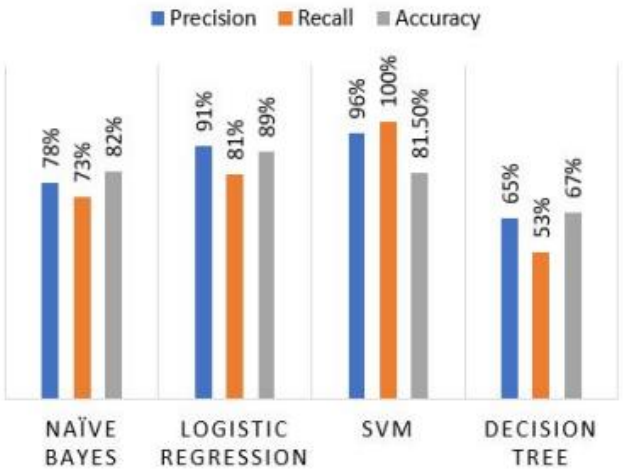


Figure 2.2: Comparison of precision, recall and accuracy on IMDB data set

Similarly information filtering techniques are compared based on their efficiency in dealing with problems such as data sparsity and cold star .Simultaneously we have compared them when they were applied on the same data set

Algorithm	**Qualitative Analysis**
User based KNN	It is not context dependent making it more reliable whereas sparsity is a major issue because most of the percentage of people who rate items is really low.
Item based KNN	Domain knowledge is not required because the embeddings are automatically learned and Cannot handle fresh items and hard to include side features for query or item

Table 2.5: Result and analysis of filtering techniques.

`

Fig below represents the quantitative analysis of RMSE and MAE of the algorithms when applied to Movie lens data set with 9000 instances.

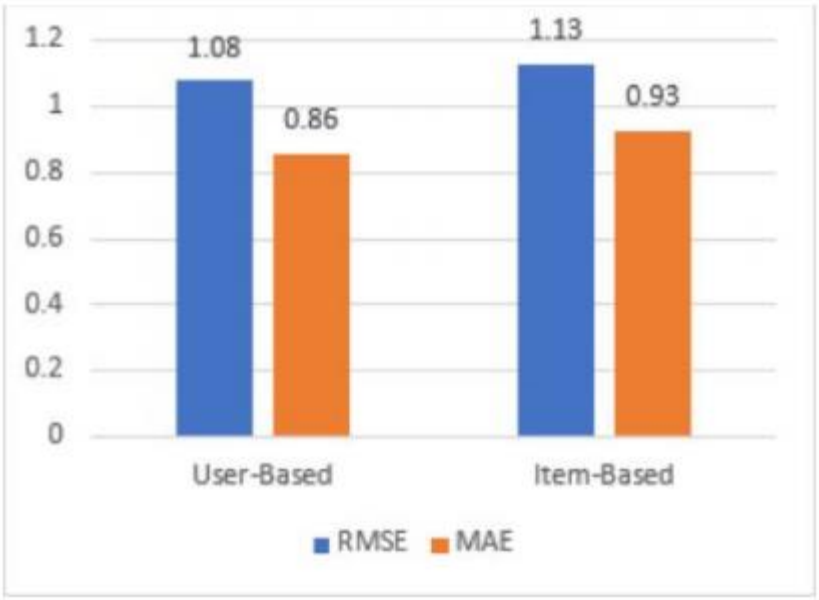


Figure 2.3: Comparison of MAE and RMSE values for Movie Lens data set

`

Chapter 3

Analysis and Design

3.1 Analysis

3.1.1 Overall Framework

The general theoretical framework for a sentiment based recommendation system is demonstrated in the figure:

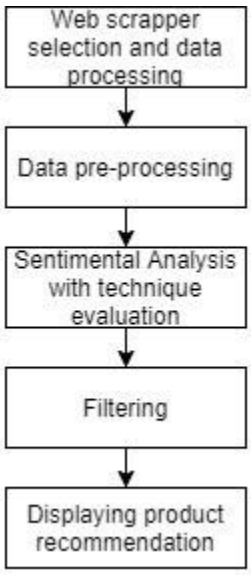


Figure 3.1: Framework

Data Gathering or data extraction is an essential part of a recommendation system; surveys, social media platforms and review websites have become important sources of such data. There are two different approaches to achieve this; one is by using a programming language script in order to extract data, the other is using a pre-built software to provide customization to the same.

The field of sentiment analysis bridges the gap between the extracted data, NLP (natural language processing) and AI (artificial intelligence). It is often referred to as “opinion mining”. As the name suggests its main purpose is to determine the opinion or the sentiments of a

`

person/group on a particular subject. Recommendation systems predicts the rating or the preference, that a user is likely to give to an item. With the help of these predictions, recommendations can be provided to the users about the items they might like. It is broadly divided into three approaches, namely: Content based approach, Collaborative approach and hybrid approach. ML approaches such as KNN (K nearest neighbor is used with the sentiment analysis output procured previously.

3.1.2 Implementation Approach

The exact steps have been shown in the figure below : (currently we are on the blue level)

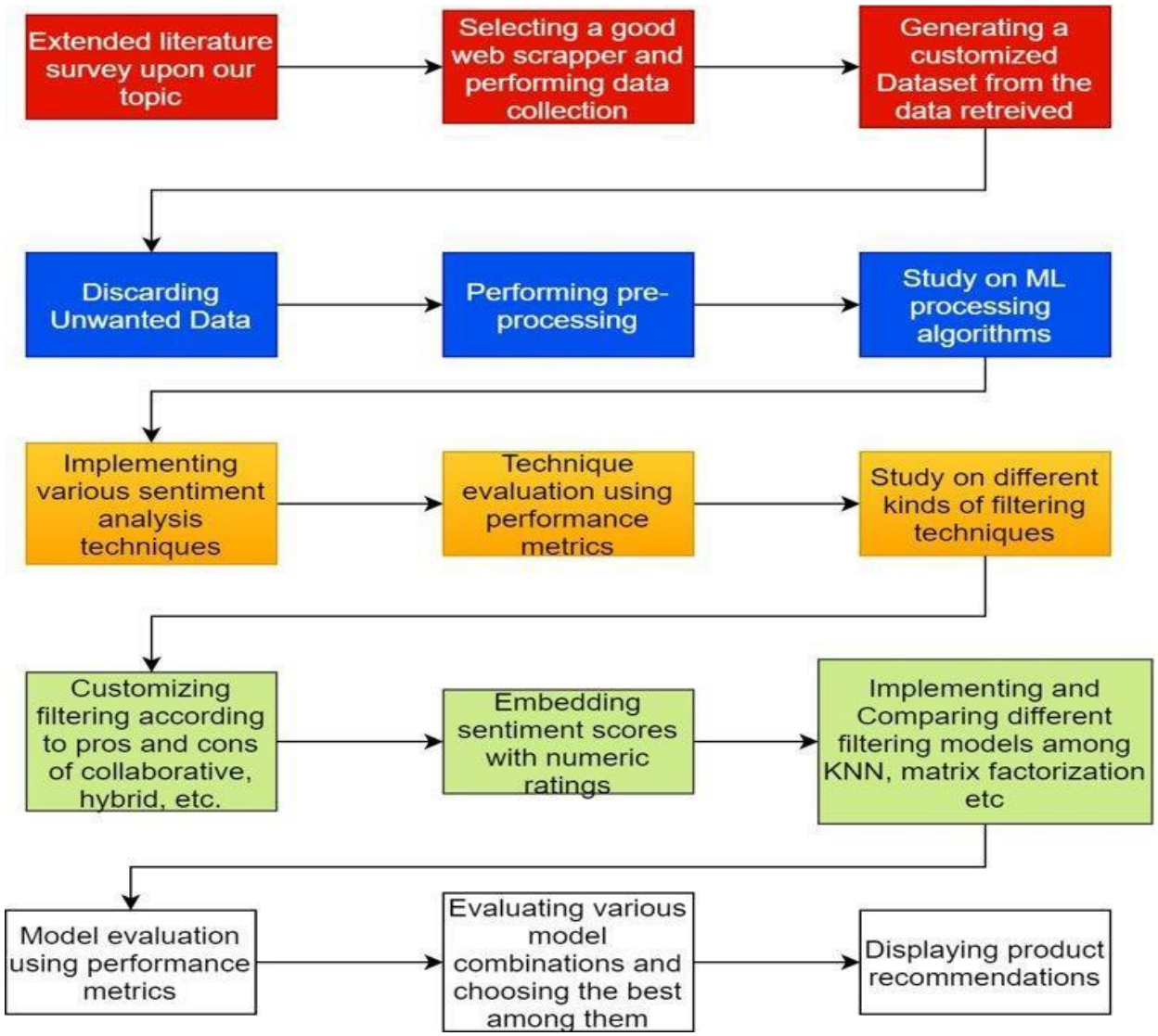


Figure 3.2 : Implementation Plan

3.2 Design

3.2.1 Architecture Diagram

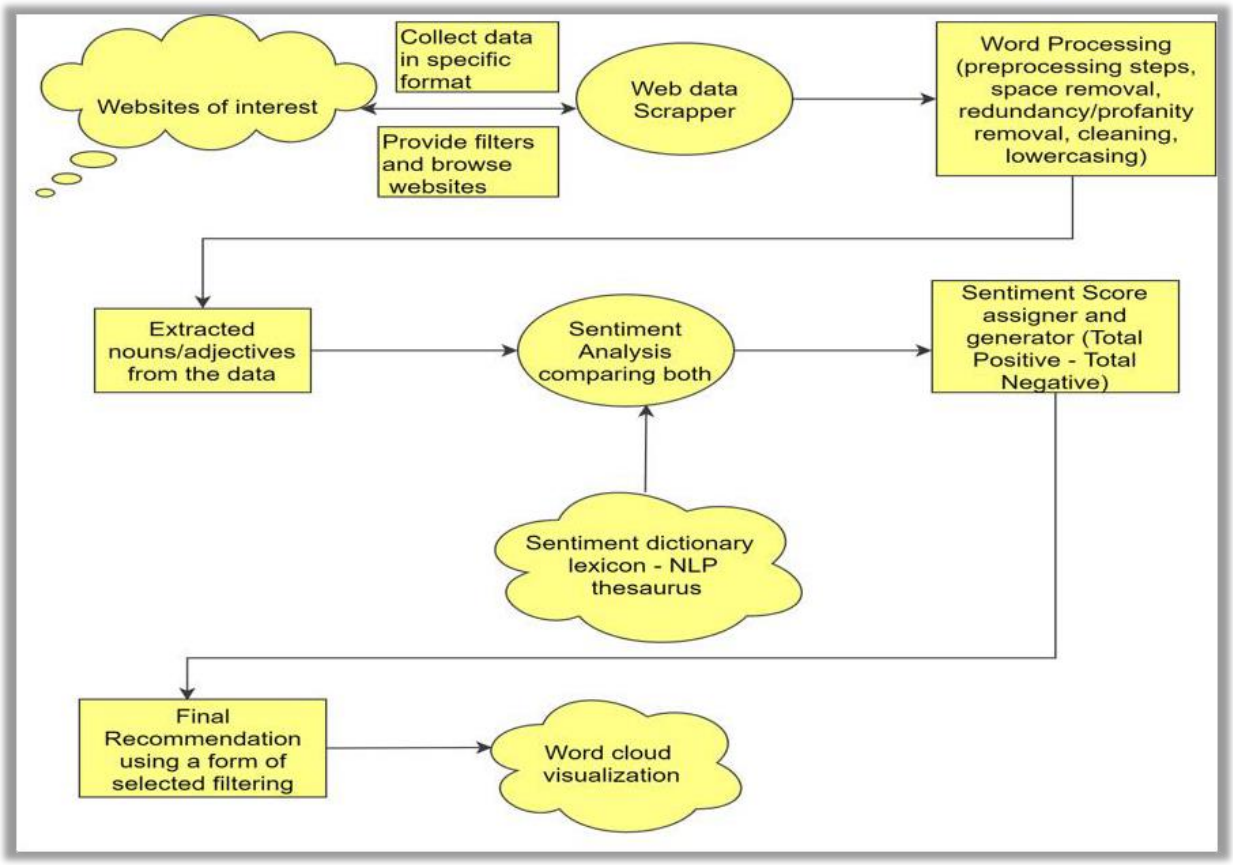


Figure 3.3 : Architecture

Based on the research papers collected, a literature survey would be drafted summarizing all the different research papers. After drafting the literature survey, the collection and studying of data will provide us with the insights of the type of data which was collected and kind of preprocessing techniques which will be required to clean and process the data. The collection of data would be through a web scraper tool.

We would then preprocess the data, i.e., cleaning the data, removing redundancies, etc in order to make the data more suitable for processing. After preprocessing the data, we would perform word and phrase extraction in order to get the required data. Using an already available sentiment dictionary, we would then apply sentiment analysis and assign a sentiment score to each of the words or phrases. For example, Positive words or phrases will be assigned 1, neutral words or

phrases will be assigned 0 and Negative words or phrases will be assigned –1.

Next step would be studying the various machine learning models which can be or were used in previous recommendation systems. This analysis will help us to understand which model provides us the best results and will be the most suitable to be used for recommendation.

After performing sentiment analysis, we would then incorporate the sentiment scores along with a filtering approach that we have finalized upon, which is filtering out products a user might like based on reviews made by other similar users, his past preferences and then displays the recommended products.

3.2.2 Use – Case Diagram

Figure 3.4 : Use Case

The diagram above visualizes the different actor or users of the recommender system (SABRE) and the way in which they use the system and the diagram also shows the internal process of recommendation done by the system. The actors involved are a target group, which includes a customer and reviewer, and an analyst group which includes both a marketing and a product

analyst. The customer group will most likely purchase products based on recommendation by SABRE system. The reviewer group will review the products.

A product analyst would most likely analyze the product's reception and try to improve the product's public reception, while a marketing analyst will try and find new ways to market the products. These two analyst groups can occasionally work in tandem and try to improve the product's public reception and improve upon it while also trying to market the group to various public demographic.

3.1.1 Sequence Diagram

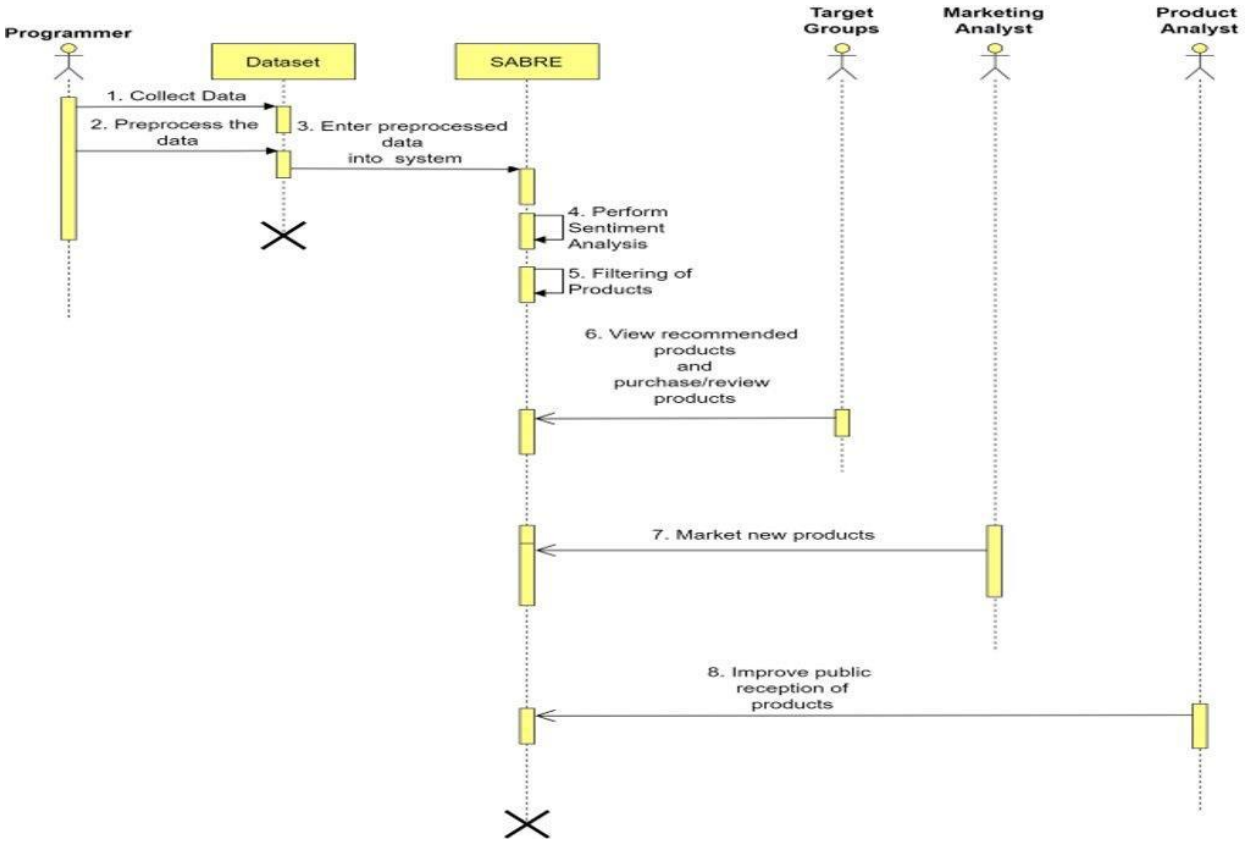


Figure 3.5: Sequence

The diagram demonstrates the normal working sequence of the SABRE system. As shown, the internal processing of the system is synchronous in nature. The results of a synchronous process are affected by the other synchronous processes.

The external processes, such as what is done with the recommendations provided by the system are all asynchronous in nature and one process's results are not affected by the other processes.

The working sequence of the SABRE system is as follows:

Initially, the data required for recommending a product is collected from various sources and pre-processing is done on it in order to make it suitable for processing. After processing the data, the preprocessed data is entered into the system, where sentiment analysis in order to find the highly rated products. The collection and preprocessing of data are done by the programmer or developer. For recommendation of the products to the users, filtering is done in order to determine which product will be the most suitable to recommend to which user.

The recommendations made by the SABRE system can be viewed by various groups, such as a target group/the customer or the analyst group. The target demographic or groups can view the recommendations and based on it, they can choose to either purchase or just review the product. The recommendations made can also be used by a marketing analyst in order to market the product while also used by a product analyst to improve the public reception of the product.

Chapter 4

Implementation

4.1 Collection/ Extraction

Research on various tools/python packages was conducted in order to find a suitable tool in order to extract data from various sources. Some examples for sources of data were amazon website, CNET. The pros and cons of each tool or package were noted down and based on that, we were able to narrow down on a particular tool which would be used for extracting data.

Using the selected tool, we were able to extract the data from the websites and compiled them into a single dataset. An excerpt of the dataset is given below:

	Rating	verified	reviewTime	reviewerID	asin	reviewerName	unixReviewTime	category	tech1	description	...	tech2	brand
0	5	True	02 22, 2015	A38RQFVQ1AKJQQ	4126895493	George Walker	1424563200	['Electronics', 'Headphones', 'Over-Ear Headph...	NaN	['Brand new and High quality! Enjoy your favor...	...	NaN	HeadGear
1	5	True	05 8, 2017	A299MRB9O6GWDE	4126895493	Carolyn B	1494201600	['Electronics', 'Headphones', 'Over-Ear Headph...	NaN	['Brand new and High quality! Enjoy your favor...	...	NaN	HeadGear
2	1	True	11 5, 2016	A3ACFC6DQQLIQT	4126895493	MK	1478304000	['Electronics', 'Headphones', 'Over-Ear Headph...	NaN	['Brand new and High quality! Enjoy your favor...	...	NaN	HeadGear
3	3	True	09 24, 2016	A36BC0YFDBNB5X	4126895493	bigboy	1474675200	['Electronics', 'Headphones', 'Over-Ear Headph...	NaN	['Brand new and High quality! Enjoy your favor...	...	NaN	HeadGear
4	1	True	07 17, 2016	A212PQ0HQPNNWM	4126895493	Kelly Hales	1468713600	['Electronics', 'Headphones', 'Over-Ear Headph...	NaN	['Brand new and High quality! Enjoy your favor...	...	NaN	HeadGear

Figure 4.1: Dataset

4.1.1 Information about the dataset:

The data for our research has been scrapped using web scrapers reviewed by us. We have performed data extraction for the formation of two datasets.

- Headphone dataset
- Laptop dataset

For the headphone dataset we have a total of 31350 tuples whereas for laptop dataset we have

49643 tuples. Our dataset is in the form of user review and user ratings. Where we have further divided are ratings by assigning them a label, namely good and bad. Any review having rating 3 or greater than 3 out of 5 is considered as a good review, otherwise it is considered as a bad review.

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 25188 entries, 0 to 49642
Data columns (total 21 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   Rating          25188 non-null  int64
 1   verified        25188 non-null  bool
 2   reviewerID      25188 non-null  object
 3   asin            25188 non-null  object
 4   reviewerName    25187 non-null  object
 5   unixReviewTime  25188 non-null  int64
 6   category        25188 non-null  object
 7   description     25188 non-null  object
 8   title           25188 non-null  object
 9   also_buy        25188 non-null  object
 10  brand           25188 non-null  object
 11  feature         25188 non-null  object
 12  rank            25188 non-null  object
 13  also_view       25188 non-null  object
 14  main_cat        25188 non-null  object
 15  similar_item    23311 non-null  object
 16  date            24070 non-null  object
 17  price           18017 non-null  object
 18  review_text     25188 non-null  object
 19  time            25188 non-null  datetime64[ns]
 20  rating_class    25188 non-null  object
dtypes: bool(1), datetime64[ns](1), int64(2), object(17)
memory usage: 4.1+ MB
```

Figure 4.2: Dataset metadata

4.2 Exploratory Data Analysis

An exploratory data analysis was conducted by us to know our dataset well. We tried to get a deeper understanding by plotting some visualisations in the form of bar graphs, line graphs, word clouds, and tables.

Below are a few of the screenshots which we would like to highlight

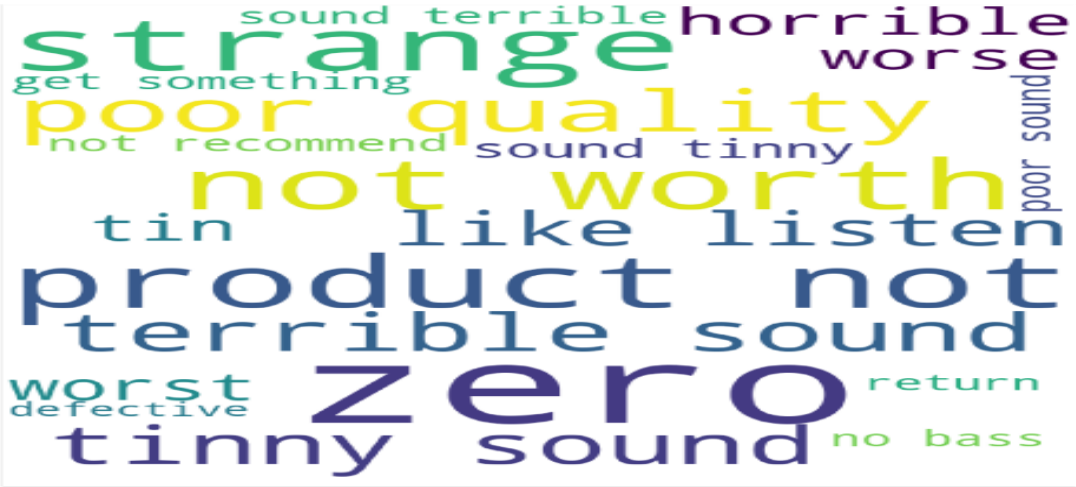


Figure 4.3: ***Word Cloud for headphone dataset***

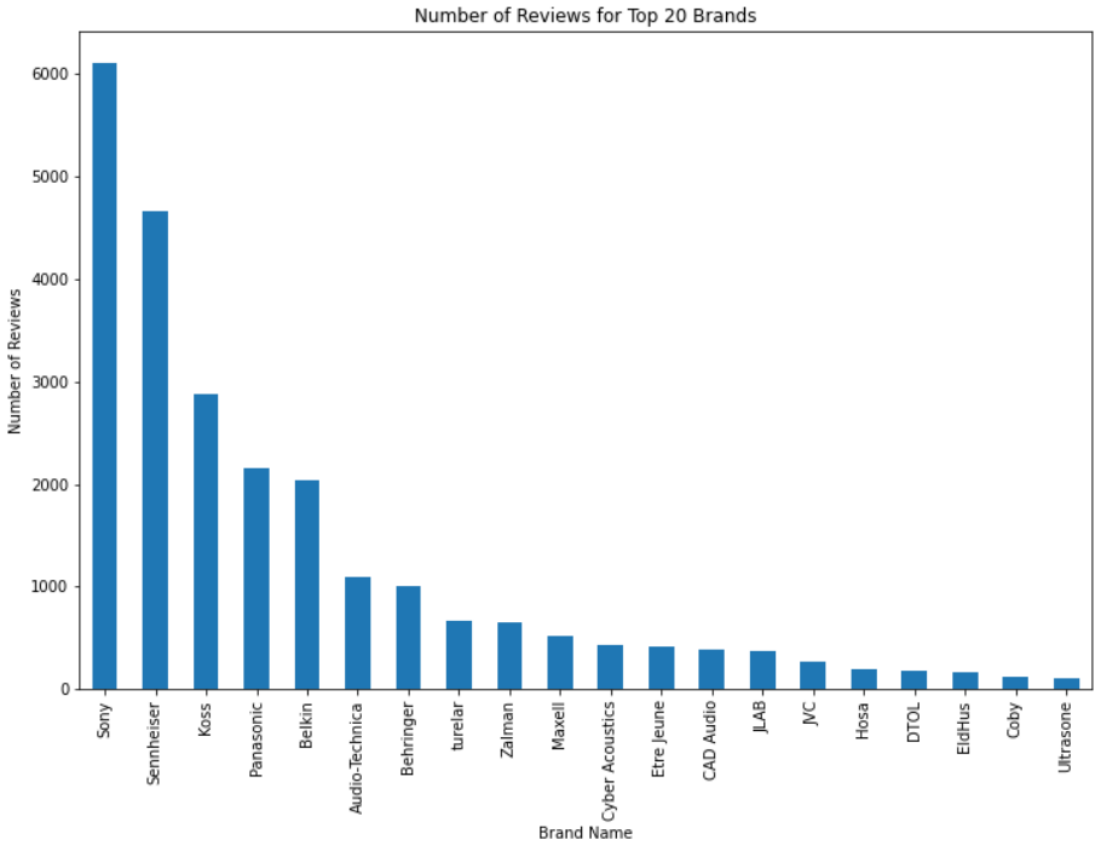


Figure 4.4: ***Number of reviews for top 20 brands***

`

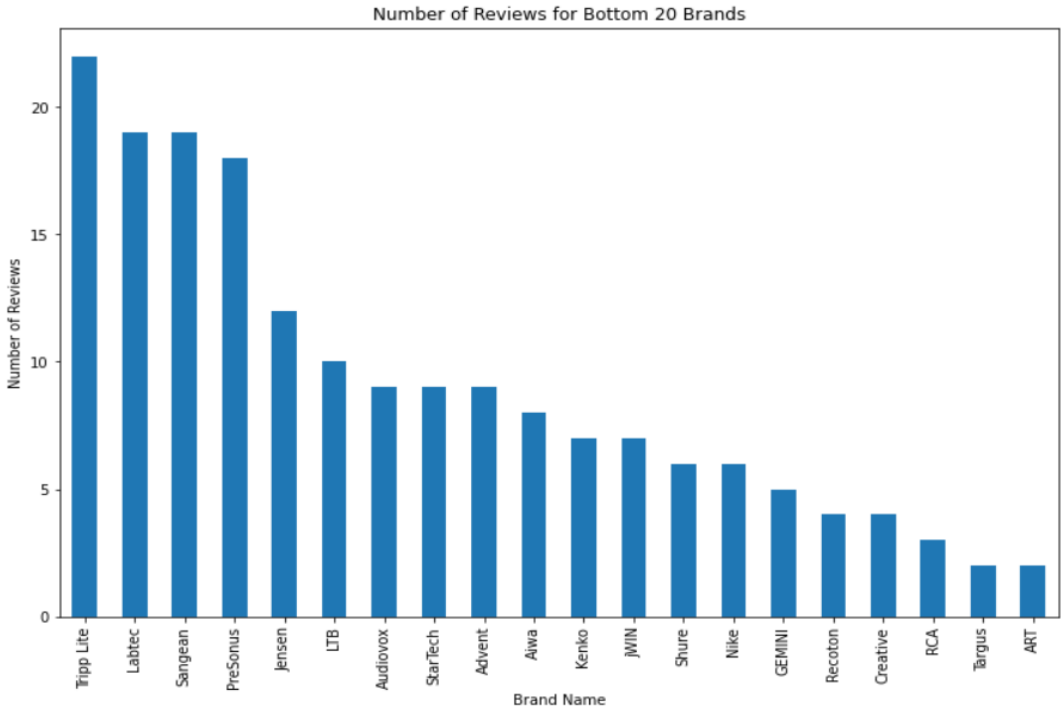


Figure 4.5: ***Number of reviews for bottom 20 Brands***

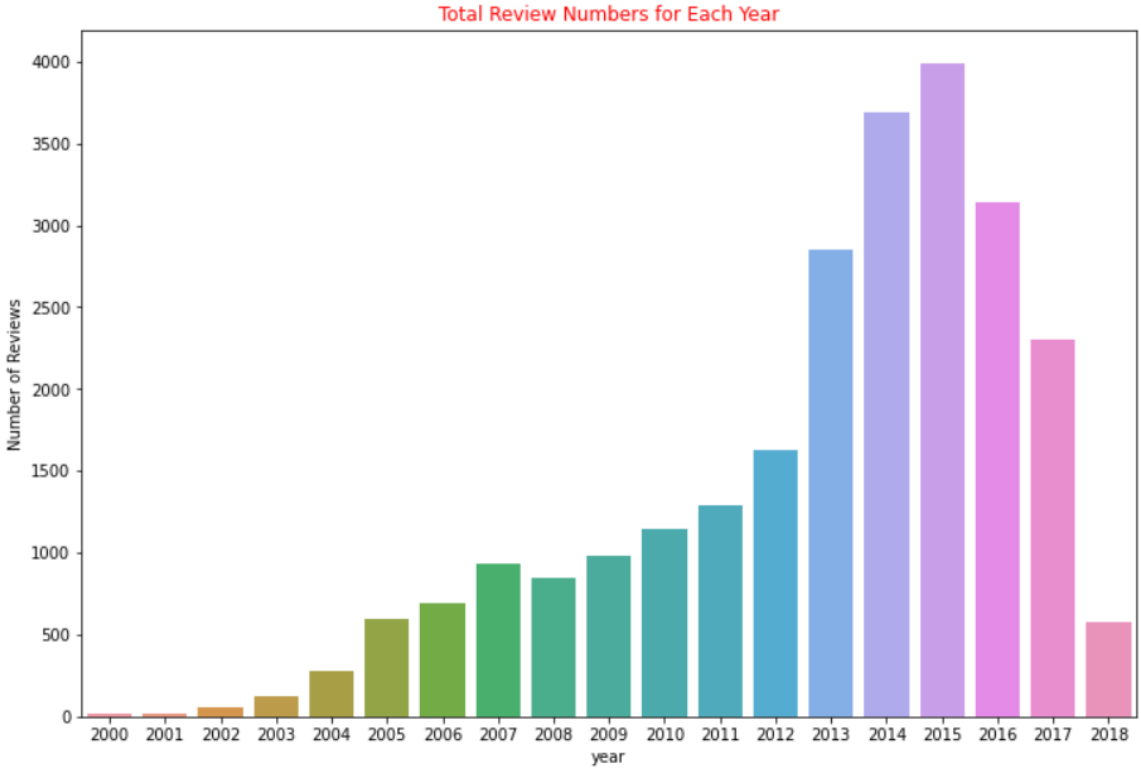


Figure 4.6: Reviews per year

`

4.3 Preprocessing

The data extracted from various sources or websites can contain extra information which are not required for processing or further implementation. Also, the data extracted is not normalized. Therefore, the data will need to undergo normalization and pre-processing in order to remove the unnecessary data and to standardize the data.

- Removing extra text in the helpful column and converting it into integer, since the helpful data was sparse Abs Max Scaler was applied to the column to normalize the data
- Calculating the helpfulness with the help of it,
- Calculating rating class with the help of Rating column

Various pre-processing techniques were applied to the dataset, which are explained below:

Expanding Contractions: Contractions are shortened versions of words or syllables. They exist in either written or spoken forms. Shortened versions of existing words are created by removing specific letters and sounds. In case of English contractions, they are often created by removing one of the vowels from the word. By nature, contractions do pose a problem for NLP and text analytics because, to start with, we have a special apostrophe character in the word. Ideally, we can have a proper mapping for contractions and their corresponding expansions and then use it to expand all the contractions in our text.

Removing Special Characters: One important task in text normalization involves removing unnecessary and special characters. These may be special symbols or even punctuation that occurs in sentences. This step is often performed before or after tokenization. The main reason for doing so is because often punctuation or special characters do not have much significance when we analyze the text and utilize it for extracting features or information based on NLP and ML.

Tokenizing Text: Tokenization can be defined as the process of breaking down or splitting textual data into smaller meaningful components called tokens. Sentence tokenization is the process of splitting a text corpus into sentences that act as the first level of tokens which the corpus is composed of.

This is also known as sentence segmentation, because we try to segment the text into meaningful sentences. Word tokenization is the process of splitting or segmenting sentences into their constituent words. A sentence is a collection of words, and with tokenization we essentially split a sentence into a list of words that can be used to reconstruct the sentence.

Removing Stop words: Stop words are words that have little or no significance. They are usually removed from text during processing so as to retain words having maximum significance and context. Stop words are usually words that end up occurring the most if you aggregated any corpus of text based on singular tokens and checked their frequencies. Words like a, the, me, and so on are stop words.

Correcting Words: One of the main challenges faced in text normalization is the presence of incorrect words in the text. The definition of incorrect here covers words that have spelling mistakes as well as words with several letters repeated that do not contribute much to its overall significance.

Correcting Repeating Characters, Correcting Spellings, Lemmatization: The process of lemmatization is to remove word affixes to get to a base form of the word. The base form is also known as the root word, or the lemma, will always be present in the dictionary.

After performing data pre-processing, we were able to clean 0.56% tokens for headphone dataset and 0.54% token for laptop dataset. Hence a pre-processed dataset was obtained and an excerpt of it is given below

Helpful	rating	id	Review	Helpfullness	rating_class	Clean_text
0.084211	1	5	Great TV,\n I'm going to make this short and ...	helpful	Bad	great tv go make short sweet great tv price go...
0.094737	4	5	Be aware of the Roku subscription needed .,\n ...	helpful	Good	aware roku subscription need great tv function...
0.063158	1	5	TV COST A HIDDEN EXTRA $100 TO CONNECTI,\n Ju...	helpful	Bad	tv cost hide extra connect june receive tvever...
0.068421	1	5	Poor picture,\n Setup went okay. I did have h...	helpful	Bad	poor picture setup go okay search around syste...
0.042105	5	5	Buyer BEWARE OF POSSIBLE SCAM!!,\n Came deliv...	helpful	Good	buyer beware possible scam come deliver great ...
0.026316	5	5	Roku is sooo much easier to navigate than Amaz...	helpful	Good	roku sooo much easier navigate amazon fire sti...
0.026316	4	5	Nope.,\n So, the tv itself is fine. You're l...	helpful	Good	nope tv fine youre look low price tv youll get...
0.026316	5	5	Great value tv, could use a bigger remote with...	helpful	Good	great value tv could use bigger remote number ...

Figure 4.7 : Pre-Processed Data

`

4.4 Feature Extraction

Feature engineering is the process of using domain knowledge of the data to create features that make machine learning algorithms work. Features are usually numeric in nature and can be absolute numeric values or categorical features that can be encoded as binary features for each category in the list using a process called one-hot encoding. The process of extracting and selecting features is both art and science, and this process is called feature extraction or feature engineering. We have performed 3 vectorizers for this namely, TF-IDF, Hash vectorizer and count vectorizer.

4.4.1 TF-IDF

TF-IDF acronym for Term Frequency & Inverse Document Frequency is a powerful feature engineering technique used to identify the important words or more precisely rare words in the text data.

Term Frequency — TF: Term Frequency or the TF is the total count of the unique words available within a document. Mathematically, If we have i is the frequency of terms in the j document then the number of times a word appears in a document divided by the total number of words in the document

Inverse Document Frequency — IDF: Inverse Document Frequency is used to identify the weights of the rare words or important words. In simple terms for example a, an, the are the frequently occurring words due to which the weights of these words are in very high range as compared to the meaningful words like Messi, Sachin, football etc. which is a problem with TF or word count matrix.

4.4.2 Hash Vectorizer

Hashing vectorizer is a vectorizer which uses the hashing trick to find the token string name to feature integer index mapping. Conversion of text documents into matrix is done by this vectorizer where it turns the collection of documents into a sparse matrix which are holding the token occurrence counts

Advantages for hashing vectorizer are:

- As there is no need of storing the vocabulary dictionary in the memory, for large data sets it is very low memory scalable. As there in no state during the fit, it can be used in a streaming or parallel pipeline.
- This class is a low-memory alternative to DictVectorizer and CountVectorizer, intended for large-scale (online) learning and situations where memory is tight, e.g. when running prediction code on embedded device.

`

4.4.3 Count Vectorizer

Count Vectorizer is a great tool provided by the scikit-learn library in Python. It is used to transform a given text into a vector on the basis of the frequency (count) of each word that occurs in the entire text. This is helpful when we have multiple such texts, and we wish to convert each word in each text into vectors (for using in further text analysis). Count Vectorizer creates a matrix in which each unique word is represented by a column of the matrix, and each text sample from the document is a row in the matrix. The value of each cell is nothing but the count of the word in that particular text sample.

4.4 Sentiment Analysis

4.5.1 Classification Models

In this project, the model needs to predict sentiment based on the reviews written by customers who bought headphones from Amazon. This is a supervised binary classification problem. Python's Scikit libraries was used to solve this problem. Following machine learning algorithms were implemented.

1. Logistic Regression: Logistic regression, despite its name, is a linear model for classification rather than regression. Logistic regression is also known in the literature as logit regression, maximum-entropy classification (MaxEnt) or the log-linear classifier. In this model, the probabilities describing the possible outcomes of a single trial are modeled using a logistic function.

2. Naive Bayes: Naive Bayes implements the naive Bayes algorithm for multinomial distributed data, and is one of the two classic naive Bayes variants used in text classification (where the data are typically represented as word vector counts). This algorithm is a special case of the popular naïve Bayes algorithm, which is used specifically for prediction and classification tasks where we have more than two classes.

3.Random Forest Classifier: A random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. The sub-sample size is always the same as the original input sample size but the samples are drawn with replacement if bootstrap=True (default).

4. XGBoost Classifier: XGBoost means eXtreme Gradient Boosting. XGBoost is a decision-tree-based ensemble Machine Learning algorithm that uses a gradient boosting framework. In prediction problems involving unstructured data (images, text, etc.) artificial neural networks tend to outperform all other algorithms or frameworks. However, when it comes to small-to-medium structured/tabular data, decision tree based algorithms are considered best-in-class right now.

5. CatBoost Classifier: CatBoost is an algorithm for gradient boosting on decision trees. "CatBoost" name comes from two words "Category" and "Boosting". The library works well with multiple Categories of data, such as audio, text, image including historical data.

4.5.2 Performance Evaluation Metrics

Precision: Precision is the fraction of relevant (True Positives) instances divided by the total number of relevant instances (true positives + false positives). It talks about how precise your model is, that is out of those predicted positive, how many of them are actual positive. It is computed as

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive}$$

Recall: Recall can also be called sensitivity. It is the fraction of the total amount of relevant instances that were retrieved. It indicates the portion of actual positives that was identified correctly. It can be computed by

$$Precision = \frac{TruePositive}{TruePositive + FalseNegative}$$

F_1 score: In statistical analysis of binary classification, the F-score or F-measure is a measure of a test's accuracy. It is calculated from the precision and recall of the test. The **F_1** score is the harmonic mean of the precision and recall.

4.6 Recommendation System

Till recently, people generally tended to buy products recommended to them by their friends or the people they trust. This used to be the primary method of purchase when there was any doubt about the product. But with the advent of the digital age, that circle has expanded to include online sites that utilize some sort of recommendation engine.

A recommendation engine filters the data using different algorithms and recommends the most relevant items to users. It first captures the past behavior of a customer and based on that, recommends products which the users might be likely to buy.

If we can recommend a few items to a customer based on their needs and interests, it will create a positive impact on the user experience and lead to frequent visits. Hence, businesses nowadays are building smart and intelligent recommendation engines by studying the past behavior of their users. In this project, item-item collaborative filtering was used.

4.6.1 Filtering

This collaborative filtering is useful when the number of users is more than the items being recommended. In this project, the number of users is more than the number of items. In this filtering, the similarity between each item pair was computed and based on that, similar items were recommended which were liked by the users in the past. The weighted sum of ratings of "item-users" were taken.

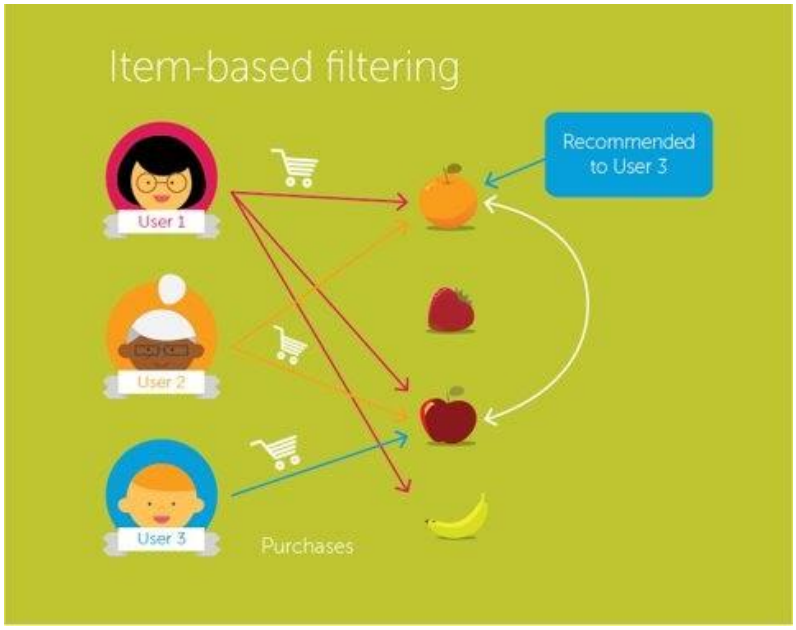


Figure 4.8: Recommendation system framework

	prod_ID	prod_name	ratings_sum
2523	B00004T8R2	Panasonic Headphones On-Ear Lightweight with X...	49
581	B00001P4ZH	Koss Porta Pro On Ear Headphones with Case, Bl...	42
1335	B00001WRSJ	Sony MDRV6 Studio Monitor Headphones with CCAW...	36
7063	B000067RC4	Belkin Speaker and Headphone 3.5 mm AUX Audio ...	31
4897	B00005N9D3	Koss UR-20 Home Headphones	31
6175	B000065BPB	Sennheiser HD280PRO Headphones (old model)	30
9844	B00007EDM8	Sony MDR-J10 H.Ear Headphones with Non-Slip De...	22
8965	B00006JPRQ	Maxell 190317 Lightweight Headphones with Flex...	21
12136	B0001FTVEK	Sennheiser RS120 On-Ear Wireless RF Headphones...	19
11765	B00018MSNI	Sennheiser HD 650 Open Back Professional Headp...	19

Figure 4.9: Ratings sum for products

4.6.2 Cosine Similarity

It is a measure of similarity that can be used to compare documents, give a ranking of documents with respect to a given vector of query words. Let x and y be two vectors for comparison. Using the cosine measure as a similarity function, the formula is

$$sim(x, y) = \frac{x \cdot y}{||x|| ||y||},$$

Where ||x|| is the Euclidean norm of vector x=(x1 ,x2 ,x3…,xp) defined as Similarly, ||y|| is the Euclidean norm of vector y. A result value equal to 0 means two vectors are very different to each other and the value closer to 1 means the vectors are very similar to each other.

4.6.3 User Interface using Flask

Flask, a python module, is a server-side web micro framework, that allows users to develop web applications in a compact and easy to understand way. It's based on the Jinja template engine and the Werkzeug WSGI toolkit. Flask comes with a library of modules and functions with backend systems for

web applications that can be created. User Interface consists of two text fields; one for the user id and one for the product name. The recommendation system will provide top 5 recommended products.

Figure 4.10: Input User Interface

Figure 4.11: Recommended products output

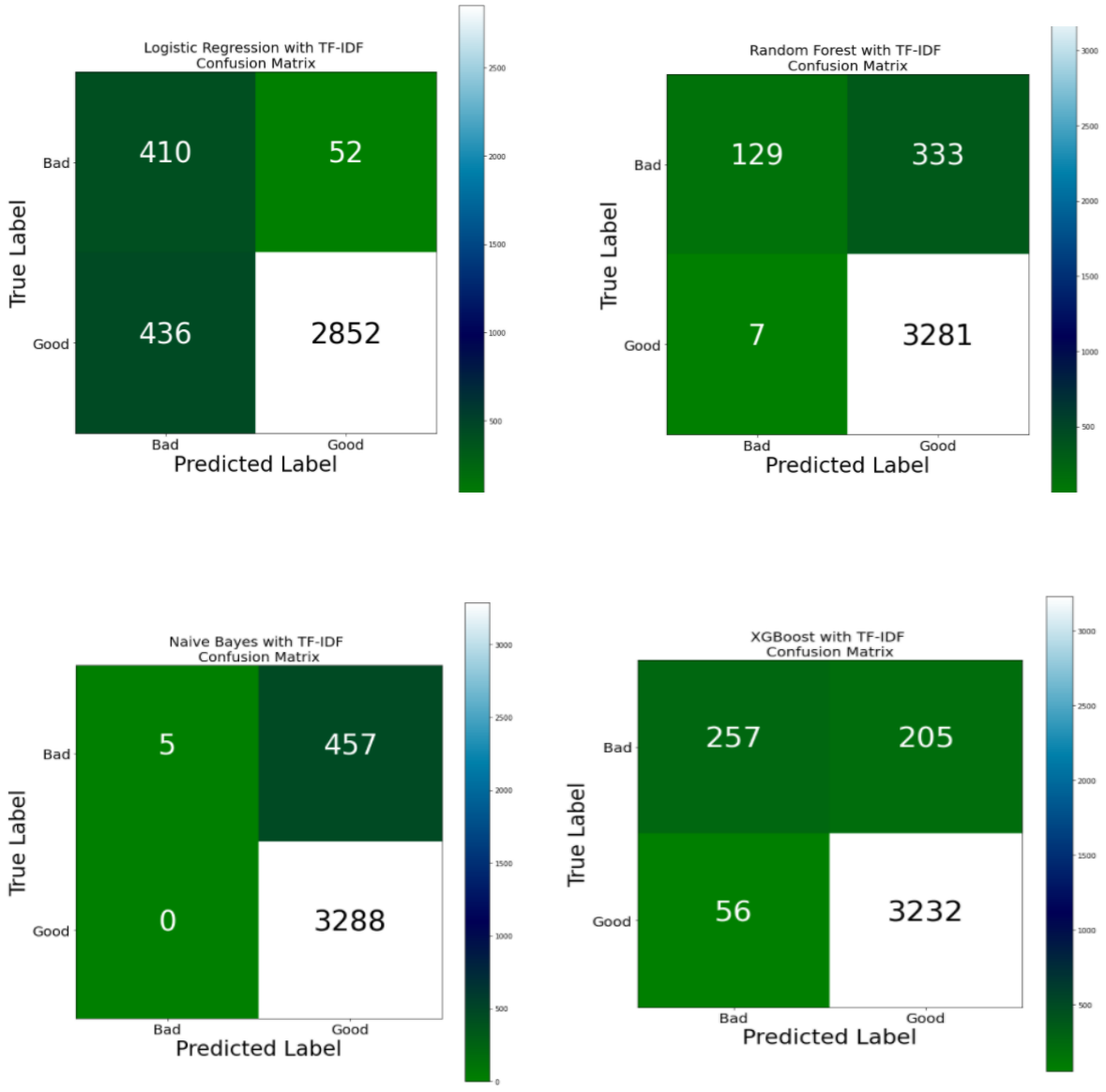
`

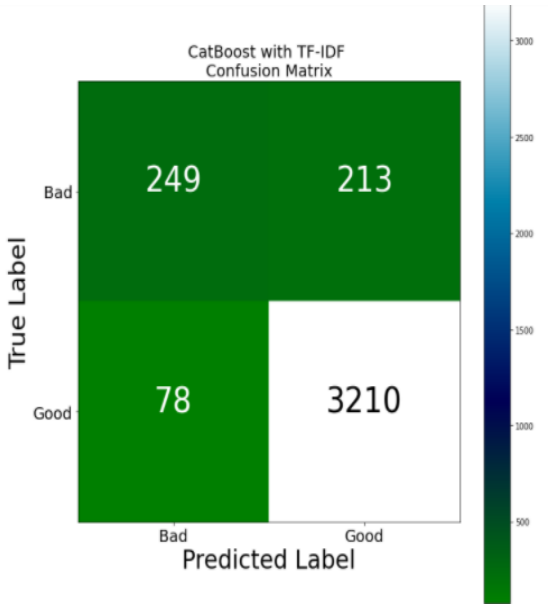
Chapter 5

Results and Discussion

5.1 Result and discussion based on Implementation

1. TF-IDF

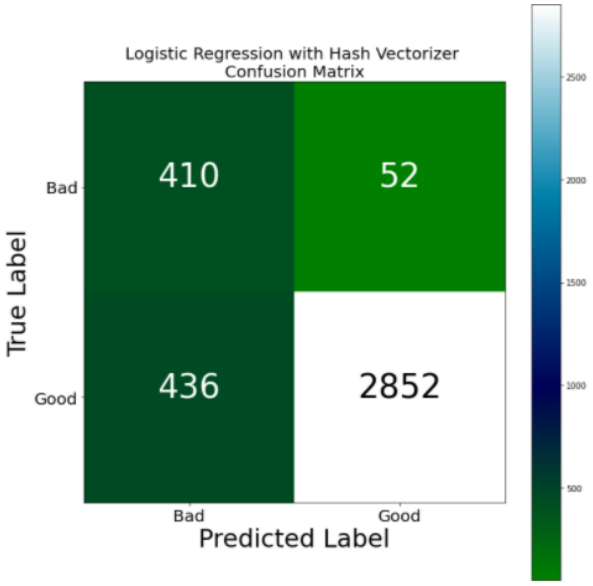


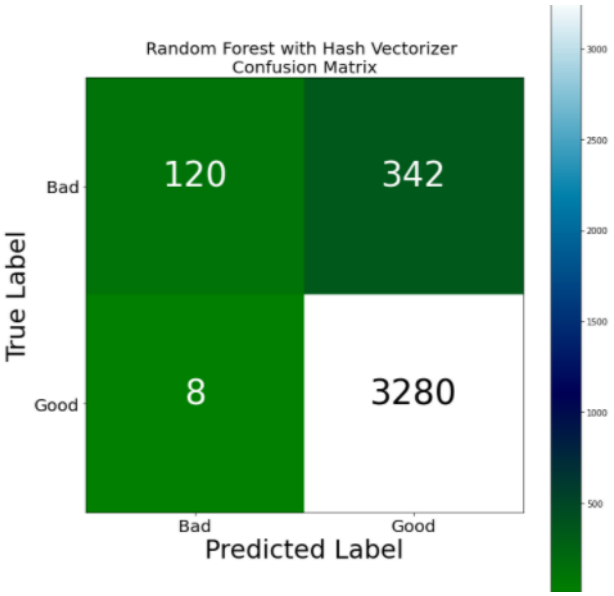


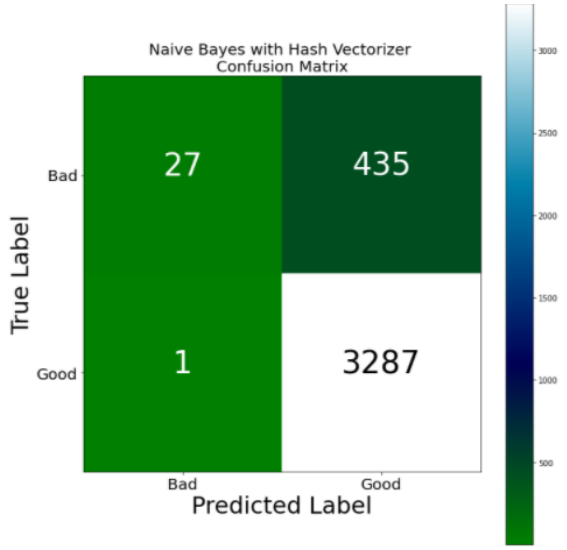
model	accuracy	class	precision	recall	f1-score
LogReg	0.869867	bad	0.484634	0.887446	0.626911
		good	0.982094	0.867397	0.921189
		average	0.920807	0.869867	0.884934
Random Forest	0.909333	bad	0.948529	0.279221	0.431438
		good	0.907858	0.997871	0.950739
		average	0.912869	0.909333	0.886761
Naive Bayes	0.878133	bad	1.000000	0.010823	0.021413
		good	0.877971	1.000000	0.935021
		average	0.893005	0.878133	0.822464
XGBoost	0.930400	bad	0.821088	0.556277	0.663226
		good	0.940355	0.982968	0.961190
		average	0.925661	0.930400	0.924480
CatBoost	0.922400	bad	0.761468	0.538961	0.631179
		good	0.937774	0.976277	0.956638
		average	0.916053	0.922400	0.916542

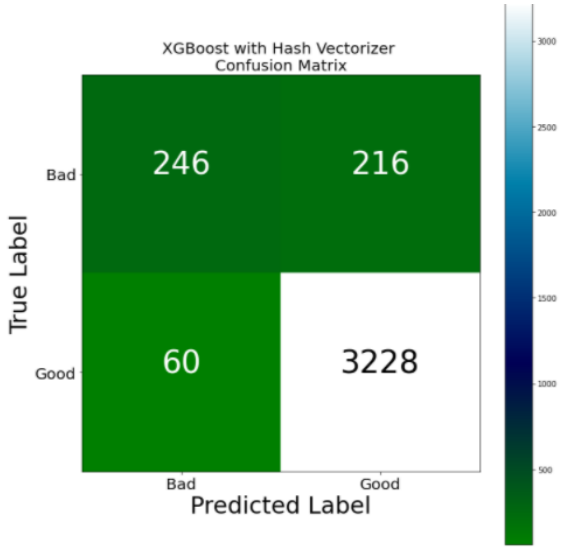
Figure 5.1 : Classification outputs for all models using TF-ID

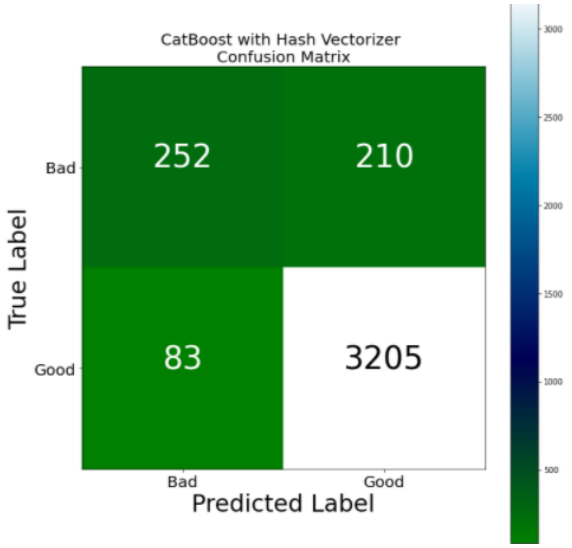
2. Hash Vectorizer







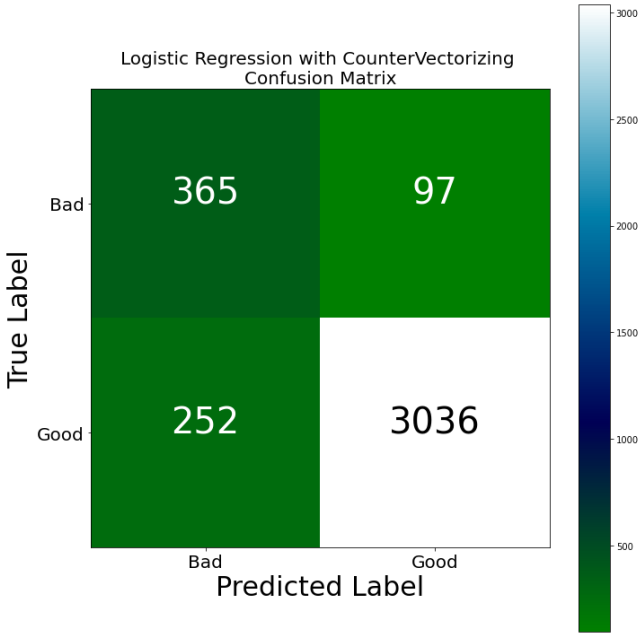


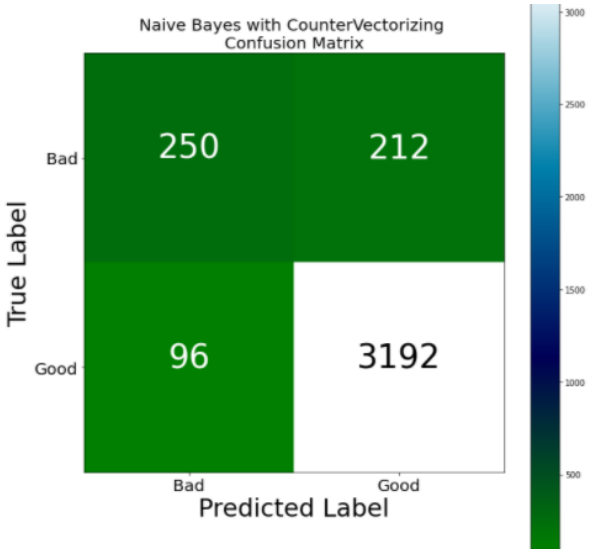


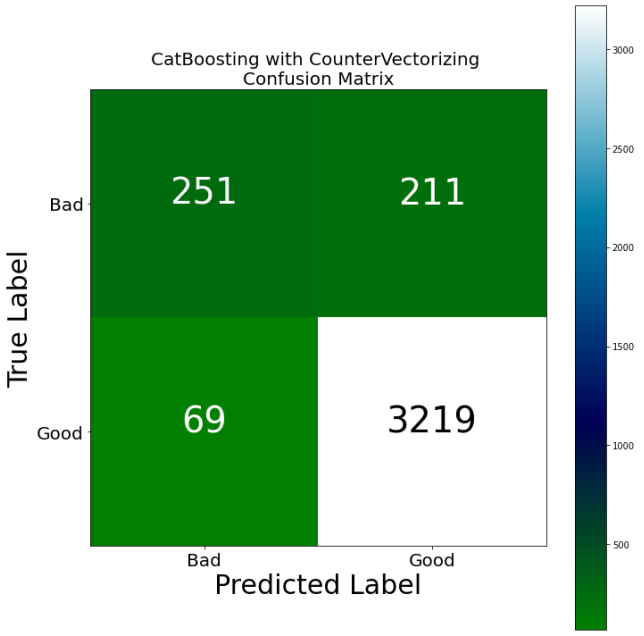
model	accuracy	class	precision	recall	f1-score	support
LogReg	0.850133	bad	0.444196	0.861472	0.586156	462.0
		good	0.977575	0.848540	0.908499	3288.0
		average	0.911883	0.850133	0.868786	3750.0
Random Forest	0.906667	bad	0.937500	0.259740	0.406780	462.0
		good	0.905577	0.997567	0.949349	3288.0
		average	0.909510	0.906667	0.882504	3750.0
Naive Bayes	0.883733	bad	0.964286	0.058442	0.110204	462.0
		good	0.883127	0.999696	0.937803	3288.0
		average	0.893126	0.883733	0.835843	3750.0
XGBoost	0.926400	bad	0.803922	0.532468	0.640625	462.0
		good	0.937282	0.981752	0.959002	3288.0
		average	0.920852	0.926400	0.919778	3750.0
CatBoost	0.921867	bad	0.752239	0.545455	0.632371	462.0
		good	0.938507	0.974757	0.956288	3288.0
		average	0.915558	0.921867	0.916382	3750.0

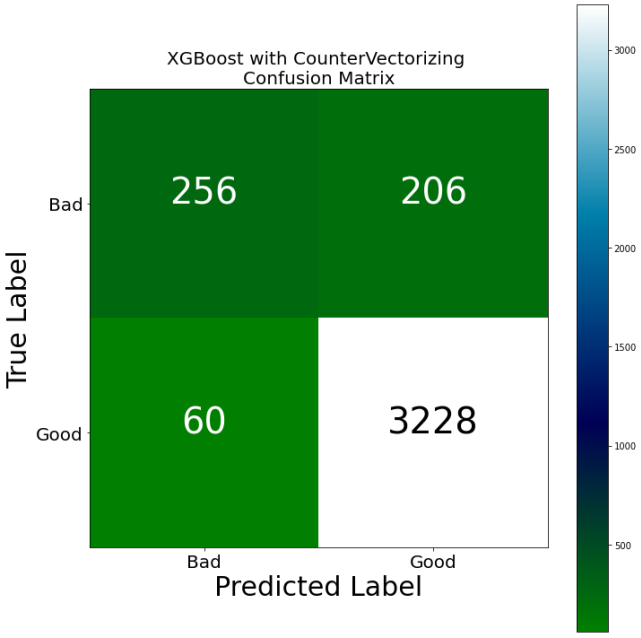
Figure 5.2: Classification outputs for all models using Hash Vectorizer

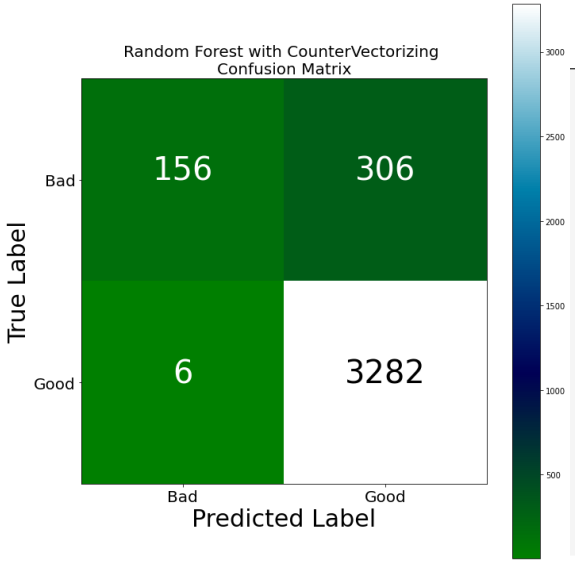
3. Count Vectorizer











vectorizer	model	accuracy	class	precision	recall	f1-score	support
CountVect	LogReg	0.906933	bad	0.591572	0.790043	0.676552	462.0
			good	0.969039	0.923358	0.945647	3288.0
			average	0.922535	0.906933	0.912495	3750.0
	Random Forest	0.916800	bad	0.962963	0.337662	0.500000	462.0
			good	0.914716	0.998175	0.954625	3288.0
			average	0.920660	0.916800	0.898615	3750.0
	Naive Bayes	0.917867	bad	0.722543	0.541126	0.618812	462.0
			good	0.937720	0.970803	0.953975	3288.0
			average	0.911211	0.917867	0.912683	3750.0
	XGBoost	0.929067	bad	0.810127	0.554113	0.658098	462.0
			good	0.940012	0.981752	0.960428	3288.0
			average	0.924010	0.929067	0.923181	3750.0
	CatBoost	0.925333	bad	0.784375	0.543290	0.641944	462.0
			good	0.938484	0.979015	0.958321	3288.0
			average	0.919498	0.925333	0.919343	3750.0

Figure 5.3 : Classification outputs for all models using count vectorizer

Data sparsity is a term used for how much data we have for a particular dimension/ entity of the model. It is calculated as the fraction of the user/item rating matrix that is not empty. We were able to solve the problem of data sparsity by finding the missing values using sentiment analysis. We found that for both datasets used by us Count vectorizer gave the best results. Using it we got an f1 score of 0.90 for random forest and 0.92 for Cat boost on the headphones dataset. Results were similar for laptop dataset. Comparing the results obtained from the classifier we found that Cat boost helps in giving the most accurate results. Using it we got an f1 score of 0.92 on the headphones dataset.

`

Chapter 6

Conclusion and Future Scope

6.1 Conclusion

Data collection is the initial phase of the process. Various web scrapers and data extractors help in effectively collecting data and finally choosing appropriate data for further processing. Various performance evaluation metrics such as RMSE, MAE, precision and recall portray an effective way to compare algorithms and choose the best one. Sentiment analysis uses various ML algorithms and segregates data assigning them a sentiment score. Major obstacle faced in a sentiment analysis-based recommendation system is data sparsity.

We designed a sentiment analysis-based recommendation system by performing data extraction, data preprocessing, sentiment analysis and filtering. The operations were performed for the creation and processing of two datasets used to understand the diversity of the implementation done. It was found that count vectorizer is the best feature extraction algorithm and Catboost is the best classifier in the conditions we faced. Finally, Flask based user interface was implemented to display recommendations.

Results can further be improved by applying deep learning algorithms. Increasing the number of class labels increases the complexity but improvises the results to a very good extent. More number of attributes which can be indirectly found such as time spent by a user using a product, can be further be used. The algorithm can be further applied for recommendation of different types of products. It can also be used to provide recommendation in systems like movie recommendation and flight recommendation system.

6.2 Societal Impact

An unbiased recommendation system is the need of the hour owing to the rapid increase in faux “recommended” electronic products in our social media/search results’ feed which are nothing more than blatant product promotions and advertisements that claim to be personally recommended to the users.

Often, users do not know about the internal components of electronic products which usually

require a thorough amount of research beforehand. There are so many ways for them to get confused between dual-core and quad-core systems, requirement for high RAM or low RAM and so on. At some point all of us have been at the stage of not understanding which product to choose from the plethora of them in the market.

It is also a common occurrence for users to have finalized upon an excellent product on an e-commerce website after a good amount of research but upon checking the user reviews they have been bamboozled by the mixed or negative reception. Sometimes the product will be rated high enough, but it has plenty of bad reviews which places the user in an uncomfortable and confused position; of whether to buy the product and take the risk or abandon the entire search for the best product itself.

This is where the recommendation system comes into play; by predicting and anticipating the users' requirement of a specific product and recommending them those that are appropriate and sit well with their sentiments. Instead of the user being put in a quandary our system would recommend only those products which will have a high chance of being preferred by the user based on social media sentiment as well as their own reviews for previously purchased products.

6.2 Research Impact

There is considerable amount of research being done on sentiment analysis of textual data available as customer reviews on E-commerce websites for consumers and manufacturers alike, for example amazon reviews are available for almost all the products out there, but why stop right there?

In this project we will scrape important data from different product review websites, social media platforms and merge these datasets. It will not only give us a wider set of opinions/reviews as dataset for sentiment analysis, it will in turn increase the accuracy of the general sentiment prediction for a product.

`

References

1. Surya, P. P., & Subbulakshmi, B. (2019, March). Sentimental Analysis using Naive Bayes Classifier. In *2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN)* (pp. 1-5). IEEE.

2. Schouten, K., & Frasincar, F. (2015). Survey on aspect-level sentiment analysis. IEEE Transactions on Knowledge and Data Engineering, 28(3), 813-830.

3. Lei, C., Liu, D., & Li, W. (2015). Social diffusion analysis with common-interest model for image annotation. IEEE Transactions on Multimedia, 18(4), 687-701.

4. Gomathi, R. M., Ajitha, P., Krishna, G. H. S., & Pranay, I. H. (2019, February). Restaurant Recommendation System for User Preference and Services Based on Rating and Amenities. In *2019 International Conference on Computational Intelligence in Data Science (ICCIDS)* (pp. 1-6). IEEE.

5. Venil, P., Vinodhini, G., & Suban, R. (2019, March). Performance evaluation of ensemble based collaborative filtering recommender system. In *2019 IEEE International Conference on System, Computation, Automation and Networking (ICSCAN)* (pp. 1-5). IEEE.

6. Pradhan, R., Khandelwal, V., Chaturvedi, A., & Sharma, D. K. (2020, February). Recommendation System using Lexicon Based Sentimental Analysis with collaborative filtering. In *2020 International Conference on Power Electronics & IoT Applications in Renewable Energy and its Control (PARC)* (pp. 129-132). IEEE.

7. Sharma, S., Sharma, A., Sharma, Y., & Bhatia, M. (2016, April). Recommender system using hybrid approach. In *2016 International Conference on Computing, Communication and Automation (ICCCA)* (pp. 219-223). IEEE.

8. Hassan, A. K. A., & Abdulwahhab, A. B. A. (2017). Reviews Sentiment analysis for collaborative recommender system. Kurdistan journal of applied research, 2(3), 87-91.

9. Papageorgiou, A., Zahn, M., & Kovacs, E. (2014, June). Auto-configuration system and algorithms for big data-enabled internet-of-things platforms. In *2014 IEEE International Congress on Big Data* (pp. 490-497). IEEE.

`

10. Thomas, D. M., & Mathur, S. (2019, June). Data analysis by web scraping using python. In *2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA)* (pp. 450-454). IEEE.

11.Parvez, M. S., Tasneem, K. S. A., Rajendra, S. S., & Bodke, K. R. (2018, January). Analysis of different web data extraction techniques. In *2018 International Conference on Smart City and Emerging Technology (ICSCET)* (pp. 1-7). IEEE.

12. Osman, N. A., & Noah, S. A. M. (2018, March). Sentiment-based model for recommender systems. In *2018 Fourth International Conference on Information Retrieval and Knowledge Management (CAMP)* (pp. 1-6). IEEE.

13. Hu, S., Kumar, A., Al-Turjman, F., Gupta, S., & Seth, S. (2020). Reviewer credibility and sentiment analysis based user profile modelling for online product recommendation. *IEEE Access*, *8*, 26172-26189.

14. Singla, Z., Randhawa, S., & Jain, S. (2017, July). Statistical and sentiment analysis of consumer product reviews. In *2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT)* (pp. 1-6). IEEE.

15. Goel, S., Banthia, M., & Sinha, A. (2018, August). Modeling recommendation system for real time analysis of social media dynamics. In *2018 Eleventh International Conference on Contemporary Computing (IC3)* (pp. 1-5). IEEE.

16. Ziani, A., Azizi, N., Schwab, D., Aldwairi, M., Chekkai, N., Zenakhra, D., & Cheriguene, S. (2017, December). Recommender system through sentiment analysis. In *2nd International Conference on Automatic Control, Telecommunications and Signals*.

17. Ren, Q., Zheng, Y., Guo, G., & Hu, Y. (2019, February). Resource Recommendation Algorithm Based on Text Semantics and Sentiment Analysis. In *2019 Third IEEE International Conference on Robotic Computing (IRC)* (pp. 363-368). IEEE.

`

18. Osman, N. A., Noah, S. A. M., & Darwich, M. (2019). Contextual sentiment based recommender system to provide recommendation in the electronic products domain. *International Journal of Machine Learning and Computing*, *9*(4), 425-431.

19. Barley, N. S., & Keole, R. R. A Review of Recommendation System in Domain Sensitive Manner.

20. Priyadharsini, R. L., & Felciah, M. L. P. (2017). Recommendation system in e-commerce using sentiment analysis. *Int. J. Eng. Trends Technol.(IJETT)*, *49*(7).

21. Fang, X., & Zhan, J. (2015). Sentiment analysis using product review data. *Journal of Big Data*, *2*(1), 1-14.

`

Publications

Nair A, Paralkar C., Pandya J, Chopra Y .et al. 'Comparative Review on Sentiment Analysis-Based Recommendation System'. Accepted for Publication for IEEE Sponsored 6th International Conference for Convergence in Technology (I2CT) 2021. **In Press: 978-1-7281-8876-8/21.** Link to the same is here.

`

Acknowledgement

We are using this opportunity to express our gratitude to everyone who supported us throughout this research. We are thankful for their inspiring guidance, constructive criticism and friendly advice during the research work. We are sincerely grateful to them for sharing their truthful and illuminating views on a number of issues related to the project. We are also thankful to Prof. Deepa Krishnan and management of NMIMS's Mukesh Patel School of Technology, Management and Engineering for their support and encouragement. We are highly indebted to Dr. Shubha Puthran for her guidance and constant supervision as well as for providing necessary information regarding the research & also for her support in completing the project on research.