SENTIMENT ANALYSIS FOR E-BOOK REVIEWS ON AMAZON TO DETERMINE E-BOOK IMPACT RANK

Thesis

Submitted to

The College of Arts and Sciences of the

UNIVERSITY OF DAYTON

In Partial Fulfillment of the Requirements for

The Degree of

Master of Computer Science

By

Afnan Abdulrahman A Alsehaimi

Dayton, Ohio

May 2021

SENTIMENT ANALYSIS FOR E-BOOK REVIEWS ON AMAZON TO DETERMINE E-BOOK IMPACT RANK

Name: Alsehaimi, Afnan Abdulrahman A

APPROVED BY:

James P. Buckley, Ph.D.
Committee Chair

Saeedeh Shekarpour, Ph.D.
Committee Member

Tam V. Nguyen, Ph.D.
Committee Member

ABSTRACT

SENTIMENT ANALYSIS FOR E-BOOK REVIEWS ON AMAZON TO DETERMINE E-BOOK IMPACT RANK

Name: Alsehaimi, Afnan Abdulrahman A
University of Dayton

Advisor: Dr. James P. Buckley

User-generated content platforms have changed the dynamics of the business environment and redefined how organizations and governments communicate with the public. Further, such platforms act as the primary means to measure customer satisfaction. Thus, those organizations need to analyze the content generated by their customer to extract their opinions then decide based on trustable information. Also, knowing user behavior and perception for a specific product is useful to customers in the decision-making process. In this thesis, a comparative study has been conducted to develop a model to measure customer satisfaction on Amazon e-book products by applying natural language processing (NLP), machine learning, deep learning, and text mining techniques on costumers reviews. This thesis will study the possibility of generating a rating based on sentiment analysis of each product instead of rating-based stars, which is already applied to the Amazon e-book rating system.

Dedicated to my family

ACKNOWLEDGMENTS

First, I would like to thank my advisor Dr. James Buckley, who gave me much excellent knowledge and consideration while working on this thesis. Thanks again Dr. James Buckley for providing me with all the necessary facilities to complete this project at a high level.

Also, I am highly obliged in taking this opportunity to sincerely thank Dr. Saeedeh Shekarpour and Dr. Tam Nguyen, for serving on my thesis committee, taking time out of their busy schedules.

I am incredibly thankful to all professors in the Department of Computer Science and Intensive English Program (IEP), the University of Dayton, for their efforts and support many times.

Finally, my most tremendous gratitude goes to my own family, who have helped in so many ways, and they are beside me every time.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

NLP	Natural Language Processing
E-book	An electronic book
ML	Machine Learning
HMM	Hidden Markov Model
CRF	Conditional Random Field
PMI	Pointwise Mutual Information
NLTK	Natural Language Toolkit
ANN	Artificial Neural Networks
SVM	Support Vector Machines
AdaBoost	Adaptive boosting
Kneighbors	K- nearest neighbors
SGD	Stochastic Gradient Descent
TF-IDF	Term Frequency–Inverse Document Frequency
PTA	Percentage of Tick Accuracy

CHAPTER 1

INTRODUCTION

1.1 Background

To execute business operations successfully depends on meeting customer demands. Customer satisfaction plays a crucial role in maintaining brand value and plays an essential role in the company's sustainability [1]. The companies spend much to conduct market analysis to determine customer preferences, behavior, and needs. Hence, to build brand and position in the market, it is essential to analyze the customer response about products they purchase. The brand name and value influence the customer decision, which directly results in its sustainability [2].

The rapid development in digital technology constructed a global world through online platforms. The online platforms that are accessible via the internet generate a massive amount of information daily. This information presents customer opinions towards a particular product, messages, images, etc., and develop opportunities to explore, analyze and use this vast information (BigData) for the decision-making process. It has been proven that online comments are useful to express customer insights, and it has the potential to determine customer demands [3][4]. Therefore, discovering knowledge from user-generated content in terms of sentiment is useful in the decision-making process. In [5], it is argued that opinion mining assists companies in determining customer needs and helps to rank and advertise products.

Also, the term Artificial Intelligence (AI) first appeared in 1956 [6]. However, artificial intelligence is more common today because we have big data that is difficult to analyze and understand. Artificial intelligence (AI) enables machines to learn from experience, adapt to new inputs, and perform a human-like task [7]. Most examples of artificial intelligence rely heavily on machine learning and natural language processing. Computers can be trained to accomplish many tasks by processing large amounts of data and recognizing patterns in the data. For example, we have significant online review data to do sentiment analysis through natural language processing (NLP). Machine learning has been able to demonstrate artificial intelligence through natural language processing. Natural Language Processing (NLP) deals with computational algorithms' construction to automatically analyze and represent human language [8]. NLP uses machine learning models that help him with Sentiment analysis. Sentiment analysis is detecting positive, negative, or neutral emotions in text [9]. Natural language processing and machine learning are not recent fields. However, the convergence between the two areas is contemporary, and it achieves more progress.

In this perspective, Natural Language Processing (NLP) strategies facilitate user-generated content and determine customer sentiment or opinion towards the product [15]. The NLP techniques enable extracting relevant information from unstructured forms of data. Usually, NLP involves sentiment analysis on the following levels: 1) document, 2) sentence and 3) aspect [14]. This thesis aims to analyze information over the document and sentence levels to determine which level is accurate with user-generated content, considered informal text. Further, this thesis will study the impact of several text mining techniques on sentiment analysis with informal text.

1.2 Problem Statement

Currently, due to the preventive precautions to combat the ongoing global pandemic of Coronavirus disease, it has been observed that many aspects of life are going online [10]. For example, many countries are adopting distance learning around the world. As a result, individuals need e-books that allow distance learning in emergencies, such as when colleges, classrooms, educational centers, and libraries are temporarily closed during the confrontation with the Coronavirus [11]. As a result, people and organizations have been trying to replace paper sources with e-sources, and e-book have become the preferred option. That means e-commerce increased in general, particularly in e-books [10] [12]. Recently, a large scale of data is generated daily, such as online reviews over e-commerce networks, which is crucial to analyze and determine product or services impact and user behavior. This massive amount of information enables business owners to make valuable decisions to compete in the market. Numerous studies have indicated online e-book reviews on purchase decisions and varying user perceptions towards a specific product. The user-generated content is considered a vital source to build a sustainable competitive market environment. T

This thesis's primary goal is to analyze online e-book reviews to compute product scores based on sentiment analysis (positive, negative, natural). We focus on adopting sentiment analysis techniques on e-book online reviews to aggregate product impact. The proposed approach first applies machine learning and deep learning algorithms to classify each review or sentence as positive, negative, or natural. Secondly, we calculate the product rank based on the sentiment extracted from the related reviews to each product.

Third, we compare the extracted ranks for each product to the overall score (calculate by Amazon) to study if the overall score reflects the content of user's reviews.

1.3 Aim and Objectives

The user-generated content in online reviews or comments leads to significant advantages for customers and companies. The vast range of reviews presents diverse opinions from hundreds to thousands, making it complicated in terms of computation resources to process and extract information from BigData.

This thesis presents a comparison study between many text mining and machine learning techniques to apply sentiment analysis on product reviews where features are extracted from each review. The proposed approach is divided into five phases:

1) Prepare data

2) Pre-processing

3) Features Extraction

4) Model Building

5) Evaluation

The evaluation results will be presented in several forms of measures such as accuracy, recall, precision, and F1.

1.4 Contribution

The following are the main contribution of the thesis:

1. Utilize two different types of sentiment analysis levels (document, sentence levels)

2. Present a comparison study covering several machine learning and deep learning algorithms including studying the impact of several text mining techniques on sentiment analysis.
3. Compute product impact score from identified sentiment to assist customers and businesses in knowing user behavior and perception for a specific product.
4. Compare the sentiment analysis result and overall score to study if the overall score reflects user reviews' content.

1.5 Thesis Structure

The thesis is organized as follows: Section 1 presents the introduction. Section 2 presents literature reviews that have been conducted in the past. Section 3 presents the methodology in detail, including experiment design, system components and phases, the selected dataset to build the solution, and the algorithms used. Section 4 presents the results with discussion. Section 5 presents the conclusion and future work.

CHAPTER 2

LITERATURE REVIEWS

In this thesis, we use sentiment analysis, which is necessary to discover knowledge from user-generated content in rating e-book products. This will be an efficient method to present a ranking system based on the sentiment analysis for rating e-book products. For example, our approach will be able to rank products that do not have a star system. Besides, we can see if the overall score for product stars reflects user reviews' content if there are star systems. For that, we will have two sections for our related fields. 1. Studies on sentiment analysis, and 2. Studies on ranking products.

2.1. Studies on sentiment analysis

Sentiment Polarity has been the primary issue in sentiment classification [16,17,18,19]. Classification of a given sequence of words into positive or negative sentiment is called polarity.

2.1.1 Studies based on sentiment analysis levels:

Generally, sentiment polarity is categorized into three levels as follows: i) document level, ii) sentence level, and iii) aspect level [14].

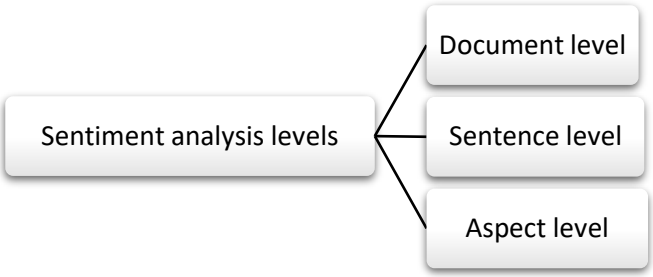

Figure 1: Sentiment analysis levels

The document level refers to determining opinion polarity, while sentence level refers to determining the given sequence of words. The aspect level deals with what people think about different aspects such as battery life, price, service, etc. A rule-based semantic analysis approach was used and compared between the proposed rule-based approach and three learning-based approaches. The rule-based approach showed good performance in the dataset. However, this only focuses only on sentiment analysis based on sentence level [20]. In [21], they focused on extracting sentiments associated with a phrase or sentence. Also, they demonstrate how the atomic sentiments of individual phrases merge in the presence of conjuncts to assess a sentence's ultimate sentiment. They used word dependencies and dependency trees to analyze the sentence constructs. Some of the research that has been done at the sentence level is [22, 23]. In 2013, Moraes, R., Valiati, J. F., & Neto, W. P. G used document-level sentiment classification for expressing positive or negative sentiment. Also, they used supervised methods consisting of two stages. The first stage is the extraction/selection of informative features, and the second stage is a classification of reviews using learning models. They used (ANN) and (SVM). ANN has produced the superior result of SVM's e4result in their research [24]. In 2017, Dou capturing user and product information for document-level sentiment analysis with a deep memory network. He used Yelp datasets and focused on the influence of users who express the sentiment and products which are evaluated [25]. Some of the research that has been done at the document level is [26,27,29]. In 2004, Hu et al. presented the concept of aspect term extraction from user content and to extract aspect adopted rule-based and statistical approaches [11]. Later, [12] [13] improved the Hu et al. approach where noun phrases are considered product features and a PMI score computer

between product class and extracted noun phrase to determine aspect as product features. Scaffidi et al. proposed a linguistic model to determine aspects. They assumed that similar aspects are frequently used in reviews, but the presented model was not precise due to noise in aspects extraction. In [17], they presented CRF based approach to extract aspects across multiple domains. Some of the considerable research that has been done in the context aspect extraction [14] [18] [19].

In this thesis, we study 2 sentiment analysis levels: document and sentence level.

2.1.2 Studies based on sentiment analysis techniques

The techniques for sentiment analysis can be divided into two groups, 1) methods for sentiment analysis based on machine learning [37, 39, 40] and methods for lexicon-based sentiment analysis [20, 21, 38]. In this thesis, we focus on the techniques based on supervised machine learning that are used; a series of labeled training samples are needed by training various kinds of supervised machine learning algorithms using the samples. In 2015, there is a study about Chinese comments sentiment classification based on word2vec and SVMperf. They focus on classifying the comments into two classes (positive and negative) according to the sentiment's polarity. Much of the existing research is centered on the Extraction of lexical features and syntactic features, while the semantic relationships between words are ignored. They used a machine learning-based method for sentiment classification because of its outstanding performance [41]. In [42] another paper about sentiment analysis model based on supervised learning they would use a machine learning approach using unigram feature with two types of information (frequency and TF-IDF) to realize polarity classification of documents. They compared

two NLP techniques and one machine learning algorithm, which was SVM. As a result, the information of TF-IDF is more effective than frequency.

This thesis is different from previous approaches that have mostly relied on comparing the machine learning algorithms or deep learning algorithms separately since we compare several machine learning and deep learning algorithms. Further, this thesis studies the impact of several text mining techniques on sentiment analysis.

2.2. Studies on ranking products.

Most research in ranking products is focused on Recommendation systems and ranking alternative products [33, 34, 35,36]. However, we focus intensely on analyzing each product individually to help the customer know about each product's quality without effect from other resources and help the developers make positive changes to their product. Guo, Du, and Kou (2018) have a paper that proposes a ranking method for online reviews based on different aspects of the alternative products, which combines both objective and subjective sentiment values [30]. Ghose and Ipeirotis (August 2007) propose two ranking mechanisms for ranking product reviews: a consumer-oriented ranking mechanism and a manufacturer-oriented ranking mechanism. The first one ranks the reviews according to their expected helpfulness. However, the second one ranks the reviews according to their desired effect on sales [31]. Their approach is dependent on the dependent variable HELPFUL, which is the log of the ratio of helpful votes to total votes received for a review. This was a great study, but there is a disadvantage: their approach does not work with reviews that do not contain variable HELPFUL. However, our research depends on sentiment analysis for the text review and is independent of any variable, making it work efficiently with any textual review. This research conducts a

comparative study to cover several machine learning, text mining, and NLP techniques that can help us build a product ranking system based on customer text reviews. This study will fill the literature gap shown above which none of the related works aimed to study the impact of different data mining techniques on the building of a ranking system based on text reviews. Further, our study will compare the developed ranking system and the traditional ranking system used by Amazon.

CHAPTER 3

METHODOLOGY

Large volumes of data are being generated daily through social media and digital content. If utilized intelligently and employed for leveraging better insights, these data provide better revenue while giving the users a better experience. For this, we need a proper way to leverage knowledge from that data and perform significant statistical inferencing. Most of this information is unstructured and available in the form of text. Hence, to leverage and analyze the data, we need Natural Language Processing techniques (NLP). Traditionally, extracting information from text data has been done through statistical modeling and traditional machine learning algorithms. In recent times, there has been a rise of usage of Deep Neural Networks (DNN) in the field of Natural Language Processing (NLP) with the advent of state-of-the-art technologies, like transformers, Long Short-Term Memory Network (LSTM), and through the usage of models like BERT, GPT-3, etc. There have been various techniques and tricks that have been utilized to push the accuracy of NLP algorithms further. These include hierarchical techniques like document, sentence, and word level analysis, n-grams, long and short-range interactions, special filtering techniques, etc. Here, in this research work, we have tried to deal with a specific part of the field of Natural Language Processing (NLP), dealing with information extraction from reviews and comments and determining their sentiment. In short, we have dealt with sentiment analysis in the context of the review rating system. We have tried to provide and design a new and improved sentiment-based rating system as a better substitute to the already in use star-based rating system as our

approach provides better insights into the ratings of a product compared to that of the traditional one. We have tried to use various tricks and techniques to leverage the most from our statistical modeling line and tried to infer which work the best in a generalized scenario upon comparison using various performance measuring metrics, like precision, recall, and f1 score. We have tended to minimize sample bias and other sources of errors in our inferencing.

3.1 Experiments Design

We have designed our experiment pipeline to be extremely robust and free of biases as much as possible. We first have collected our dataset, pre-processed it, and then used appropriate techniques and modeling strategies to derive our experimental analysis. We have then gone onto test our performances of our developed models and hence, determined the efficacies of our design process. We first need to collect a relevant dataset that can be employed in our use case of generating a rating based on sentiment analysis of the product. We have tended to leverage platforms that have content generated by users, which can be leveraged to change the environment circumscribing the dynamics related to a business. This has redefined and reshaped the pathway that is utilized traditionally by business organizations and government as an interface for communication with the public. These platforms thus also act as the primary means to the task of measuring customer satisfaction. In this regard, it is vital for such organizations that tend to leverage such platforms to analyze the content being generated by the public. This can be utilized to extract customer opinion upon which a well-informed decision can be vested and thereby grounded on information derived from such sources. In the scope of this research, we have conducted a comparative study to develop and derive a model to measure the

satisfaction of the customers on E-book products of Amazon. The study has been made with the application of Natural Language Processing (NLP), text mining techniques, machine learning, and deep learning on reviews received from customers for these products. The research has also explored the possibility of generating a more robust and insightful rating system based on sentiment analytics of each of the products in the question of our use case.

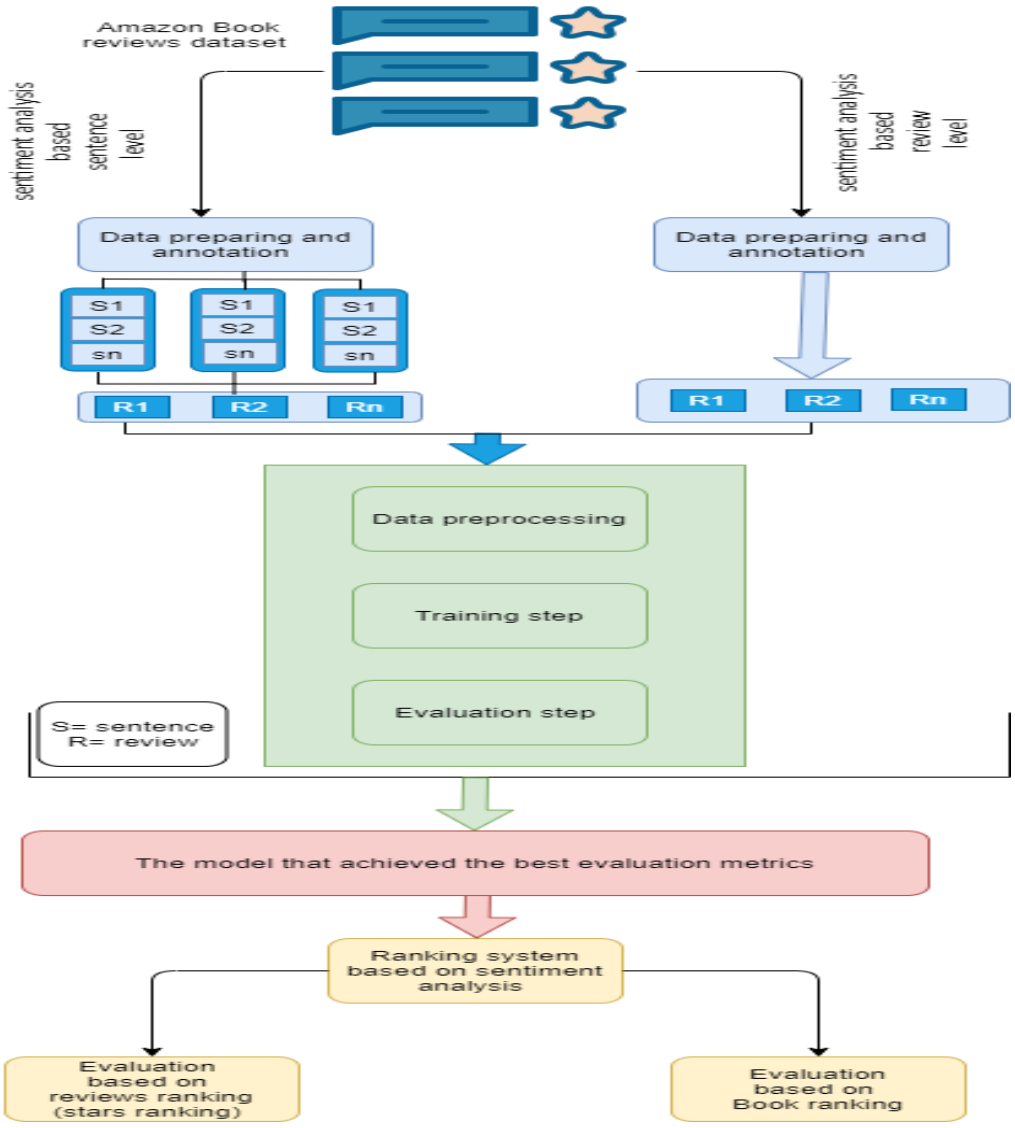

Figure 2:Research methodology

The proposed rating system will serve as a better replacement for the existing rating system based on stars. This research work and its experiments is comprised of four major components. Figure 2 presents our pipeline is based on the latest machine learning pipeline which uses: -

1. The collection and preparation of the data
2. The pre-processing step
3. Training the model
4. Evaluation of the results

Hence, it is focused on developing our desired rating system. The pipeline also includes the testing of our developed sentiment method to justify its usage and establish its prevalence over existing systems.

The experiment design has been done keeping in mind the end goal of our research. This has been done in the following way: -

1. We have experimented with various algorithms and their variants to arrive at the proper algorithm that meets our goal.
2. This exploration of various algorithms and their numerous better variants, done on our dataset, guarantees the usage of the most proper and optimized technique in our use case.
3. The solution has also been refined with techniques of hyperparameter tuning, thus improvising our basic solution which is feasible for our scenario.

The system and our methodology have been divided into two lines of analysis techniques used in natural language processing (NLP) as follows:

1. The first method comprises of all analysis being done on the document-level
2. The second method comprises of sentence-level granularity of analysis

This has been done to identify which aspects of the review affect the analysis of the sentiment of the review. Upon conducting the experiments, the results indicated that document-level granularity provides better prediction accuracy for sentiment analysis. This is in line with the current research findings. And hence, it also fortifies our results obtained.

In the coming passages, we discuss in detail every aspect of our experiment. This can be established well from our research technique that we can minimize the bias errors and variance errors while maximizing our model accuracies using state-of-the-art techniques. Thus, we can conclude that our experiment design is robust enough to be able to give us distinguished and viable research findings on which we can then build our well established and efficient sentiment-based rating system that can provide deeper insights into the rating of perceiving by the customers. This will lead to a better rating system.

3.1.1 Data Descriptions

Considering our research work, we have first strived to complete the collection of appropriate data that can be utilized to meet our end goal. The benchmark dataset utilized in this thesis has been downloaded from Kaggle, which provides open-source datasets [28]. We have selected the Amazon e-books to review the dataset for use in training our model and subsequent evaluation of them to achieve our desired rating system development. The dataset contains the following attributes:

Table 1: Dataset Attribute

#	Attribute	Type	Summary
1	ID of the review	Decimal	This attribute presents the review identifier.

2	ID of the product	String	This attribute presents the unique identifier for product in alphanumeric format.
3	Helpful	Array	This attribute presents array where lower bound = 0, upper bound = 5 that shows how much the product was helpful.
4	Overall	Integer	This attribute presents overall rating of the product ranging from 1 to 5.
5	reviewText	String	The descriptive review given by user.
6	reviewTime	Timestamp	The timestamp of the review.
7	reviewerID	String	The unique identifier of user.
8	reviewerName	String	The name of the user.
9	Summary	String	Overall summary of the review, however, the information given in this column showed that it doesn't present aspect terms.

We have collected the aforesaid kindle reviews of Amazon E-book and hence employed them into our inferencing by dividing the data into two separate methodological pipelines of statistical inferencing:

1. For the first method, we have utilized sentence-level segmentation and tried to see how this method works for our dataset.
2. For the other method, we have utilized document level segmentation to view and perceive how global information extraction for a document leads to better prediction quality.

In the first method, we have limited the view-scope of our algorithm to sentences only, while in the next method we have given our algorithms full access to the view-scope of

the entire document. The document-level methodology simplifies reasoning about knowledge extraction and providing faster convergence and learning rate.

For our use case, we can hypothesize that sentiment analysis of reviews can be boiled down to the overall flow of emotions of the whole document. This can thus be used as a technique for information extraction with the least amount of headache and latency. Using document-level information extraction can lead to an efficient and performant pipeline, which can be used and utilized effectively without losing generality and acceptable accuracy.

In this methodology, we tested out several of the NLP techniques and used various other variants of such techniques. For example, we have used various information providence levels to such algorithms as Adaboost, NaiveBayesClassifier, Deep neural network (DNN), Random Forest, etc.

BIGRAMS USAGE FOR CAPTURING OF LINGUISTIC FEATURES:

In linguistics related to computational resources, the sequence formed of adjacent two words taken from text data is termed as bigrams. Bigrams generally help with modeling the conditional probability of a given word in a situation where also its previous word is given. This technique is like the unigram model.

Below are some highlights of bigrams used here: -

It also tries to give the decision boundary between two consecutive entity tokens of text. This algorithm uses bigrams for the determination of the decision boundary.

In a way, it has a lookup of one word ahead to form the boundary of separation between the two classes.

The classifier with bigram tries to model the relationship between the output and the input with the help of a bigram of input text data.

The output here in this research is the sentiment analysis label for given text input.

This algorithm tries to incorporate sentiment information through bigram streams of input data.

This classifier with bigram uses any machine learning algorithm like k-nearest neighbors algorithm for classification by using bigrams formed from the given textual input data.

This algorithm focuses mainly on classification through decision-making being done on bigrams for comparison of nearness.

This algorithm has a limited window scope of word visibility.

In addition to this, algorithms like random forest use bigrams as input to determine the probable output. There is a variant of the random forest model that tries to minimize error based on word sequences using bigrams.

Below we will discuss both these methodologies in detail one by one.

3.1.2 METHOD 1 (sentence level)

The first line of the methodology followed in our machine learning model that has been configured to work with sentence-level information extraction has been discussed here first.

1. Data collection and annotation

In this method, data has been first collected, which comprises of reviews as sentences composed of text and their respective sentiments. The sentiments included positive, negative, neutral, and unnamed labels. We have processed the data, removed all un-labeled data rows, and processed the data further to become usable to our model pipeline.

The data have been comprised of 2412 lines or rows of data initially. Then null rows have been removed from the dataset, and the final size became 1534 rows of sentiment analysis data consisting of the polarity of a sentence assigned manually by human annotators and the review sentence statements along with Id.

2. Data Pre-processing

a- Data cleaning

Text cleaning refers to remove or eliminate unwanted information or characters from a given sentence. The review text is clean by removing unwanted characters (e.g., punctuations, HTML entities), short words, and optionally English stop words. While experimenting with removing stop words, some text may change its meaning if stop words are removed.

For example:

Class: negative
Origin: Save your money. ***This was not worth the time it took to read***. At least not for me. Gratuitous sex, not much plot. but then again, if that's what you're looking for, you might like it.
Clean: save money ***worth time took*** *read* least gratuitous sex much plot looking might like

The bold text shows that removing such a “not” word from the text may change its sentiment meaning. Finally, the advantage of text cleaning is as it reduces the data dimension and computation complexity decrease.

b- Feature Extraction:

This step involves transforming review text into a numeric representation. The most common technique is TF-IDF (Term Frequency - Inverse Document Frequency). This is done using scikit-learn’s TfidfVectorizer. We apply unigram (one word) and

unigrams-bigrams. ngram_range: the range of n-values for different n-grams to be extracted. For example, the range (1,1) allows only unigrams; range (1,2) allows both unigrams and bigrams. We ran our experiment with default parameters of TfidfVectorizer and machine/deep learning model to compare the accuracy and classification report, including precision/recall and f1 score metrics of each sentiment class.

3. Training and evaluation

Techniques of natural language processing (NLP) have been implemented, which are traditionally employed as a means of initial pre-processing of textual data. We have first cleansed the text data for any unnecessary symbols and unwanted tokens such as stop-words, short words. This has been followed as there might be some unprecedented symbols that can break our prediction pipeline and might result in lower prediction capabilities. Finally, after all our pre-processing and subsequent cleaning of the dataset subjected the resulting data to some preliminary data visualization. We divided the dataset as 80% for training purposes and 20% for testing purposes. Then, various machine learning models were trained on our annotated dataset and evaluated on different performance metrics like precision, recall, and f1 score.

3.1.3METHOD 2 (document level)

For the next methodology of our machine learning pipeline, which deals with document-level sentiment analysis, the steps followed were mirrored from that of the first case. Below, a discussion of this method has been carried out.

1. Data collection and annotation

The original dataset used here containing 982619 reviews was highly imbalanced. This imbalance could hinder our machine learning prediction accuracy. So, the dataset

was subjected to balancing of sentiment classes. The goal was to have a dataset of sufficient quantity of examples adequate for a proper machine learning model. The dataset should also be balanced in this regard. Owing to the limitation of computer resources, the dataset had to be reduced to having 9999 reviews for training and testing purposes. However, the reduced dataset was balanced, containing 3333 positive, 3333 negative, and 3333 neutral examples.

2. Data Pre-processing

For the Data Pre-processing, the procedure was completely mirrored as per the procedure followed in the previous method.

3. Training and evaluation

For the training and evaluation, the procedure was completely mirrored as per the procedure followed in the previous method. The algorithms used were completely the same. This was done to have a fair comparison between the two methods. The models were then evaluated on the same metrics as precision, recall, and f1 score.

3.2 ALGORITHMS USED

The algorithms used for this thesis can be categorized into two main categories. One is belonging to pure traditional machine learning algorithms, and the other belonging to algorithms used in deep learning techniques. The thesis tried to meddle with both classical machine learning and newly emerging deep learning techniques. Below, a detailed discussion of all the algorithms used in this thesis has been done.

1. Naive Bayes classifier

The most simplistic classifier based on probabilistic assumptions, especially that of Bayes theorem, is the Naïve Bayes classifier. This classifier assumes that there is strong

independence among the input model features for simplification of analysis (Kaviani&Dhotre, 2017). These are very easily trained, can achieve high accuracy, and can be scaled highly. Usually, these are trained by maximum likelihood estimation. These classifiers try to model the probability that given an observation, what is the probability of that observation belonging to a particular class. It gives the posterior probability given the prior probability and likelihood of observation under given evidence [43]. The following Naïve Bayes classifier learning function:

$$\text{Log } p(C_K \mid X) \propto \log\left(p(C_k) \prod_{i=1}^{n} pki^{xi}\right)$$

$$= \log p(C_K) + \sum_{i=1}^{n} x_i \,.\log pki$$

$$= b + W_K^T X$$

2. Logistic Regression

Regression refers to the process of statistical determination of the relationship between one or more independent variables and dependent variables. Logistic regression (Peng, Lee & Ingersoll, 2002) employs the logistic or sigmoid function to convert the output of a linear regression into the domain raging between two values. Generally, the two values are taken as 0 and 1, while they can also be taken as -1 and 1 based on the logistic function used for the regression analysis. In other words, it tries to separate two classes of entities by keeping the best line which separates both the classes maximally [44]. Below is the Logistic regression modeling equation:

$$\begin{aligned} Pr(Y_{i=1 \mid x_i}) &= Pr\,(Y_{i > 0 \mid x_i}) \\ &= Pr\,(\beta \,.\, x_i + \varepsilon_{>0}) \\ &= Pr\,(\varepsilon > -\beta \,.\, x_i) \\ &= Pr\,(\varepsilon < \beta \,.\, x_i) \\ &= \text{logit}^{-1}\,(\beta \,.\, x_i) \\ &= p_i \end{aligned}$$

3. AdaBoost Classifier

Adaptive boosting is also called AdaBoost in short (Tharwat, 2018). This meta-algorithm works by the boosting principle. Boosting refers to the usage of outputs from several weak classifiers and subsequent usage of these outputs to get the result from a strong classifier. This algorithm is an adaptive one as weak classifiers are trained based on instances of output misclassified in the previous round. This mainly aims at reducing the problem of overfitting. Generally, decision trees are used as weak learners for this method [45]. Below is AdBoost classifier modeling equation:

$$H(x)= \text{sign}\left(\sum_{t=1}^{t} \alpha_t \, h_t \, (x)\right)$$

4. SGD Classifier

SGD classifier uses stochastic gradient descent as the learning algorithm. This algorithm can thus support and provide for various loss functions and gives a greater degree of control over the learning (Diab, 2019). For hinge loss, this algorithm behaves similarly to that of an SVM trained on linear data. This algorithm finds great use in various parts of statistical modeling and hence has been a pioneering algorithm to be used for any modeling task. Due to its simplicity and variability, it is used in the preliminary stages of statistical investigation often rather than employing complex algorithms [46]. Below is SGD classifier formula:

$$V_t = \beta \, V_{t-1} + \alpha \, \nabla_w \, L \, (W , X, Y)$$

$$W=W- V_t$$

5. KNeighbors Classifier

KNeighbors classifier uses the famous K- nearest neighbors algorithm for classifying (Cunningham & Delany, 2007). This algorithm samples the nearest k examples and tries

to determine the class of the new example. This algorithm doesn't have a training stage as it just takes the data and uses it only during prediction. One thing of this algorithm that can decide its efficiency is its distance metric used for defining nearness. It has been found that various distance metrics tend to work well in differing circumstances, and hence choosing an appropriate one becomes the main hyperparameter tuning step of this algorithm [47]. Below is Kneighbors classifier formula:

$$D(p,q) = d(q,p) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2}$$

$$= \sqrt{\sum_{i=1}^{n}(q_i - p_i)^2}$$

6. Random Forest Classifier

Random Forest Classifier incorporates the principle of ensemble learning by the formation of several decision trees during training (Xu, et al., 2012). This algorithm usually performs voting by the majority to decide the final output of classification from the prediction outputs done by each of the constituent decision trees. Generally, a random forest model works better than a single decision tree, but the accuracy of such a model is generally lower than gradient boosted trees. This algorithm is a general-purpose model which is often sought for use in various application. They generate very likely and highly reasonable outputs as predictions and apply to a wide range of data. They also require very little configuration and can work right off the bat [48]. Below is Random Forest classifier aggregation formula:

$$K_k^{cc}(x,z) = \sum_{k_1,\ldots,k_d,\sum_{j=1}^{d} k_j = k} \frac{k!}{k_1! \ldots k_d!} \left(\frac{1}{d}\right)^k \prod_{j=1}^{d} 1_{[2^k j\, x_j] = [2^k j\, z_j]},$$

For all x,z ∈ $[0,1]^d$.

7. Neural Networks (Deep Learning)

Neural Networks are a class of algorithms that try to mimic the functions of a neuron of our brain. It tries to recognize and replicate an underlying relationship between a set of inputs and outputs by mimicking the procedure through which the human brain works in its simplest sense (Sarvepalli, 2015). This is formed of individual units called artificial neurons, which hire and give outputs on receiving a specific input above a certain threshold. In such a network, there happens to be at least one input layer and one output layer with or without one or more hidden layers. Usually, back propagation algorithm and stochastic gradient descent are mainly used as learning algorithms for this model. These networks find application in various fields and can be suited appropriately very easily according to the needs. Here, we have utilized a sequential neural network model for the identification of the sentiment of text based on input streams of the tokens of the text [49].

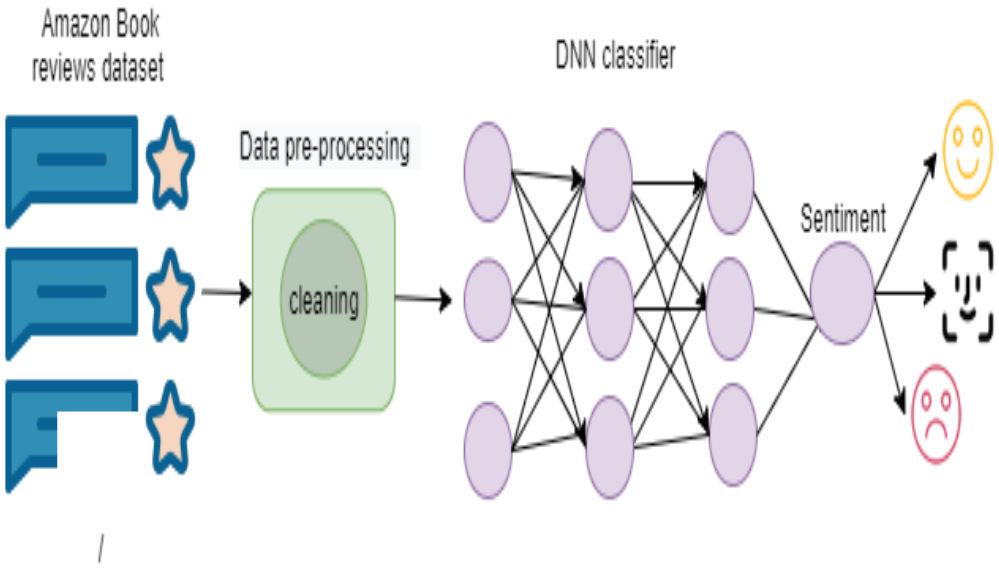

Figure 3: Deep Neural Network

8. Deep Learning with Word Embeddings

In this methodology, we employed a neural network model along with the use of a pre-trained word embedding (Wang, Zhou, & Jiang, 2020). We used word embedding in conjunction with the sequential neural network model to achieve higher accuracy over a traditional bag-of-words encoding model as word embedding representation can reveal many hidden relationships between words (Weng, 2017) [53]. The usage of word embedding generally is to better modeling of word tokens so that the model performs better [50].

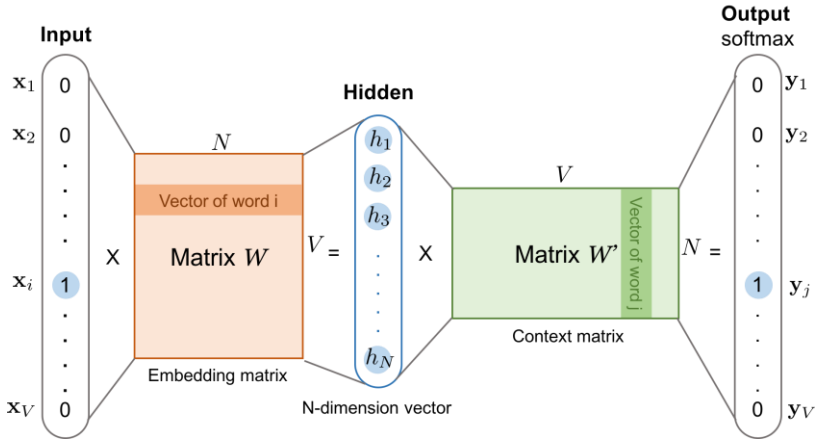

Figure 4: Deep learning with Word Embedding

.

9. Deep Learning with Part of Speech

For this method, utilization of part of speech tags along with usage of a deep learning neural network architecture (Kumar, Kumar, &Kp, 2018). This employs the usage of part of Speech tags and thereby forming a tokenizer and formation of word embedding level encoders so that a deep frequential neural network model can be trained on our dataset. Usage of part of speech was employed to provide a greater amount of information to the

deep learning model so that the model can learn better. This may lead to capturing of some underlying hidden relationship between the input and output [51].

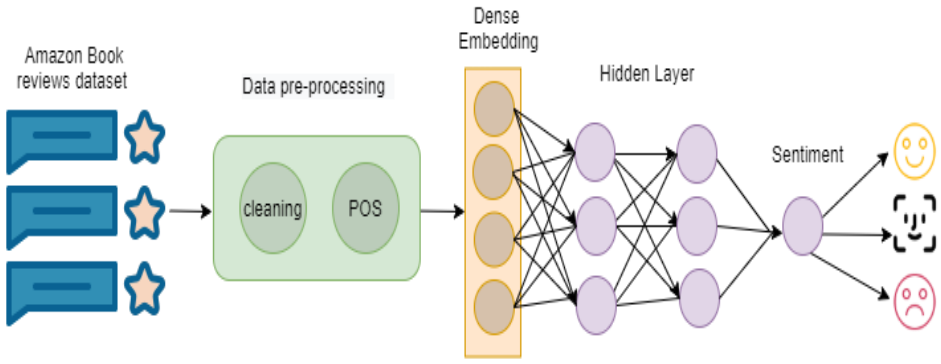

Figure 5: Deep learning with parts of speech.

10. Simple Baseline

This algorithm gives us the simplest baseline of model for performance based on the metrics of precision, recall, and f1 score. It is a very rudimentary and random learner which gives the same output for every type of input as here in the research wok the output was a classification of sentiments of different classes; this algorithm thus provides us a clear understanding of the baseline from which we can compare a consider our model performances based on the precision, recall, f1 score metrics. This algorithm gives us a rough idea of how worse a model can perform if it were only to predict and provide the output of only one type of class.

3.3 Training

For training, the research work has devoted its focus to the supervised machine learning model, in which the first step is to collect and pre-process the data and then feed it into our model for training purposes. The training can be elaborated as: -

1. For the machine learning algorithms, the training mainly used algorithms right from the scikit-learn library of Python.
2. These algorithms were fed the processed annotated data for sentiment analysis in the current context of this research.
3. For the deep learning algorithms, the neural network architectures the libraries utilized for training included Keras for model building, Scikit-learn for learning utilities, Nltk and Spacy for text processing, etc.
4. The pipeline thus incorporates many components to deliver the final products as a ready to deploy models.
5. The training part also considers various used techniques that employ different methodologies and lines of reasoning. This has been done to test out which of the varying techniques would result in the best model for our research purpose.
6. The training also tries to incorporate proper visualization steps so that the model training progress can be visualized appropriately. This has been done to view and decide the course of learning of the said algorithms in the correct way.

This training phase serves as the pivotal stage in both of our methodologies in this research project for making and getting the most out of our models. This training phase concludes with the resultant models which are then passed on to the next phase, which is the evaluation phase of our machine learning pipeline.

3.4 Evaluation

The evaluation stage of our machine learning pipeline denotes the stage in which the resultant models which have been obtained from the learning or training phase have been subjected to the traditionally employed testing or evaluation through the usage of various

performance metrics. This is done to evaluate how well the trained models perform when subjected to unseen data taken from the real world. This stage can be summarized as: -

1. The evaluation for our research purpose has been done by using a portion of the data as an evaluation set and for measuring the performances of the generated trained models, the performance metrics used were accuracy, precision, recall, and f1-score.
2. Accuracy is the measure of how close a measurement is to the truth. Accuracy is determined by how close a measurement is to an existing value that has been measured by many scientists.

$$\textbf{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$

where:

TP = True positive; FP = False positive; TN = True negative; FN = False negative.

3. Precision refers to the metrics which measure how many of the predicted outputs were predicted correctly. In other words, it tries to show how precise the model is in doing the predictions for the classification tasks at hand.

$$\textbf{Precision} = \frac{TP}{TP+FP}$$

4. Recall employs the concept of measuring the ratio of correctly identified classification outputs to the total number of actual instances of that class. In other words, this tries to show how the model performs in the scope of recalling the correct classification of an instance out of all the instances of that class shown to the model.

$$\textbf{Recall} = \frac{TP}{TP+FN}$$

5. F1-score can be said to be the harmonic mean of both precision and recall of a model which has been scaled appropriately.

$$\textbf{F1 Score} = 2*\frac{Precision*Recall}{Precision+Recall}$$

6. Precision matters in places in which false-positive rates matter more, while recall matters in places in which the false-negative rates matter highly.
7. F1-score is employed to obtain a balance between both the error of first-class and second class.
8. This can be seen as to be aiming at reducing the overall model error rate so that the model can have the least amount of bias and variance errors and can be employed for the generalization of the task at hand.
9. The result is that this allows us to select the best model, which can give high accuracy for real-world usage.

This thus employs one of the most important stages of our machine learning pipeline and strives for allowing selection of the best model at the end of the training stage. This thus concludes the evaluation stage of our machine learning pipeline.

3.5 Designing of a rating system based on sentiment analysis

Based on the resultant models, the end goal for our research work and analysis was to design a robust rating system that was based on sentiment analysis of the reviews of the users rather than using the already existing star-based rating system as is generally employed on most of the platforms. Some highlights of our proposed system:

1. The new system that has been proposed tries to do away with lack of the current star-based rating system, which does not necessarily capture the user sentiment related to the product and tries to quantize the data.
2. This quantization can be beneficial for the modeling system, which must deal with easier and lesser data, but in the process of doing so, it loses its original intent of usage, which is to give a clear understanding and one to one corresponding to the users or customer sentiments in real-time.
3. The proposed rating system tried to do away with such discrepancies and loopholes of the current rating system by employing the usage of the reviews received for a product from the users of that product.
4. This has been done by analyzing the sentiments of these reviews received for the product.

The algorithm used has been detailed below:

1. The algorithm tries to get the sentiments of each of the reviews first.
2. Then for each of these reviews whose sentiments we have found, it tries to get a score for the rating.
3. For each neutral review sentence, the algorithm gives a rating of 3, which serves as the middle ground of our 5-point scale of the proposed rating system.
4. Then for each negative review, it maps the prediction output probability from the range [0.0, 1.0] to the overall score of 2 or 1 in the 5-point scale. The higher the negative probability is, the lower the overall score. That is if the probability is near 1.0, it means the overall score is likely to be 1, and if the probability is near 0.0, it means the overall score is likely to be 2.

5. Like each positive review, the algorithm maps the prediction output probability from the range [0.0, 1.0] to the overall score of 4 or 5 in the 5-point scale. The higher the positive probability is, the higher the overall score. That is, if the probability is near 1.0, it means the overall score is likely to be 5, and if the probability is near 0.0, it means the overall score is likely to be 4.
6. Then all the reviews are aggregated accordingly and then it gives the final resulting rating for that product.

This algorithm thus gives us a rating system with the desired qualities and does not suffer from the loopholes and drawbacks of the original star rating system employed in most places.

The system encapsulates the goals that were proposed earlier for the proper design of the rating system. This algorithm designed finally gives the best architecture to employ the sentiment analysis and thus use this for production purposes. This concludes the design aspect of the proposed algorithm for a rating system based on sentiment analysis.

In this rating system, we use two variants of the tick percentage Accuracy (PTA) [54]. To calculate the Percentage Tick Accuracy (PTA), we convert the y_i and $\hat{y}_i$ for each review. Next, we calculated the tick as the difference between two consecutive reviews for the predicted review and the observed review. In this study, we applied two levels of PTA as follows:

$$PTA_0 = \frac{r_0}{n}$$

Where r_0 is the number of observations in the test set that predicted review is equal to the observed review, and *n* is the number of observations in the testing set.

$$PTA_1 = \frac{r_1}{n}$$

Where r_1 is the number of observations in the test set that met the following condition: The difference between the predicted and observed examination is less than or equal to one consecutive examination, and *n* is the number of observations in the test set.

In all variants of PTA, higher values (close to 1) indicate a better prediction model. A perfect prediction model is a model with PTA0 = 1, which indicates that all reviews were correctly classified in the testing set.

CHAPTER 4

RESULT AND DISCUSSION

In this section, we discourse about the result we achieve it and our step development.

4.1 Results Obtained (Model comparison)

In this thesis, each review has been classified as positive, negative, or neutral based on the rating reviews from 1 to 5. After that, we will run experiments on our data, which means we took the same number of positive, negative, and neutral reviews.

1. Method 1 results (sentence level)

Table 2:Method 1 results (sentence level)

Classification method	Accuracy	Recall	Precision	F1 Score
Neural Network (Deep Learning)	0.66	0.65	0.66	0.65
Logistic Regression	0.64	0.62	0.67	0.63
SGD Classifier	0.64	0.62	0.66	0.62
RandomForest Classifier	0.59	0.59	0.62	0.58
Deep Learning with Part of Speech	0.59	0.56	0.64	0.57
Naive Bayes classifier	0.57	0.53	0.69	0.53
AdaBoost Classifier	0.56	0.55	0.56	0.55
Deep Learning with Word Embeddings	0.51	0.51	0.51	0.51
KNeighbors Classifier	0.37	0.37	0.40	0.23
Simple Baseline	0.40	0.33	0.13	0.19

In terms of accuracy, recall, precision, and F1score, we can notice that Neural Networks (Deep Learning) outperforms all the other models, and the simple baseline model and KNNeighbors Classifier with bigram have the least performances.

From the above table, we can notice that the rest of the models are following with a slight decrease in the Neural Networks (Deep Learning) in terms of model performance.

2. Method 2 results (document level)

Table 3:Method 2 results (document level)

Classification method	Accuracy	Precision	Recall	F1 Score
Neural Network (Deep Learning)	0.761	0.76	0.76	0.76
Naive Bayes classifier	0.755	0.77	0.76	0.76
LogisticRegression	0.755	0.76	0.76	0.76
SGDClassifier	0.748	0.75	0.75	0.75
Deep Learning with Word Embeddings	0.743	0.74	0.74	0.74
RandomForestClassifier	0.735	0.73	0.74	0.73
AdaBoostClassifier	0.668	0.67	0.67	0.67
Deep Learning with POS	0.661	0.66	0.66	0.66
KNeighbors Classifier	0.549	0.59	0.55	0.55
Simple Baseline	0.333	0.11	0.33	0.17

The entire dataset of 9999 reviews was divided into a training set (80%) and a test set (20%). We have implemented the Naive Bayes classifier, Logistic Regression, Ada Boost Classifier, Stochastic Gradient Descent Classifier, K Nearest Neighbors Classifier, Random Forest Classifier, Neural networks, Deep learning with word embeddings, deep learning with part of speech (POS), and simple baseline.

From the above table, we have the following observations:

- The Simple Baseline accuracy is 33% that is around the general probability of a sentient class (one out of three possible classes). This is indeed the lowest performance classifier.
- Naive Bayes classifier outperforms the other models in terms of Precision (77%), which means it has the highest true positive predicted ratio among the entire predicted positive. And both the Naive Bayes classifier and Neural Network achieved 76% precision.
- However, that the Neural Network classifier outperforms the other models in terms of Accuracy (76.1%), which is the overall accuracy. And both the Naive Bayes classifier and Logistic Regression achieved 75.5% accuracy.
- In terms of Recall and F1 score, the highest value is 0.76 that all achieved by the Naive Bayes Classifier, Logistic Regression, and Neural Networks (Deep Learning) classifier.
- The SGD classifier, Deep Learning with Word Embeddings, and Random Forest classifier achieved all metrics around 73% - 74%.
- The three remaining classifiers, the AdaBoost, Deep Learning with POS, and K-neighbor classifier, all performance metrics drop down to 67%. The worst-performance classifier is the K-neighbor classifier.

4.2 Development the result

In this section, our experiment focuses on turning hyper-parameters of the Feature Extraction and Feature Selection to find the highest accuracy with two classifier models: Naive Bayes classifier and Neural Network. Since the Neural Network classifier gave us

the highest accuracy of 76%, while the Naive Bayes classifier gave us the highest precision of 77%.

We extract the sample dataset from the provided dataset to create the main dataset for the sentiment classifier. This dataset contains 1000 samples per class. So, the whole dataset contains 3000 samples. This is large enough for evaluating the accuracy of our classifier while having reasonable running time. At the end of this experiment, the model is tested on a larger dataset that contains 9999 samples (3333 samples per class).

4.2.1 TF-IDF Feature Extraction

This step involves transforming review text into a numeric representation. The most common technique is TF-IDF (Term Frequency - Inverse Document Frequency). This is done using scikit-learn's TfidfVectorizer class. The following hyper-parameters are used in this experiment:

• Min_df: the minimum frequency of a term to be included. In other words, ignore terms that have a document frequency strictly lower than the given threshold.

• Max_df: the maximum frequency of a term to be included. In other words, ignore terms that have a document frequency strictly higher than the given threshold.

• Ngram_range: the range of n-values for different n-grams to be extracted. For example, the range (1,1) allows only unigrams; range (1,2) allows both unigrams and bigrams.

our experiment runs with the following steps:

• With ngram_range = (1,1) (only extracting unigrams):

• Experiment with min_df from [1..10].

• Pick min_df that results in the highest accuracy.

• The experiment with max_df from [0.5, 0.6, .. 1.0] to observe the improvement in accuracy result.

• Repeat the above steps with ngram_range = (1,2) (extracting unigrams and bigrams)

4.2.1.1 Naive Bayes classifier

With unigrams extraction, changing in min_df parameter results in changing inaccuracy. The best min_df value is three, which results in the highest accuracy of 76.7%. This is an improvement in comparison with the previous accuracy we achieved, that was (75.5%) where running time is significantly reduced. Having min_df = 3, max_df parameter is experimented, which shows no improvement in classifier accuracy. We can develop results, and I achieve the highest accuracy that is 77.5%, with min_df=10 and max_df=0.5. Also, the same experiment is run with unigrams-bigrams extraction. The accuracy is improved over unigram extraction. The accuracy is increased to 78% with min_df=4. Like bigrams, changing max_df has a minimum effect on the accuracy result.

Finally, we can develop results, and we achieve the highest accuracy that is 79.3%, with min_df=5 and max_df=0.6

4.2.1.2 Neural Network classifier

With the same experiment running on Neural Network classifier, the accuracies are around 78.3% with both unigrams and unigrams-bigrams extractions. The overall performance is lower than the Naive Bayes classifier in both accuracies and running time metrics.

Table 4: Comparison of accuracy development between Naive Bayes and Neural Network

Accuracy (%)	With remove stop word		Without remove stop word	
	Unigrams	Unigrams-Bigrams	Unigrams	Unigrams-Bigrams
Naïve Bayes	76.7	78.0	77.5	79.3
Neural Network	76.5	76.8	76.8	78.3

4.2.2 Accuracy Improvement with Feature Selection

In the experiment above, we can decide the classifier model as well as its parameters to achieve the highest accuracy so far of 79.3% with:

• Naive Bayes classifier

• Tf-idfVectorizer with unigrams and bigrams, min_df = 5 and max_df = 0.6

This is given as the baseline model for improvement. In this step, we apply Feature Selection, which is a technique to choose the "best" features that contribute most to the classification target. It is done by using sci-learn's SelectKBest after performing TF-IDF feature extraction to select the k-best features. Our experiment is done with two scoring methods, chi2 and f_classif., And k in range 500 to 8000 to observe the improvement in accuracy. There is a significant improvement in accuracy from 79.3% to higher than 80% without removing the stop word. The result is slightly better f_classif scoring and k = 4500.

To confirm our experiment, we tested with the larger dataset of 9999 samples. The accuracy is achieved to 84% with the following model:

• Naive Bayes classifier

• Tf-idfVectorizer with unigrams and bigrams, min_df = 5 and max_df = 0.6

• SelectKBest with k = 10,000 and chi2 score function.

2.3 PTA for the Developed Ranking System

1. PTAs for reviews

Table 5: Snapshot of ranking result (pred_overall column) in comparison with human rating (overall column) per review level

ReviewerID	Overall	Pred_Overall
A00085083TSCV82430YT4	5	5
A0010876CNE3ILIM9HV0	4	4
A00207583M69Q8KX3BOFQ	5	4
A002359833QJM7OQHCXWY	4	4
A00328401T70RFN4P1IT6	5	4

2. PTAs for developed rating method analysis

Table 6: Snapshot of ranking result (pred_overal column) in comparison with human rating (overall column) per book rating level

Asin	Overall	Pred_Overall
B000F83SZQ	4	3
B000FA64PA	4	3
B000FA64PK	4	4
B000FA64QO	4	3
B000FBFMVG	4	4

After making the review predictions, we concatenate the original amazon reviews. Starting from that point, we have created the PTA's system to count the difference between the two rows, the overall and the predicted row. In this stage, we apply our sentiment model to predict the review rate for all reviews in our dataset. The following table shows statistical factors related to our dataset.

Table 7: Number of reviews, unique users, and unique books in the whole dataset

Number of reviews	**982,597**
Number of unique users	68,223
Number of unique books	61,934

We apply PTA's evaluation metrics in two levels; the first level considered reviews, and the second level focuses on book rating rather than reviews.

Table 8: PTA0 and PTA1 evaluation in review level and book rating level

Evaluation based on:	**PTA0**	**PTA1**
Reviews	0.58	0.98
Book rating	0.63	0.99

For PTA-0, if the difference between the two rows is equal to zero, then the algorithm will count plus one, which means that the overall and the predicted value are equal. We have the result of 58% (39546 correct predictions over 68,223 reviewers) on the reviewer level and 63% (38756 correct predictions over 61,934 books) on the book level.

The second algorithm, PTA1 will count for all differences between zero and one. We have the result of 98% (66,821 correct observations over 68,223 reviewers) on the reviewer level and 99% (61,441 correct predictions over 61,934 books) on the book level.

Finally, we can conclude that our experiment design has been robust enough to be able to give us distinguished and viable research findings based on which we can then build our satisfactorily a well-established and efficient sentiment-based rating system that can provide deeper insights into the rating of perceiving by the customers.

CHAPTER 5

CONCLUSION AND FUTURE WORK

Reviews are essential for both individuals and companies. Consumers use them to make good decisions before buying a specific product, and businesses use them to find out how satisfied their consumers are with the products. This ensures that the customers can make optimal decisions based on the input query. In this work, we have presented a novel approach for a user query-based rating system that gives shoppers and reviews corresponding to them if required as output as per the user queries. Our experiments are based on the customer reviews dataset collected from Amazon E-book.

In this research, we investigated sentiment analysis of the Amazon E-book using different types of machine learning classifiers such as Logistic Regression (LR), Naıve Bayes (NB), Stochastic Gradient Decent (SGD), and deep learning algorithms such as neural networks. These algorithms are applied using different feature extraction approaches. We applied the word frequency technique to interpret the reasons for classifying reviews as positive, negative, or neutral. Concerning model evaluation, the Naïve Bayes classifier offers the best performance compared to all other algorithms for the data used in our work. Then we applied the PTA algorithm to give us a better review with only a difference equal to zero or/and one between the real and the predicted review.

Despite the good results obtained by our model, few limitations can be considered in the future. Firstly, as recent opinions are consulted more regularly, examining the evolution of consumer approval of opinions over time can provide us with additional information. Methods to address the class imbalance, such as over-or random under-

sampling, bootstrap-based aggregation techniques, or boosting techniques, could be tested for better results. Methods such as the fuzzy set method or the Dempster-Shafer method, etc., could be tested to form a set of methods.

In this study, we implemented ten types of algorithms. Among the algorithms that remain to be applied in future work are SVM, naive LSTM, and maximum entropy. Then we will compare the result with the one we obtained in this current study. We also intend to add the Arabic language to increase the scope of the research. Our research has certain limitations: NLP is a relatively new subject and very advanced, so it requires a lot of research to understand the field and how it works. Also, we encountered problems with computer memory, which made the experiments very time-consuming. We also used Google Colab to increase performance, but it did not give us the expected speed.

Finally, this study can be of great practical importance. The proposed system can be integrated into e-commerce systems in different ways. Instead of searching rigorously for products with better reviews on various online websites, our approach can be adopted by shoppers to find products with aspects such as better staff, better value, etc., and reviews.

REFERENCES

[1] Kang, M., Choi, Y., & Choi, J. (2019). The effect of celebrity endorsement on sustainable firm value: evidence from the Korean telecommunications industry. International Journal of Advertising, 38(4, 563–576).

[2] Whang, H., Ko, E., Zhang, T., & Mattila, P. (2015). Brand popularity as an advertising cue affecting consumer evaluation on sustainable brands: a comparison study of Korea, China, and Russia. International Journal of Advertising, 34(5, 789-811).

[3] Vriens, M., Chen, S., & Vidden, C. (2019). Mapping brand similarities: Comparing consumer online comments versus survey data. International Journal of Market Research, 61(2,130–139), 130-139.

[4] Timoshenko, A. and Hauser, J., 2020. Identifying Customer Needs From User-Generated Content. 4th ed. Marketing Science.

[5] Fan, Z., Xi, Y., & Li, Y. (2018). Supporting the purchase decisions of consumers: A comprehensive method for selecting desirable online products. Kybernetes, 47(4), 689–715.

[6] McCarthy, J., Minsky, M., Rochester, N. and Shannon, C., 2006. A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence. AI Magazine, (Vol. 27 No. 4: Winter 2006).

[7] Badruddin, M. and Mohammed, E., 2017. Machine Learning Algorithms For Industrial Applications. CRC Press.

[8] Cambria, E. and White, B., 2014. Jumping NLP Curves: A Review of Natural Language Processing Research. IEEE Computational Intelligence Magazine, 9(2), pp.48-57.

[9] Liu, B., 2012. Sentiment Analysis And Opinion Mining. Morgan & Claypool Publishers, pp.7-12.

[10] Mukhopadhyay, S., Booth, A., Calkins, S., Doxtader, E., Fine, S., & Gardner, J. et al. (2020). Leveraging Technology for Remote Learning in the Era of COVID-19 and Social

Distancing. Archives Of Pathology & Laboratory Medicine, 144(9), 1027-1036. doi: 10.5858/arpa.2020-0201-ed

[11] In Library.tiu.edu .,2020. Libguides: Free Resources During COVID-19 Crisis: E-books And More.

[12] Khan, I., 2020. Amazon reports profit, sales surge amid COVID-19 spending. Lose Angeles Times.

[13] Pang, B. and Lee, L., 2008. Opinion Mining And Sentiment Analysis. Foundations and Trends in Information Retrieval.

[14] Patil, P. and Yalagi, P., 2016. Sentiment Analysis Levels and Techniques: A Survey. International Journal of Innovations in Engineering and Technology (IJIET), 6(4), pp.523-528.

[15] Jurafsky, D., & Martin, J. (2008). Speech and Language Processing, 2nd Edition (2nd ed.). Prentice-Hall.

[16] Pang, B. and Lee, L., 2008. Opinion Mining And Sentiment Analysis. Foundations and Trends in Information Retrieval.

[17] Chesley, P. Vincent, B. Xu, L. and Srihari, RK., 2006. Using verbs and adjectives to automatically classify blog sentiment. Training 580(263):233

[18] Choi, Y., & Cardie, C. (2009). Adapting a polarity lexicon using integer linear programming for domain-specific sentiment classification. In Conference on Empirical Methods in Natural Language Processing (pp. 590–598). PA, USA.: Association for Computational Linguistics, Stroudsburg.

[19] Tan, L.K-W, Na, J. Theng, Y-L. Chang, K . (2011) Sentence-level sentiment polarity classification using a linguistic approach. In: Digital Libraries: For Cultural Heritage, Knowledge Dissemination, and Future Creation. Springer, Heidelberg, Germany. pp 77–87

[20] Zhang, C., Zeng, D., Li, J., Wang, F. and Zuo, W., 2009. Sentiment analysis of Chinese documents: From sentence to document level. Journal of the American Society for Information Science and Technology, 60(12), pp.2474-2487.

[21] Meena, A., & Prabhakar, T. V. (2007, April). Sentence level sentiment analysis in the presence of conjuncts using linguistic analysis. In European conference on information retrieval (pp. 573-580). Springer, Berlin, Heidelberg.

[22] Appel, O., Chiclana, F., Carter, J., & Fujita, H. (2016). A hybrid approach to the sentiment analysis problem at the sentence level. Knowledge-Based Systems, 108, 110-124.

[23] Shoukry, A., & Rafea, A. (2012, May). Sentence-level Arabic sentiment analysis. In 2012 International Conference on Collaboration Technologies and Systems (CTS) (pp. 546-550). IEEE.

[24] Moraes, R., Valiati, J. F., & Neto, W. P. G. (2013). Document-level sentiment classification: An empirical comparison between SVM and ANN. Expert Systems with Applications, 40(2), 621-633.

[25] Dou, Z. Y. (2017, September). Capturing user and product information for document-level sentiment analysis with a deep memory network. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (pp. 521-526).

[26] Tang, D. (2015, February). Sentiment-specific representation learning for document-level sentiment analysis. In Proceedings of the eighth ACM international conference on web search and data mining (pp. 447-452).

[27] Rhanoui, M., Mikram, M., Yousfi, S., & Barzali, S. (2019). A CNN-BiLSTM model for document-level sentiment analysis. Machine Learning and Knowledge Extraction, 1(3), 832-847.

[28] Kaggle.com. Amazon reviews: Kindle Store Category. [online] Available at: <https://www.kaggle.com/bharadwaj6/kindle-reviews>.

[29] Bhatia, P., Ji, Y., & Eisenstein, J. (2015). Better document-level sentiment analysis from rst discourse parsing. arXiv preprint arXiv:1509.01599.

[30] Guo, C., Du, Z., & Kou, X. (2018). Products ranking through aspect-based sentiment analysis of online heterogeneous reviews. Journal of Systems Science and Systems Engineering, 27(5), 542-558.

[31] Ghose, A., & Ipeirotis, P. G. (2007, August). Designing novel review ranking systems: predicting the usefulness and impact of reviews. In Proceedings of the ninth international conference on Electronic commerce (pp. 303-310).

[32] Medhat, W., Hassan, A., & Korashy, H. (2014). Sentiment analysis algorithms and applications: A survey. Ain Shams engineering journal, 5(4), 1093-1113.

[33] Najmi, E., Hashmi, K., Malik, Z., Rezgui, A., & Khan, H. U. (2015). CAPRA: a comprehensive approach to product ranking using customer reviews. Computing, 97(8), 843-867.

[34] Chen, K., Kou, G., Shang, J., & Chen, Y. (2015). Visualizing market structure through online product reviews: Integrate topic modeling, TOPSIS, and multi-dimensional scaling approaches. Electronic Commerce Research and Applications, 14(1), 58-74.

[35] Peng, Y., Kou, G., & Li, J. (2014). A fuzzy PROMETHEE approach for mining customer reviews in Chinese. Arabian Journal for Science and Engineering, 39(6), 5245-5252.

[36] Zhang, K., & Ramanathan Narayanan, A. C. CUCIS Technical Report Mining Online Customer Reviews for Ranking Products.

[37] Hassan Khan, F., Qamar, U., & Bashir, S. (2015). Building normalized SentiMI to enhance semi-supervised sentiment analysis. Journal of Intelligent & Fuzzy Systems, 29(5), 1805-1816.

[38] Moreo, A., Romero, M., Castro, J. L., & Zurita, J. M. (2012). Lexicon-based comments-oriented news sentiment analyzer system. Expert Systems with Applications, 39(10), 9166-9180.

[39] Khan, F. H., Bashir, S., & Qamar, U. (2014). TOM: Twitter opinion mining framework using a hybrid classification scheme. Decision support systems, 57, 245-257.

[40] Khan, F. H., Qamar, U., & Bashir, S. (2016). SWIMS: Semi-supervised subjective feature weighting and intelligent model selection for sentiment analysis. Knowledge-Based Systems, 100, 97-111.

[41] Zhang, D., Xu, H., Su, Z., & Xu, Y. (2015). Chinese comments sentiment classification based on word2vec and SVMperf. Expert Systems with Applications, 42(4), 1857-1863.

[42] Shi, H. X., & Li, X. J. (2011, July). A sentiment analysis model for hotel reviews based on supervised learning. In 2011 International Conference on Machine Learning and Cybernetics (Vol. 3, pp. 950-954). IEEE.

[34] Kaviani, Pouria &Dhotre, Sunita. (2017). Short Survey on Naive Bayes Algorithm. International Journal of Advanced Research in Computer Science and Management. 04.

[44] Peng, Joanne & Lee, Kuk & Ingersoll, Gary. (2002). An Introduction to Logistic Regression Analysis and Reporting. Journal of Educational Research - J EDUC RES. 96. 3-14. 10.1080/00220670209598786.

[45] Tharwat, Alaa. (2018). AdaBoost classifier: an overview. 10.13140/RG.2.2.19929.01122.

[46] Diab, Shadi. (2019). Optimizing Stochastic Gradient Descent in Text Classification Based on Fine-Tuning Hyper-Parameters Approach. A Case Study on Automatic Classification of Global Terrorist Attacks. International Journal of Computer Science and Information Security, 16. 155-160.

[47] Cunningham, Padraig & Delany, Sarah. (2007). k-Nearest neighbor classifiers. MultClassif Syst.

[48] Xu, Baoxun& Guo, Xiufeng& Ye, Yunming& Cheng, Jiefeng. (2012). An Improved Random Forest Classifier for Text Categorization. Journal of Computers. 7. 10.4304/jcp.7.12.2913-2920.

[49] Sarvepalli, Sarat Kumar. (2015). Deep Learning in Neural Networks: The science behind an Artificial Brain. 10.13140/RG.2.2.22512.71682.

[50] Wang, S., Zhou, W. & Jiang, C. A survey of word embeddings based on deep learning. Computing 102, 717–740 (2020). https://doi.org/10.1007/s00607-019-00768-7

[51] Kumar S, Sachin& Kumar, M. &Kp, Soman. (2018). Deep Learning-Based Part-of-Speech Tagging for Malayalam Twitter Data (Special Issue: Deep Learning Techniques for Natural Language Processing). Journal of Intelligent Systems. 28. 10.1515/jisys-2017-0520.

[52] Tang, C., Luktarhan, N., & Zhao, Y. (2020). SAAE-DNN: Deep Learning Method on Intrusion Detection. *Symmetry*, *12*(10), 1695.

[53] Weng, L. (2017). Learning Word Embedding.

[54] Ren, Z., Ning, X., Lan, A. S., & Rangwala, H. (2019). Grade Prediction Based on Cumulative Knowledge and Co-taken Courses. *International Educational Data Mining Society*.