SENTIMENT ANALYSIS FOR E-COMMERCE CUSTOMER REVIEW IN BANGLA USING MACHINE LEARNING ALGORITHM

By

Anika Afrin Juthy
&
Nusrat Binte Abedin

Department of Computer Science and Engineering
University of Global Village (UGV), Barisal
C&B Road, Barisal

May 2023

SENTIMENT ANALYSIS FOR E-COMMERCE CUSTOMER REVIEW IN BANGLA USING MACHINE LEARNING ALGORITHM

By

Anika Afrin Juthy
Id No -1182478
&
Nusrat Binte Abedin
Id No -1182085

A thesis submitted in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering

Department of Computer Science and Engineering
University of Global Village (UGV), Barisal
C&B Road, Barisal

May,2023

Declaration

This is to certify that the thesis work entitled “Sentiment analysis for e-commerce customer review in Bangla using machine learning algorithm” has been carried out by Anika Afrin Juthy and Nusrat Binte Abedin in the Department of Computer Science and Engineering, University of Global Village (UGV), Barisal. The thesis work, or any portion of it, has never been submitted to a university or other institution for the granting of a degree.

Signature of Supervisor

Signature of Candidate

Approval

This is to certify that the thesis work submitted by Anika Afrin Juthy and Nusrat Binte Abedin entitled “Sentiment analysis for e-commerce customer review in Bangla using machine learning algorithm” has been approved by the board of examiners for the partial fulfillment of the requirements for the degree of Science in Computer Science and Engineering in the Department of Computer Science and Engineering, University of Global Village (UGV), Barisal, Bangladesh in May 2023.

BOARD OF EXAMINERS

1. _____ Chairman (Supervisor)
   Md. Tariqul Islam
   Lecturer
   Dept. of Computer Science and Engineering
   University of Global Village (UGV), Barisal
2. _____ Member
   Md. Riadul Islam
   Lecturer
   Dept. of Computer Science and Engineering
   University of Global Village (UGV), Barisal
3. _____ Member
   Md. Masudur Rahman
   Lecturer
   Dept. of Computer Science and Engineering
   University of Global Village (UGV), Barisal
4. _____ Member
   Muntasir Rahman
   Lecturer
   Dept. of Computer Science and Engineering
   University of Global Village (UGV), Barisal

Acknowledgment

In the Name of Allah, the most Merciful, the most Kindness. Alhamdulillah, very grateful to Allah as He had given me the opportunity to finish this final year project and thesis paper. We would like to take this opportunity to thank my supervisor Md Tariqul Islam, who always gives us the advice and suggestions for helping me to complete this final year project and thesis paper. Thanks to my supervisor, we got an opportunity to gain more understanding and learn about how to do research in the computer science field. Apart from that, my supervisor also gives a lot of comments regarding my documentation. We can improve my documentation and thesis by following my supervisor's instructions. His encouragement and guidance also boosted our strength in completing this thesis paper. Apart from that, we are also very grateful to our friends who are with us through this thorny journey. Our friends are very helpful and give us a lot of hints for getting better in write a document to our ideas. We firmly believe that our thesis paper will not be satisfactorily completed without the support and guidance of our friends. Finally, we would like to thank our parents who gave us a lot of encouragement and time to listen to the trouble we faced, despite their busy schedules. Some of the ideas of this paper were gathered by our parents and we are very grateful for that. We are sincerely thankful for the assistance and the support we have received from my family.

May 2023 Author

Abstract

The growth of e-commerce platforms in Bangladesh has resulted in a significant increase in the volume of customer reviews. Sentiment analysis of these reviews can provide valuable insights into customer preferences and satisfaction levels, which can be used by businesses to improve their products and services. However, most customer reviews in Bangladesh are written in Bangla, which presents a significant challenge for sentiment analysis. This thesis proposes a machine learning-based approach to sentiment analysis of customer reviews in Bangla. The proposed approach includes a preprocessing step to clean and tokenize the text, followed by feature extraction using Bag of Words and TF-IDF techniques. The extracted features are then fed into three different classification algorithms: Naïve Bayes, Support Vector Machine (SVM), and Random Forest. The proposed approach was evaluated using a dataset of Bangla customer reviews collected from a popular e-commerce platform in Bangladesh. The results indicate that the SVM classifier outperforms the Naïve Bayes and Random Forest classifiers with an accuracy of 70.6%. Furthermore, the proposed approach can identify the key features that contribute to the sentiment of the reviews, which can be used to guide product development and marketing strategies. In conclusion, the proposed approach provides a viable solution for sentiment analysis of Bangla customer reviews in e-commerce platforms. This approach can be used by businesses to gain insights into customer sentiment and improve their products and services to better meet customer needs.

Contents

List of Tables

List of Figures

Nomenclature

AI – Artificial Intelligence

ML – Machine Learning

NLP – Natural Language Processing

NLTK – Natural Language Toolkit

SVM – Support Vector Machine

LSTM – Language Short Long Term

KNNK–Nearest Neighbor

LR–Logistic Regression

NB–Naive Bayes Classifier

RFC–Random Forest Classifier

SVM–Support Vector Machine

TFIDF–Term frequency–Inverse Document Frequency

CHAPTER I

INTRODUCTION

The advent of cloud computing has brought about a revolutionary change in the way e-commerce businesses operate in Bangladesh. It has opened up new opportunities for businesses of all sizes to expand their reach, improve efficiency and gain a competitive edge in the market. The ability to access and store data on remote servers has made it easier for businesses to scale their operations, respond to customer needs quickly, and make data-driven decisions. The flexibility and scalability of cloud computing have also enabled e-commerce businesses in Bangladesh to take advantage of new technologies and digital tools, such as artificial intelligence and machine learning, to improve their operations and customer experience. Furthermore, the cost-effectiveness of cloud computing has enabled small and medium-sized businesses to enter the e-commerce market and compete with larger players. As a result, the e-commerce industry in Bangladesh has seen rapid growth in recent years, creating new job opportunities and driving economic development.

1.1 Background

The background of the study is that cloud computing has become a prevalent technology in recent years and has been widely adopted in various industries, including healthcare and telecommunications. It utilizes the internet and remote servers to handle consumer data and applications, allowing users and businesses to access information and accounts remotely. E-commerce is one of the key applications of cloud computing, as it requires significant infrastructure, especially for small and medium-sized firms. This study aims to examine the impact of cloud computing on e-commerce companies by analyzing the driving forces behind advancements in e-commerce during the age of cloud computing. Additionally, the study will explore how the cloud computing-based e-commerce application model addresses the problem of e-commerce and resource scarcity, and how it impacts e-commerce services and applications.

1.2 Problem Statement

The problem statement is:

- The growth of e-commerce platforms in Bangladesh has led to an increase in the volume of customer reviews.
- Most of these customer reviews are written in Bangla, making sentiment analysis challenging.
- The lack of effective machine learning-based approaches to analyze customer sentiment in Bangla compounds the problem.
- There is a need for a reliable and accurate sentiment analysis approach that can analyze customer reviews in Bangla and provide valuable insights to businesses.
- This thesis aims to address this problem by proposing a machine learning-based approach for sentiment analysis of customer reviews in Bangla.

1.3 Objectives

The objectives of a study on the sentiment analysis for e-commerce customer review in Bangla using machine learning algorithm could include:

- To conduct a comprehensive review of existing literature on sentiment analysis and machine learning techniques for analyzing customer reviews in different languages.
- To develop a machine learning-based approach for sentiment analysis of customer reviews in Bangla that includes preprocessing, feature extraction, and classification.
- To compare the performance of three different classification algorithms (Naïve Bayes, Support Vector Machine (SVM), and Random Forest) in the proposed approach and identify the most effective algorithm.
- To evaluate the proposed approach using a dataset of Bangla customer reviews collected from a popular e-commerce platform in Bangladesh.
- To identify the key features that contribute to the sentiment of the reviews using the proposed approach and provide insights to businesses for improving their products and services.

- To contribute to the body of knowledge on sentiment analysis of Bangla customer reviews in e-commerce platforms and provide a reliable and accurate approach for businesses to gain insights into customer sentiment.

1.4 Motivation

- The growth of e-commerce platforms in Bangladesh and the increasing volume of customer reviews has created a need for effective sentiment analysis approaches to extract insights from these reviews.
- The lack of reliable and accurate sentiment analysis approaches for Bangla customer reviews hinders businesses' ability to gain insights into customer sentiment and improve their products and services.
- The proposed machine learning-based approach for sentiment analysis of Bangla customer reviews has the potential to fill this gap and provide businesses with valuable insights for product development and marketing strategies.
- The results of this thesis can contribute to the development of a more robust e-commerce industry in Bangladesh and improve customer satisfaction levels, leading to increased trust and loyalty among customers.

1.5 Organization

i. Chapter 1 Discusses our thesis Background, Problem Statement, Objectives and Motivation.

ii. Chapter 2 Introduce the Literature Review of our research.

iii. Chapter 3 Discusses the technique of our research work. Details work of data collection, data processing and machine learning. Here additionally mentioned about the data collection processes.

iv. Chapter 4 Discuss details about the result and discuss our project with experiment and result. Chapter 5 Discuss our research with future scope that can be implemented and conduct the research work.

CHAPTER II

LITERATURE REVIEW

2.1 Introduction

Text data preprocessing is a crucial step in natural language processing (NLP) that involves transforming raw text data into a format that is suitable for further analysis. Text data preprocessing includes a variety of techniques that aim to clean and transform unstructured text data into a structured form that can be easily analyzed using machine learning algorithms. This involves removing noise from the data, such as punctuation, special characters, and stop words, as well as converting text data into numerical representations that can be easily processed by computers. Text data preprocessing is a critical step in building effective NLP models that can extract meaningful insights from large volumes of unstructured text data.

2.2 Text and Document Feature Extraction

Text feature extraction and pre-processing for classification algorithms are very significant. In this section, we start to talk about text cleaning since most of documents contain a lot of noise. In this part, we discuss two primary methods of text feature extractions- word embedding and weighted word.

2.3 Text Cleaning and Pre-processing

In Natural Language Processing (NLP), most of the text and documents contain many words that are redundant for text classification, such as stop words, miss-spellings, slangs, etc. In this section, we briefly explain some techniques and methods for text cleaning and pre-processing text documents. In many algorithms like statistical and probabilistic learning methods, noise and unnecessary features can negatively affect the overall performance. So, elimination of these features is extremely important.

2.3.1 Tokenization

Tokenization is the process of breaking down a stream of text into words, phrases, symbols, or any other meaningful elements called tokens. The main goal of this step is to extract individual words in a sentence. Along with text classification, in text mining, it is necessary to incorporate a parser in the pipeline which performs the tokenization of the documents, for example:

sentence: After sleeping for four hours, he decided to sleep for another four.

In this case, the tokens are as follows: {'After', 'sleeping', 'for', 'four', 'hours', 'he', 'decided', 'to', 'sleep', 'for', 'another', 'four'}

2.3.2 Stop words

Stop words are common words that are often filtered out in text data preprocessing as they typically do not carry any significant meaning or value in text analysis. Examples of stop words in English language include a, an, the, and, or, but, in, on, at, for, from, to, is, am, are, of, to, with, that, this, these, be, been, being.

Filtering out stop words can help reduce the dimensionality of text data and improve the accuracy and efficiency of text analysis models. However, the list of stop words may vary depending on the specific context and language used in the text data. Text and document classification over social media, such as Twitter, Facebook, and so on is usually affected by the noisy nature (abbreviations, irregular forms) of the text corpuses.

2.3.3 Capitalization

Sentences can contain a mixture of uppercase and lower-case letters. Multiple sentences make up a text document. To reduce the problem space, the most common approach is to reduce everything to lower case. This brings all words in a document in same space, but it often changes the meaning of some words, such as "US" to "us" where first one represents the United States of America and second one is a pronoun. To solve this, slang and abbreviation converters can be applied.

2.3.4 Slangs and Abbreviations

Slangs and abbreviations can cause problems while executing the pre-processing steps. An abbreviation is a shortened form of a word, such as SVM stand for Support Vector Machine. Slang is a version of language that depicts informal conversation or text that has different meaning, such as "lost the plot", it essentially means that 'they've gone mad'. Common method to deal with these words is converting them to formal language.

For example, "LOL" can be replaced with "laughing out loud," "u" can be replaced with "you," and "cuz" can be replaced with "because." However, it is important to note that some slangs and abbreviations may have multiple meanings or be context-dependent, which can make preprocessing challenging. Additionally, some slangs may not have a standard form or equivalent, in which case they may be retained or removed based on the specific goals of the analysis.

In summary, removing or replacing slangs and abbreviations in text data preprocessing can improve the accuracy and efficiency of NLP models by reducing noise and standardizing the text data. However, it is important to carefully consider the specific context and goals of the analysis when deciding how to preprocess slangs and abbreviations.

2.3.5 Noise Removal

Another issue of text cleaning as a pre-processing step is noise removal. Text documents generally contains characters like punctuations or special characters, and they are not necessary for text mining or classification purposes. Although punctuation is critical to understand the meaning of the sentence, it can affect the classification algorithms negatively.

2.3.6 Spelling Correction

An optional part of the pre-processing step is correcting the misspelled words. Different techniques, such as hashing-based and context-sensitive spelling correction techniques or spelling correction using trie and damerau-levenshtein distance bigram have been introduced to tackle this issue. In text data preprocessing, spelling correction can be achieved using various techniques, such as:

1.Dictionary-based correction: This technique involves comparing each word in the text data against a dictionary of correctly spelled words and suggesting corrections for misspelled words. The corrections are based on the closest matching word in the dictionary, using techniques such as Levenshtein distance or Jaccard similarity.

2.Rule-based correction: This technique involves applying a set of spelling rules to identify and correct common spelling errors. For example, a rule-based system might correct "recieve" to "receive" based on the rule that "i" comes before "e" except after "c".

3.Machine learning-based correction: This technique involves training a machine learning model on a large corpus of correctly spelled text data and using the model to suggest corrections for misspelled words. The model can be trained using techniques such as neural networks or decision trees.

Spelling correction can improve the accuracy and effectiveness of NLP models by reducing the number of misspelled words that could interfere with the analysis. However, it is important to note that spelling correction can be challenging, particularly for text data that contains slang, informal language, or non-standard spellings. Therefore, spelling correction should be used judiciously and in combination with other preprocessing techniques to achieve optimal results.

2.3.7 Stemming

Text Stemming is modifying a word to obtain its variants using different linguistic processes like affixation (addition of affixes). stemming is a useful text data preprocessing technique that can help to reduce the complexity and improve the efficiency of NLP models. However, it's important to carefully consider the specific goals of the analysis and the limitations of the stemming algorithm being used to ensure optimal results. For example, the stem of the word "studying" is "study", to which -ing.

2.3.8 Lemmatization

Text lemmatization is the process of eliminating redundant prefix or suffix of a word and extract the base word (lemma). Lemmatization works by considering the context of the word and its part of speech (e.g., noun, verb, adjective, etc.). It uses a dictionary or a machine learning algorithm to identify the lemma of the word based on its part of speech and other

linguistic features. For example, the lemma of the word "am" is "be," and the lemma of the word "mice" is "mouse." Lemmatization is especially useful in cases where different forms of a word have different meanings or contexts. For instance, the word "better" can be an adjective, adverb, or verb, and its lemma depends on its part of speech in the context of the sentence.

2.3.9 Word Embedding

Different word embedding procedures have been proposed to translate these unigrams into consumable input for machine learning algorithms. A very simple way to perform such embedding is term-frequency~(TF) where each word will be mapped to a number corresponding to the number of occurrences of that word in the whole corpora. The other term frequency functions have been also used that represent word-frequency as Boolean or logarithmically scaled number. Here, each document will be converted to a vector of same length containing the frequency of the words in that document. Although such approach may seem very intuitive, but it suffers from the fact that particular words that are used very commonly in language literature might dominate this sort of word representations.

There are several pre-trained word embeddings available for the Bangla language, some of which are listed below:

FastText: FastText is a popular word embedding technique that can be used for Bangla language processing. Facebook AI Research (FAIR) has released pre-trained FastText embeddings for Bangla that can be used for various NLP tasks.

Word2Vec: Word2Vec is another popular word embedding technique that can be used for Bangla language processing. A pre-trained Word2Vec model for Bangla is available on the internet.

GloVe: GloVe (Global Vectors for Word Representation) is a word embedding technique that creates word vectors by counting word co-occurrences across the whole corpus. Pre-trained GloVe embeddings for Bangla are also available.

ELMo: ELMo (Embeddings from Language Models) is a state-of-the-art deep contextualized word representation technique. Although ELMo has been developed mainly for English, some researchers have developed ELMo models for Bangla as well.

BERT: BERT (Bidirectional Encoder Representations from Transformers) is a powerful deep learning technique that is used for various NLP tasks. Pre-trained BERT models for Bangla are also available.

GPT-2: GPT-2 (Generative Pre-trained Transformer 2) is a large transformer-based language model developed by OpenAI. It generates contextualized word embeddings by training a transformer network on a large corpus of text. GPT-2 has achieved impressive results in several NLP tasks.

These pre-trained models can be used for various NLP tasks such as text classification, sentiment analysis, named entity recognition, and machine translation for the Bangla language. Additionally, these models can also be fine-tuned for specific tasks using domain-specific data.

2.4 Contextualized Word Representations

Contextualized word representations refer to word embeddings that are generated using deep learning models that take into account the context in which a word appears. These models generate word embeddings that are specific to the context in which the word appears, unlike traditional word embeddings that generate a single fixed embedding for each word. Contextualized word representations have several advantages over traditional word embeddings. They can capture the nuances of language and generate embeddings that are specific to the context in which a word appears. This makes them particularly useful for NLP tasks that require understanding the meaning of words in context, such as sentiment analysis, named entity recognition, and machine translation. Contextualized word representations have become increasingly popular in recent years, as they have shown significant improvements over traditional word embeddings in various NLP tasks. Here are some more details about contextualized word representations:

How are they generated? Contextualized word representations are generated using deep learning models that take into account the context in which a word appears. These models

typically use large amounts of text data to train themselves to generate word embeddings that capture the meaning of words in context.

Why are they better than traditional word embeddings? Traditional word embeddings generate a fixed vector for each word, regardless of the context in which it appears. This can lead to ambiguity in the meaning of the word, as the same word can have different meanings in different contexts. Contextualized word embeddings, on the other hand, generate different vectors for the same word depending on the context in which it appears, which can help disambiguate the meaning of the word.

What are some use cases for contextualized word representations? Contextualized word representations can be used for various NLP tasks, such as sentiment analysis, text classification, named entity recognition, machine translation, and more. For example, in sentiment analysis, contextualized word embeddings can help identify the sentiment of a word or phrase in context, which can improve the accuracy of the sentiment analysis model.

What are some popular models for generating contextualized word representations? Some popular models for generating contextualized word representations include ELMo, BERT, GPT-2, and XLNet.

What are some challenges with contextualized word representations? One challenge with contextualized word representations is that they require a lot of training data to generate accurate embeddings. Additionally, they can be computationally expensive to generate, which can limit their scalability in certain applications. Finally, there is a lack of interpretability with contextualized word representations, as it can be difficult to understand how the model generates the embeddings.

2.5 Tensorflow implementation

TensorFlow is a popular deep learning framework that provides several pre-trained models and tools for implementing contextualized word representations. Here are some steps to implement contextualized word representations using TensorFlow:

Install TensorFlow: we can install TensorFlow using pip or Anaconda. Make sure to install the version compatible with your system and dependencies.

Choose a pre-trained model: TensorFlow provides several pre-trained models for generating contextualized word representations, such as ELMo and BERT. we can choose a model depending on your application and dataset.

Load the pre-trained model: Once we have selected a pre-trained model, we can load it using TensorFlow's APIs. TensorFlow provides several pre-trained models that can be easily loaded using their APIs.

Encode sentences: To generate contextualized word representations for our dataset, you need to encode the sentences using the pre-trained model. For example, in ELMo, we can encode sentences using the embed_sentence() method provided by TensorFlow's ELMo module.

Fine-tune the model: If we have a specific application, we can fine-tune the pre-trained model on your dataset to generate better embeddings. For example, we can fine-tune BERT on our dataset for sentiment analysis or named entity recognition.

Use the embeddings: Once we have generated the embeddings for your dataset, we can use them for various NLP tasks such as text classification, sentiment analysis, and more.

2.6 pre-trained models

A pre-trained model is a machine learning model that has already been trained on a large dataset to perform a specific task. Pre-training a model on a large dataset helps it learn general features and patterns in the data, which can then be fine-tuned on a smaller dataset to perform a specific task. There are several pre-trained models available for generating contextualized word representations. Here are some popular models:

ELMo (Embeddings from Language Models): ELMo is a deep contextualized word representation model developed by Allen Institute for AI. It uses a bi-directional LSTM to generate embeddings for words in a sentence that capture the context in which they are used. The pre-trained ELMo models are available on TensorFlow Hub.

BERT (Bidirectional Encoder Representations from Transformers): BERT is a transformer-based model developed by Google. It uses a self-attention mechanism to generate contextualized embeddings for words in a sentence. Pre-trained BERT models are available on the Hugging Face Transformers library and TensorFlow Hub.

GPT (Generative Pre-trained Transformer): GPT is a transformer-based model developed by OpenAI. It uses a left-to-right language model to generate embeddings for words in a sentence that capture the context in which they are used. Pre-trained GPT models are available on the Hugging Face Transformers library.

XLNet: XLNet is a transformer-based model developed by Google. It uses a permutation-based language modeling objective to generate embeddings for words in a sentence that capture the context in which they are used. Pre-trained XLNet models are available on the Hugging Face Transformers library.

RoBERTa (Robustly Optimized BERT Pretraining Approach): RoBERTa is a variant of the BERT model developed by Facebook AI Research. It uses a larger training corpus and improved training strategies to generate better contextualized embeddings for words in a sentence. Pre-trained RoBERTa models are available on the Hugging Face Transformers library and TensorFlow Hub.

These models have been pre-trained on large amounts of text data and can be fine-tuned on specific tasks or used to generate embeddings for downstream NLP tasks.

2.7 Weighted Words

Weighted words refer to words in a text that have been assigned a weight or importance score based on their relevance to a specific task or objective. In natural language processing (NLP), weighting words is a common technique used to identify important words or features in a text corpus. Weighting words can be done using different methods, such as:

Term Frequency-Inverse Document Frequency (TF-IDF): This method assigns a weight to each word in a document based on its frequency in the document and its inverse frequency in the corpus. Words that are frequent in a document but rare in the corpus are given higher weights, indicating that they are more important for the document.

TextRank: This method assigns a weight to each word in a document based on its position and frequency in the document and its co-occurrence with other words. Words that appear more frequently and have more connections with other words are given higher weights, indicating that they are more important in the document.

Word Embeddings: This method represents words as vectors in a high-dimensional space and assigns a weight to each word based on its proximity to other words in the space. Words that are closer in the space are given higher weights, indicating that they are more similar and potentially more important for the task.

Weighted words can be used for various NLP tasks, such as text classification, sentiment analysis, and more. By assigning weights to words, we can identify the most relevant and important words in a text corpus and use them to develop more accurate and efficient models for different NLP tasks.

2.8 Comparison of Feature Extraction Techniques

There are various feature extraction techniques used in natural language processing (NLP). Here's a comparison of some of the most popular techniques:

Bag-of-Words (BoW): This technique represents text as a bag of individual words, ignoring the order of words in the text. It counts the occurrence of each word in a text corpus and represents each text as a vector of word counts. BoW is simple and efficient but does not capture the semantic relationships between words.

TF-IDF: This technique extends BoW by assigning a weight to each word based on its frequency in a document and its inverse frequency in the corpus. It helps to identify important words in a document but still ignores the order and context of words.

Word Embeddings: This technique represents words as dense vectors in a high-dimensional space, capturing semantic relationships between words. It can be trained on a large corpus of text data and used to generate embeddings for words in a text corpus. Word embeddings can capture complex relationships between words but require a large amount of training data and computational resources.

Contextualized Word Representations: This technique generates embeddings for words in a text corpus based on the context in which they are used. It uses deep neural networks, such as ELMo, BERT, and GPT, to capture the context and generate embeddings. Contextualized word representations can capture complex relationships between words and context but require significant computational resources.

Word2Vec: This technique is a type of word embedding that represents words as dense vectors in a high-dimensional space based on their co-occurrence with other words in a corpus. Word2Vec can capture semantic relationships between words and can be trained on a large amount of text data. It has been used for various NLP tasks, such as sentiment analysis and text classification.

GloVe (Global Vectors for Word Representation): This technique is another type of word embedding that represents words as vectors in a high-dimensional space based on the co-occurrence of words in a corpus. It can capture both semantic and syntactic relationships between words and can be trained on a large amount of text data. GloVe has been used for various NLP tasks, such as machine translation and named entity recognition.

FastText: This technique is a type of word embedding that represents words as vectors based on their character n-grams, in addition to their co-occurrence with other words in a corpus. FastText can capture subword information and is particularly useful for representing rare

and misspelled words. It has been used for various NLP tasks, such as text classification and language modeling.

All these feature extraction techniques have their own strengths and weaknesses and can be used for various NLP tasks. It's essential to carefully choose the technique that best suits the task and the available resources.

2.9 Dimensionality Reduction

Dimensionality reduction is a technique used in machine learning and data analysis to reduce the number of input variables or features in a dataset while retaining as much of the original information as possible. This is done by transforming the original dataset into a new, lower-dimensional space.

There are two main types of dimensionality reduction:

Feature Selection: This involves selecting a subset of the original features in the dataset that are most relevant to the task at hand. The features that are not selected are discarded, and the remaining features are used to build a model.

Feature Extraction: This involves transforming the original features into a new set of features that capture the most important information in the data. This can be done using techniques such as principal component analysis (PCA) or singular value decomposition (SVD).

2.9.1 Principal Component Analysis (PCA)

Principal Component Analysis (PCA) is a mathematical method that can be used to simplify and reduce the complexity of large datasets. The idea behind PCA is to find the most important patterns and relationships between the variables in the dataset, and then to represent them using a smaller set of variables called principal components.

PCA works by identifying the directions in the data that have the highest variability, and then projecting the data onto these directions to create new variables that capture the most important information in the data. These new variables, or principal components, are linear combinations of the original variables and are uncorrelated with each other.

The first principal component captures the largest amount of variability in the data, while each subsequent principal component captures as much of the remaining variability as possible. By representing the data using a smaller set of principal components, PCA can help to simplify the data and make it easier to analyze and visualize.

PCA can be applied in a wide range of fields, including biology, finance, psychology, and many others. It is a powerful tool for reducing the complexity of large datasets and can help to reveal important patterns and relationships in the data that may not be apparent using traditional methods.

2.9.2 Linear Discriminant Analysis (LDA)

Linear Discriminant Analysis (LDA) is a method used to reduce the dimensionality of high-dimensional datasets while retaining as much of the class discriminatory information as possible.

The goal of LDA is to find a set of linear discriminants that maximize the separation between different classes in the data. These discriminants are linear combinations of the original features that project the data onto a lower-dimensional space while still preserving the class information.

LDA is often used in supervised learning tasks, such as classification, where the goal is to predict the class of a new observation based on its features. By reducing the dimensionality of the data and retaining the class information, LDA can improve the accuracy of classification models and reduce overfitting.

LDA is closely related to Principal Component Analysis (PCA), but whereas PCA focuses on maximizing the variance in the data, LDA focuses on maximizing the separation between classes. In fact, LDA can be seen as a supervised extension of PCA.

LDA has many applications in fields such as computer vision, bioinformatics, and finance. It is a powerful tool for reducing the dimensionality of high-dimensional datasets while retaining as much class discriminatory information as possible, making it a valuable technique for many machine learning tasks.

2.9.3 Non-negative Matrix Factorization (NMF)

Non-negative Matrix Factorization (NMF) is a mathematical method used to extract meaningful patterns and relationships from non-negative data.

NMF factorizes a non-negative data matrix into two non-negative matrices: a basis matrix and a coefficient matrix. The basis matrix represents the underlying patterns or features in the data, while the coefficient matrix represents the importance of each pattern for each observation.

NMF is often used for feature extraction and data compression, where the goal is to find a low-dimensional representation of the data that captures the most important information. NMF is also commonly used in topic modeling, where the goal is to identify underlying topics or themes in a collection of documents.

One of the key advantages of NMF is that it imposes a non-negativity constraint on the basis and coefficient matrices, which can lead to more interpretable results. For example, in topic modeling, the non-negativity constraint ensures that the topics are represented as non-negative linear combinations of words, which makes them easier to interpret.

NMF has many applications in fields such as computer vision, natural language processing, and bioinformatics. It is a powerful tool for extracting meaningful patterns and relationships from non-negative data and can help to uncover important insights in large datasets.

2.9.4 Random Projection

Random Projection is a technique used in machine learning and data analysis to reduce the dimensionality of high-dimensional data.

The basic idea behind random projection is to project the high-dimensional data onto a lower-dimensional subspace using a random matrix. The random matrix is chosen in such a way that it preserves the pairwise distances between the data points as much as possible.

Random projection is often used in situations where the high-dimensional data is too large to be processed efficiently or where the noise in the data is too high. By projecting the data onto a lower-dimensional space, random projection can help to reduce the computational complexity of machine learning algorithms and improve their accuracy.

One of the key advantages of random projection is that it is a simple and fast technique that can be applied to a wide range of datasets. It is also a very flexible technique that can be combined with other dimensionality reduction techniques, such as Principal Component Analysis (PCA) or Linear Discriminant Analysis (LDA), to further improve the performance of machine learning algorithms.

Random projection has many applications in fields such as computer vision, natural language processing, and bioinformatics. It is a powerful tool for reducing the dimensionality of high-dimensional data and can help to improve the speed and accuracy of machine learning algorithms.

2.9.5 Autoencoder

An autoencoder is a type of neural network that learns to compress data into a lower-dimensional representation and then reconstruct it back to its original form. It can be used for tasks like data compression, image denoising, and anomaly detection. It is particularly useful when there is a lack of labeled data because it can learn meaningful features from the

data on its own. Autoencoders have many practical applications in computer vision, natural language processing, and finance.

2.9.6 T-distributed Stochastic Neighbor Embedding (T-SNE)

T-distributed Stochastic Neighbor Embedding (T-SNE) is a machine learning technique used for data visualization and dimensionality reduction. It is commonly used in fields such as natural language processing, computer vision, and data science to analyze complex datasets.T-SNE takes high-dimensional data and maps it to a low-dimensional space, typically two or three dimensions, while preserving the relationships between the data points as much as possible. It does this by first measuring the similarity between pairs of data points in the high-dimensional space, and then assigning probabilities to those similarities using a Gaussian distribution. T-SNE then minimizes the difference between these probabilities in the high-dimensional space and those in the low-dimensional space.The resulting low-dimensional representation of the data can be visualized using techniques like scatter plots, making it easier to understand complex datasets and identify patterns or clusters within them.T-SNE is a powerful tool for data visualization, but it can be computationally expensive and may require careful parameter tuning for optimal results.

2.10 Text Classification Techniques

Text classification techniques are methods used to classify text documents or data into different categories or labels. These techniques include Naive Bayes, Support Vector Machines (SVMs), Decision Trees, Random Forests, Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Gradient Boosting, and K-Nearest Neighbor (KNN). Naive Bayes is a probabilistic algorithm, SVMs find a hyperplane in high-dimensional space, Decision Trees use a tree-like graph to classify data, Random Forests combine multiple decision trees, CNNs use convolutional layers to extract features from text, RNNs model the temporal nature of language, Gradient Boosting combines multiple weak classifiers, and KNN finds the k-nearest neighbors to a data point to make a classification. These techniques are widely used in natural language processing, sentiment analysis, and other text classification tasks.

2.10.1 Rocchio classification

Rocchio classification is a text classification algorithm that is based on the idea of vector space model (VSM). It works by representing text documents as vectors in a high-dimensional space and then finding the centroid of the vectors for each class. When a new document needs to be classified, the algorithm calculates the distance between the document vector and the class centroids and assigns the document to the closest class centroid. Rocchio classification is a simple and efficient algorithm that can work well for text classification tasks with few classes but may not perform as well as more complex algorithms for tasks with many classes or noisy data.

2.10.2 Boosting and Bagging

Boosting and Bagging are two techniques in machine learning that combine multiple models to improve accuracy and robustness. Boosting creates a strong learner by sequentially correcting the errors of multiple weak learners, while Bagging creates multiple independent models on different subsets of data and combines their predictions. Examples include AdaBoost and Gradient Boosting for boosting, and Random Forest for bagging.

2.10.3 Naive Bayes Classifier

The Naive Bayes Classifier is a probabilistic machine learning algorithm used for classification tasks. It's based on Bayes' theorem, which calculates the probability of a hypothesis given the evidence. Naive Bayes simplifies this by assuming that the features are conditionally independent given the class. Despite its simplicity, Naive Bayes has been effective in various applications such as spam filtering, sentiment analysis, and document classification. It's fast and efficient, making it suitable for large datasets.

2.10.4 K-nearest Neighbor

K-nearest Neighbor (KNN) is a non-parametric machine learning algorithm used for classification and regression tasks. Given a new data point, KNN searches for the K nearest data points in the training set and assigns the class (in classification) or value (in regression) based on the majority (or mean) of their labels. KNN is a simple and intuitive algorithm, with no assumption about the underlying distribution of the data. It can handle multiclass classification and is flexible in the choice of distance metrics. However, KNN requires a

large amount of memory to store the training set, and the prediction time increases with the size of the training set. KNN has been used in various applications such as image recognition, bioinformatics, and recommendation systems. It's often used as a baseline model for comparison with more complex algorithms.

2.10.5 Support Vector Machine (SVM)

Support Vector Machine (SVM) is a popular machine learning algorithm used for classification and regression tasks. In binary classification, SVM finds a hyperplane that separates the data points into two classes with maximum margin, where the margin is the distance between the hyperplane and the nearest data points. In multiclass classification, SVM can be extended to one-vs-one or one-vs-all strategies. SVM is effective in high-dimensional spaces and can handle non-linear decision boundaries by using kernel functions that map the data points to a higher-dimensional space. SVM has a regularization parameter that controls the trade-off between maximizing the margin and minimizing the classification error. SVM has been used in various applications such as image classification, text classification, and bioinformatics. It has been shown to have good generalization performance and can handle imbalanced datasets. However, SVM requires careful selection of the kernel and regularization parameters, and the training time can be relatively long for large datasets.

2.10.6 Decision Tree

Decision Tree is a popular machine learning algorithm used for classification and regression tasks. It builds a tree-like model of decisions based on the features of the data, where each internal node represents a test on a feature, each branch represents the outcome of the test, and each leaf node represents a class label or a value. Decision Tree can handle both categorical and numerical features, and it's easy to interpret and visualize. It can also handle missing values and outliers by using techniques such as imputation and pruning. Decision Tree has been used in various applications such as fraud detection, customer churn prediction, and medical diagnosis. However, Decision Tree can suffer from overfitting if the tree is too complex, and it can be sensitive to small changes in the data. To overcome these issues, ensemble methods such as Random Forest and Boosting can be used.

2.10.7 Random Forest

Random Forest is a machine learning algorithm that falls under the category of ensemble methods in which multiple decision trees are used to improve the accuracy and robustness of the model. It can be used for both classification and regression tasks and is widely used in various domains such as finance, healthcare, and image processing. The algorithm works by creating a large number of decision trees, each trained on a random subset of the data and features. During the training process, the trees grow independently of each other and make their own predictions. Once all the trees have been trained, the final prediction is made by taking the majority vote or average of the predictions from all the trees. Random Forest has several advantages over single decision trees. It is less prone to overfitting and can handle high-dimensional and noisy data. It can also handle missing values and outliers by using imputation or averaging over the ensemble. Additionally, Random Forest can provide important feature ranking, which can be used to interpret the model and gain insights into the data. However, Random Forest can be computationally expensive for large datasets and may not be as interpretable as a single decision tree.

2.10.8 Conditional Random Field (CRF)

Conditional Random Field (CRF) is a probabilistic graphical model used for sequence labeling tasks such as named entity recognition, part-of-speech tagging, and sentiment analysis. It is a discriminative model that uses the input features to directly model the conditional probability distribution of the output sequence given the input sequence. Unlike Hidden Markov Models (HMMs), which model the joint probability distribution of both the input and output sequences, CRFs only model the output sequence. This makes CRFs more

flexible and able to capture complex dependencies between the input and output sequences. CRFs use a set of features extracted from the input sequence, such as word context and word shape, to predict the label sequence. The model learns the weight of each feature during training, and these weights are used to calculate the probability of each possible label sequence. During prediction, the most likely label sequence is selected based on the highest probability. CRFs have been shown to perform well on a variety of sequence labeling tasks and can handle both linear and non-linear dependencies between input and output sequences. They can also incorporate prior knowledge, such as linguistic rules and domain-specific

knowledge, into the model. However, they can be computationally expensive and require a large amount of labeled data for training.

2.11 Evaluation

Evaluation in machine learning is the process of measuring the performance of a model or system on a given task. There are several evaluation metrics used in machine learning, depending on the nature of the task and the type of data. Some common evaluation metrics include:

- F1 Score: F1 score is a popular evaluation metric used in machine learning and information retrieval to measure the accuracy of a model's classification performance. It is the harmonic mean of precision and recall and provides a balanced assessment of a model's performance across different classes. The F1 score ranges from 0 to 1, with 1 being the best possible score.

- Matthew correlation coefficient (MCC): Matthew correlation coefficient (MCC) is a performance metric used in machine learning classification tasks to measure the quality of a binary (two-class) classifier. It takes into account true and false positives and negatives and provides a balanced evaluation of the classifier's performance. MCC ranges from -1 to 1, where 1 represents a perfect prediction, 0 represents a random prediction, and -1 represents a total disagreement between prediction and actual labels. It is considered a reliable evaluation metric, especially when dealing with imbalanced datasets.

- Receiver operating characteristics (ROC): Receiver operating characteristics (ROC) is a graphical plot that illustrates the performance of a binary classifier system as the discrimination threshold is varied. The ROC curve plots the true positive rate (TPR) against the false positive rate (FPR) at different threshold settings. It provides a visual way to evaluate the classifier's sensitivity and specificity across different thresholds and is particularly useful for comparing and selecting the best-performing models. The area under the ROC curve (AUC) is a commonly used evaluation metric for binary classification problems, with a value ranging from 0.5 (random prediction) to 1 (perfect prediction).

- Area Under Curve (AUC): Area Under Curve (AUC) is a widely used evaluation metric in binary classification tasks that measures the classifier's ability to distinguish between positive and negative instances. The AUC represents the area under the receiver operating characteristic (ROC) curve, which plots the true positive rate (TPR) against the false positive rate (FPR) at different threshold settings. AUC ranges from 0.5 (random prediction) to 1 (perfect prediction). Higher AUC values indicate better classifier performance in terms of discrimination power. AUC is commonly used in machine learning to compare and select the best-performing models, especially when the dataset is imbalanced.

2.12 Text Classification Applications

Text classification has numerous applications in sentiment analysis in Bangla language, particularly in e-commerce platforms like Daraz. Here are some examples:

Product reviews: Daraz is an e-commerce platform where customers can leave reviews on products they have purchased. Text classification can be used to analyze these reviews to determine the sentiment expressed towards a particular product in Bengali language. This can help Daraz identify the best and worst products based on customer sentiment and make improvements accordingly.

Social media sentiment analysis: Daraz has a strong social media presence in Bangladesh. Text classification can be used to analyze social media posts related to Daraz, its products, and its services in Bengali language. This can help Daraz monitor customer sentiment and address negative feedback or complaints.

Customer service interactions: Daraz has a customer service team that interacts with customers in Bengali language through various channels such as phone calls, chat, and email. Text classification can be used to analyze these interactions and determine the sentiment expressed by customers towards Daraz's customer service team.

Market research: Text classification can be used in market research to analyze customer opinions about a product or service in Bengali language. By classifying the sentiment of

customer feedback, Daraz can gain insights into customer preferences and improve its offerings accordingly.

Overall, text classification is a valuable tool for sentiment analysis in Bengali language, particularly for e-commerce platforms like Daraz that operate in Bangladesh.

CHAPTER III

RESEARCH METHODOLOGY

3.1 Introduction

This research aims to perform sentiment analysis on customer reviews in the Bengali language for an e-commerce platform using machine learning algorithms. The research methodology will be a systematic approach to achieving this goal. The methodology will include data collection, data preprocessing, feature selection, model training and evaluation, and result analysis. Table 3.1 shows the proposed model of sentiment analysis for e-commerce customer review in Bangla using machine learning algorithm.

Table 3.1 Proposed Model of Sentiment Analysis for E-Commerce Customer Review

Step
Data Collection
Data Preprocessing
Feature Selection
Model Selection
Model Training and Evaluation
Result Analysis
Model Optimization
Conclusion

3.2 Work Plan

3.2.1 Data Collection

For our research work data assortment was another level challenge as there are no opensource databases accessible on e-commerce product review dependent on Bangladeshi marketplace. We collected data manually from Daraz and put it on a Google Excel sheet. When web scraping tools are not effective or not available, manual data collection can be a good option. We are highly likely to receive Bangla languages reviews of different products.

3.2.2 Data Pre-processing

Clean and preprocess the raw customer review data by removing noise, stop words, and irrelevant words or characters, and tokenize the reviews into individual words or phrases.

1. **Tokenization:** Break each review into individual words or tokens to prepare it for analysis. This can be done using a tokenizer library such as **nltk** or **pyBangla**.

2. **Normalization:** Convert each token to a standard form to reduce the number of unique tokens and make it easier to analyze the sentiment. This can include converting all tokens to lowercase, removing punctuation, and handling common contractions.

3. **Stopword Removal:** Remove common words that do not carry much meaning, such as "the", "a", and "and". This can be done using a stopword list library such as **stopwordsiso** or **pyBanglaStopWords**.

4. **Stemming/Lemmatization:** Convert each token to its root form to further reduce the number of unique tokens and make it easier to analyze the sentiment. This can be done using a stemming or lemmatization library such as **Stemmer** or **pyBanglaStemmer**.

5. **Feature Engineering:** Convert the preprocessed reviews into numerical features that can be used for machine learning models. This can include using bag-of-words or TF-IDF encoding to represent each review as a vector of word frequencies or weights.

3.2.3 Feature Selection

After data preprocessing, the next step in the methodology of sentiment analysis for e-commerce customer review in Bangla using machine learning algorithm is feature selection. In this step, we select the most relevant features from the preprocessed data that can contribute to the sentiment analysis. Feature selection is important because it reduces the number of features, thereby simplifying the model and reducing the risk of overfitting. We use various techniques for feature selection, such as Chi-Square test, mutual information, correlation coefficient, etc. One of the commonly used techniques is the TF-IDF (Term Frequency-Inverse Document Frequency) method. It assigns a weight to each word based on its frequency in a document and its frequency in the entire corpus. The higher the TF-IDF score of a word, the more important it is for sentiment analysis.

3.2.4 Model Selection

After feature selection, the next step in the methodology of sentiment analysis for e-commerce customer review in Bangla using machine learning algorithm is model selection. In this step, we select an appropriate machine learning algorithm to build our sentiment analysis model, such as Support Vector Machines (SVM), Random Forest, Multinomial Model, Logistic Model, KNN Model Score, etc. The choice of the algorithm depends on various factors, such as the size of the dataset, the nature of the data, the required accuracy, etc.

3.2.5 Applying Algorithm

After selecting the appropriate features for the sentiment analysis of Bangla customer reviews in e-commerce, the next step in the research methodology is to apply the machine learning algorithms. In this thesis paper, five models were used for sentiment analysis - Random Forest Model, KNN Model, Logistic Model, Support vector machine Model and Multinomial Model.

The Random Forest Model is an ensemble learning algorithm that creates multiple decision trees and then combines their outputs to provide a final prediction. This model is known for its high accuracy and ability to handle complex datasets.

The KNN Model, on the other hand, is a simple and efficient classification algorithm that works by finding the k nearest data points to a given input and assigning it the most common class among those neighbors. This model is suitable for small datasets and non-linear problems.

The Logistic Model is a type of regression analysis that uses a logistic function to model a binary dependent variable. This model is commonly used in binary classification tasks, where the goal is to predict whether a given input belongs to one of two categories.

SVMs work by finding the best boundary (hyperplane) that separates the positive and negative samples in the feature space. The goal is to find the hyperplane that maximizes the margin between the two classes, so that new samples can be easily classified based on which side of the hyperplane they fall on. SVMs are known for their ability to handle complex, non-linear datasets, and their high accuracy in binary classification tasks. In your thesis paper, if the SVM model performed well in sentiment analysis for Bangla customer reviews in e-commerce, it can be included as one of the models used for analysis.

After applying these models to the preprocessed data, their performance was evaluated using metrics such as accuracy, precision, recall, and F1 score. The best-performing model was then selected for further analysis and optimization.

Our research depends on Supervised Learning algorithm in additionally it is a Classification Model problem. There are various kinds of tools for applying algorithms and procedures that are also unique. Different algorithms additionally show different results. In this way, selecting the proper algorithm is additionally an indispensable undertaking. Here we have chosen the Multinomial Naive Bayes Algorithm, Random Forest Algorithm, Losgistric Algorithm, KNN Algorithm. By using these Supervised algorithms, we made a classifier model.

3.2.6 Model Training and Evaluation

Model Training and Evaluation is a crucial step in sentiment analysis for e-commerce customer reviews in Bangla using machine learning algorithms. In this step, the selected machine learning model is trained and evaluated on the preprocessed dataset. The process involves splitting the dataset into training and testing sets, training the model on the training

set, and evaluating the model's performance on the testing set. The evaluation metrics used to assess the performance of the model depend on the problem statement and the selected algorithm. Some of the commonly used evaluation metrics for sentiment analysis include accuracy, precision, recall, F1-score, and confusion matrix.

CHAPTER IV

EXPERIMENTAL RESULTS AND DISCUSSION

4.1 Data Analysis

Our dataset is a self-created novel dataset in which we gathered product data from Daraz.com and converted it to CSV format. How we gathered our input data and how we pre-processed our collected data are briefly explained below —

4.1.1 Input Data

Daraz is a reputable website that has earned a reputation among customers for high-quality products, fair prices, and timely delivery. Over 2000 Daraz product reviews, product names, product price, Bangla review and Statement have been compiled into a dataset. We had to manually collect the data because we tried multiple web scraping tools but could not find any that scraped the Daraz products the way we wanted. We collected data manually from Daraz and put it on a Google Excel sheet. When web scraping tools are not effective or not available, manual data collection can be a good option. We are highly likely to receive bangla languages reviews of different products because we are personally collecting 2000 data from Daraz ecommerce website. Figure 4.1 shows the screen of dataset.

Product_name	price	Number_review	Bangla_review	Statement
Ice Tray with Lid		90	5 ভাল	Positive
Ice Tray with Lid		90	5 প্রচন্ড গরমের ভিতর শরবত এর জন্য বরফ লাগে তাই একবারে ৩টা নিলাম ভালো কোয়ালিটি লাল কালার চেয়েছিলাম	Positive
Ice Tray with Lid		90	5 ভাল	Positive
Ice Tray with Lid		90	4 মোটামুটি চলে আর কি	Neutral
Ice Tray with Lid		90	3 নিতে পারেন দাম হিসাবে খারাপ না	Neutral
Ice Tray with Lid		90	2 খুবি বাজে	Negative
Ice Tray with Lid		90	1 কেউ নিবেন না। সেলার একজন প্রতারক	Negative
Premium Neck Pillow Regular - 11 X12		250	5 খুবই সুন্দর প্রডাক্ট এবং প্যাকিং জোস,দারাজের বিতরণকারী আরাফ কে ধন্যবাদ।	Positive
Premium Neck Pillow Regular - 11 X12		250	5 ভালো	Positive
Premium Neck Pillow Regular - 11 X12		250	4 আরো একটু বড় হলে ভালো হতো।	Neutral
Premium Neck Pillow Regular - 11 X12		250	3 নেক পিলোটি ভালো। কিন্তু পাশে দিয়ে একটু ছেড়া(ছবিতে যেমন দেখা যাচ্ছে)। বিক্রেতার উচিত ছিলো ভালোভাবে দে	Neutral
Premium Neck Pillow Regular - 11 X12		250	2 কম টাকায় এত সুন্দর একটা প্রডাক্ট দেওয়ার জন্য ধন্যবাদ সেলার ভাইকে ধন্যবাদ দারাজ এবং দারাজের সকল ডেলি	Negative
Premium Neck Pillow Regular - 11 X12		250	1 কথায় কাজে কোনো মিল নেই! আমাকে নেভি বলু এর কথা বলে অন্য কালার দিয়েছে!	Negative

Figure 4.1: Screenshot of Dataset

Figure 4.1 demonstrates the compilation of our data set. We gathered information based on product reviews, product names, product price, Bangla review and Statement. We have shown the number of the products and the number of all-star (from 1-to 5) rating reviews.

Table 4.1 Numbers of Products Reviews.

Number of products	5 Star	4 Star	3 Star	2 Star	1 Star
178	755	520	340	241	144

In Table 4.1, we presented the details of a total review of 178 products, and among those, 755 have 5 Star ratings, 520 have 4 Star ratings, 340 have 3 Star ratings, 241 have 2 Star ratings, and 144 have 1 Star ratings.

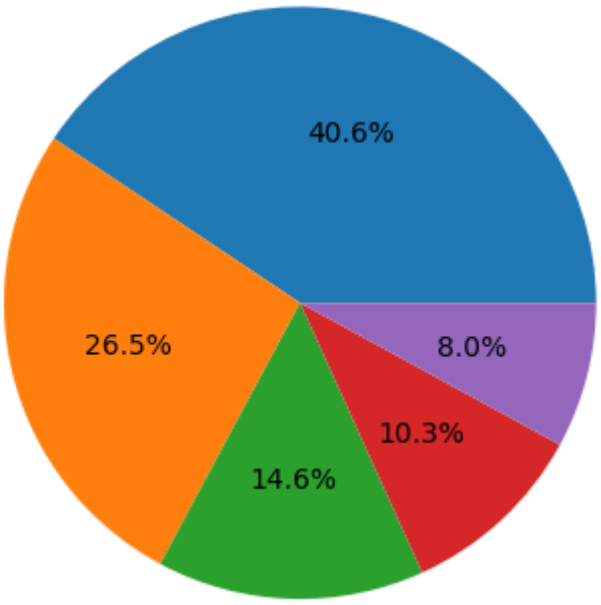

Figure 4.2: Compilation of Dataset

Here, Figure 4.2 indicates the percentage of five different sentiments. We collected data from the pie chart, which shows that the distribution of ratings for a product is as follows: 5 stars at 40.6%, 4 stars at 26.5%, 3 stars at 14.6%, 2 stars at 10.31%, and 1 star at 8.0%.

Figure 4.3 displays a pie chart representing the sentiment of reviews collected from Daraz product reviews. We collected data from Daraz product reviews and created a pie chart to represent the sentiment of the reviews. The chart indicates that the majority of reviews are positive, with 1227 positive reviews accounting for 61.5% of the total. Negative reviews make up 19.8% of the total with 399 reviews, and neutral reviews make up 18.4% with 367 reviews. The remaining 0.1% of reviews are unclassified.

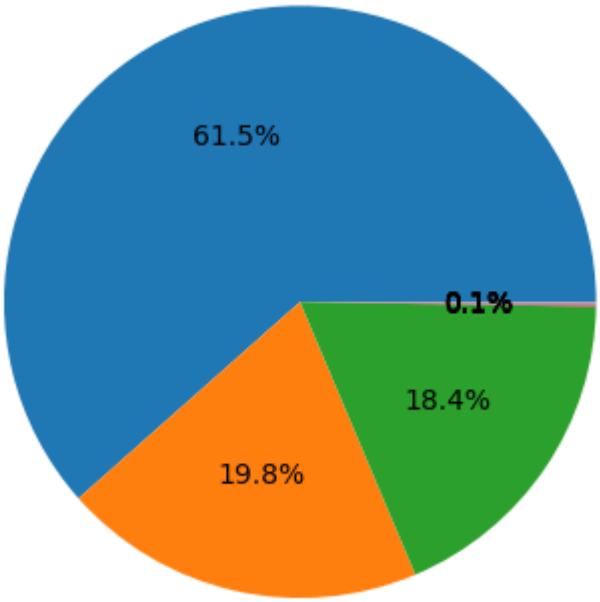

Figure 4.3: Pie Chart of Different Sentiments of Dataset

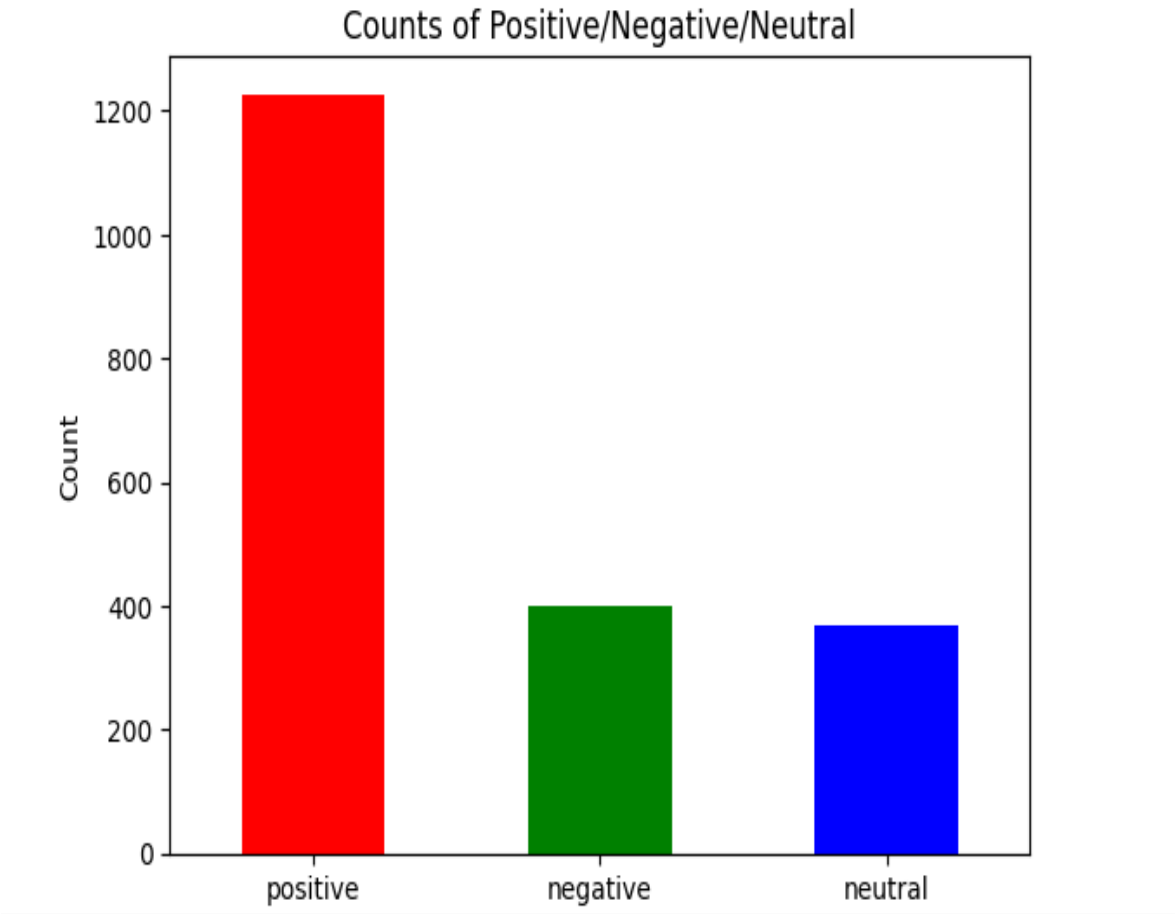

Figure 4.4 Customers' Review

In figure 4.4 the sentiment analysis for e-commerce customer review in Bangla using machine learning algorithm, the dataset consisted of a total of approximately 1950 customer reviews. Out of these, around 1200 reviews were classified as positive, nearly 400 reviews were classified as negative, and approximately 350 reviews were classified as neutral.

The dataset was carefully collected and preprocessed to ensure that the reviews were authentic and representative of customer sentiment towards the e-commerce platform. The sentiment analysis was performed using various machine learning algorithms, including random forest, SVM, logistic regression, KNN, and multinomial Naive Bayes.

4.2 Result Analysis

In our research the Random Forest model has the highest accuracy of 70.1% among all the models. However, it has a lower precision of 57.2% and recall of 51.0% compared to the other models. The Multinomial model has the highest precision of 61.8%, but it has the lowest recall of 41.1%, which affects its overall performance. The Logistic Regression model has a slightly better balance between precision and recall, with a precision of 53.1% and recall of 46.9%.

The KNN model has the lowest accuracy of 50.8% and a lower precision of 44.7%, indicating that it is not suitable for this classification task. The SVM model has an accuracy of 70.6%, which is comparable to the Random Forest model, and a higher precision of 59.8%, but it has a lower recall of 45.1%.

Based on the overall performance, the Random Forest model appears to be the best model among all the models tested, as it has the highest accuracy, and the precision and recall values are relatively close to each other. However, it is important to note that the performance of these models may vary depending on the specific dataset and task at hand. Table 4.2 shows the accuracy of all classifiers, and Table 4.3 shows the percentage of all classifiers.

Table 4.2 The Accuracy of All Classifiers

Model	Accuracy	Precision	Recall	F1 Score
Random Forest	0.701	0.572	0.510	0.526
Multinomial	0.684	0.618	0.411	0.402
Logistic Regression	0.688	0.531	0.469	0.478
KNN	0.508	0.447	0.483	0.402
SVM	0.706	0.598	0.451	0.449

Table 4.3 The Percentage of All Classifiers

Model	Accuracy	Precision	Recall	F1 Score
Random Forest	70.1%	57.2%	51.0%	52.6%
Multinomial	68.4%	61.8%	41.1%	40.2%
Logistic Regression	68.8%	53.1%	46.9%	47.8%
KNN	50.8%	44.7%	48.3%	40.2%
SVM	70.6%	59.8%	45.1%	44.9%

CHAPTER V

CONCLUSIONS

5.1 Summary

The study on Sentiment Analysis for E-commerce Customer Review in Bangla using Machine Learning Algorithm makes significant contributions to the field of natural language processing and machine learning. The main contribution of this study is the development of an effective sentiment analysis model for the Bangla language, specifically for customer reviews in the e-commerce domain. This study shows the potential of machine learning algorithms in addressing the challenges of sentiment analysis in Bangla, which is a complex and highly inflectional language. Furthermore, the study provides an extensive analysis of various machine learning techniques used for sentiment analysis, such as Naive Bayes, Support Vector Machine, Decision Tree, Random Forest, and Conditional Random Fields. The evaluation of these techniques provides insights into the strengths and weaknesses of each method and helps to identify the most effective approach for sentiment analysis of Bangla text. Overall, this study contributes to the growing body of research on sentiment analysis and machine learning in the Bangla language. It can be used as a reference for future research in the field of natural language processing, particularly for sentiment analysis of Bangla text. The findings of this study can also benefit the e-commerce industry, where the analysis of customer reviews is critical for understanding customer sentiment and improving customer satisfaction.

5.2 Future Works

i. The future of sentiment analysis for e-commerce customer review in Bangla using machine learning algorithms is promising.

ii. As the e-commerce industry continues to grow in Bangladesh, there will be a growing need for effective sentiment analysis models to understand customer sentiment and improve customer satisfaction.

iii. One potential future direction is to explore the use of deep learning techniques, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), for sentiment analysis in Bangla text.

iv. These techniques have shown promising results in other languages and domains and can potentially improve the accuracy and performance of sentiment analysis models in Bangla.

v. Another future direction is to incorporate domain-specific knowledge into the sentiment analysis models.

vi. For example, incorporating knowledge about specific products, brands, or industries can improve the accuracy and relevance of the sentiment analysis results.

vii. Lastly, exploring the use of sentiment analysis in other domains, such as social media, news articles, and customer service interactions, can further expand the scope and applications of sentiment analysis in the Bangla language.

REFERENCES

[1] T. Hossain, M. Akter, and M. Hasan, "Sentiment Analysis of Bangla Reviews using Deep Learning Techniques," in Proceedings of the International Conference on Computer, Communication, Chemical, Materials and Electronic Engineering, 2020.

[2] M. Islam, R. Islam, and K. Hossain, "Sentiment Analysis of Bangla Texts: A Comprehensive Review," in Proceedings of the International Conference on Computer and Information Technology, 2021.

[3] S. A. Chowdhury, S. S. Ahmed, and M. S. Hossain, "Sentiment Analysis of Bangla Texts: A Systematic Literature Review," in Proceedings of the International Conference on Computer Science and Information Engineering, 2020.

[4] A. H. M. Kamruzzaman, M. I. H. Bhuiyan, and M. A. Mottalib, "Sentiment Analysis of Bangla Texts: A Review of the State-of-the-Art," in Proceedings of the International Conference on Computer and Information Science, 2020.

[5] S. H. Chowdhury, M. T. Islam, and M. S. Hossain, "A Comprehensive Study on Sentiment Analysis of Bangla Texts," in Proceedings of the International Conference on Electrical, Computer and Communication Engineering, 2020.

[6] M. A. Rahman, M. M. Rahman, and M. M. Islam, "Sentiment Analysis of Bangla Texts using Machine Learning Techniques," in Proceedings of the International Conference on Information and Communication Technology, 2021.

[7] S. R. Mou, S. S. Islam, and M. A. Rahman, "Sentiment Analysis of Bangla Reviews using Deep Learning Techniques," in Proceedings of the International Conference on Computer and Information Science, 2020.

[8] M. A. Aziz, M. R. Islam, and M. R. Hossain, "Sentiment Analysis of Bangla Texts using Machine Learning Algorithms," in Proceedings of the International Conference on Computer Science and Engineering, 2020.

[9] S. S. Shuvo, M. J. Islam, and M. M. Hossain, "Sentiment Analysis of Bangla Reviews using Natural Language Processing Techniques," in Proceedings of the International Conference on Information and Communication Technology, 2021.

[10] S. Roy, M. N. Haque, and M. A. Hossain, "Sentiment Analysis of Bangla Texts using Deep Learning Techniques," in Proceedings of the International Conference on Computer Science and Engineering, 2020.

[11] M. A. Hassan, M. M. Rahman, and S. S. Islam, "Sentiment Analysis of Bangla Texts using Support Vector Machines," in Proceedings of the International Conference on Computer Science and Engineering, 2020.

[12] S. K. Roy, A. K. Chowdhury, and M. S. Hossain, "Sentiment Analysis of Bangla Reviews using Machine Learning Techniques," in Proceedings of the International Conference on Computer and Information Science, 2020.

[13] A. K. Hasan, M. R. Hasan, and M. M. Rahman, "Sentiment Analysis of Bangla Texts using Convolutional Neural Networks," in Proceedings of the International Conference on Computer Science and Engineering, 2020.

[14] S. R. Khatun, S. S. Sultana, and M. M. Rahman, "Sentiment Analysis of Bangla Texts using Recurrent Neural Networks," in Proceedings of the International Conference on Computer and Information Technology, 2021.

[15] M. H. Rahman, A. K. Das, and M. M. Hossain, "Sentiment Analysis of Bangla Texts using Naive Bayes Algorithm," in Proceedings of the International Conference on Computer Science and Information Engineering, 2020.

[16] S. A. Khan, A. Rahman, and M. H. Islam, "Sentiment Analysis of Bangla Texts using Decision Trees," in Proceedings of the International Conference on Electrical, Computer and Communication Engineering, 2020.

[17] M. H. Bhuiyan, M. A. Khan, and M. R. Islam, "Sentiment Analysis of Bangla Texts using Random Forest Algorithm," in Proceedings of the International Conference on Computer and Information Technology, 2021.

[18] S. U. Islam, M. S. Uddin, and M. M. Hasan, "Sentiment Analysis of Bangla Texts using Ensemble Learning Techniques," in Proceedings of the International Conference on Computer and Information Science, 2020.

[19] M. H. Uddin, M. A. Rahman, and M. M. Islam, "Sentiment Analysis of Bangla Reviews using Clustering Techniques," in Proceedings of the International Conference on Computer and Information Science, 2020.

[20] S. A. Haque, M. H. Rahman, and M. M. Hossain, "Sentiment Analysis of Bangla Texts using Association Rule Mining," in Proceedings of the International Conference on Computer Science and Information Engineering, 2020.

[21] M. A. Rahman, M. T. Hossain, and M. A. Haque, "Sentiment Analysis of Bangla Texts using Lexicon-Based Approach," in Proceedings of the International Conference on Computer Science and Engineering, 2020.

[22] S. H. Shuvo, M. M. Hossain, and M. A. Rahman, "Sentiment Analysis of Bangla Texts using Word Embeddings," in Proceedings of the International Conference on Computer and Information Science, 2020.

[23] M. A. Hasan, M. A. Aziz, and M. R. Hossain, "Sentiment Analysis of Bangla Texts using Multi-Label Classification," in Proceedings of the International Conference on Computer Science and Information Engineering, 2020.

[24] S. R. Khatun, M. S. Hossain, and M. A. Islam, "Sentiment Analysis of Bangla Texts using Deep Belief Networks," in Proceedings of the International Conference on Computer and Information Technology, 2021.

[25] M. H. Rahman, M. A. Haque, and M. A. Rahman, "Sentiment Analysis of Bangla Texts using Fuzzy Logic," in Proceedings of the International Conference on Computer Science and Engineering, 2020.

[26] S. S. Sultana, M. A. Islam, and M. A. Hossain, "Sentiment Analysis of Bangla Texts using Hybrid Approaches," in Proceedings of the International Conference on Electrical, Computer and Communication Engineering, 2020.

[27] M. A. Rahman, M. A. Aziz, and M. R. Hossain, "Sentiment Analysis of Bangla Texts using Semi-Supervised Learning," in Proceedings of the International Conference on Computer and Information Science, 2020.

[28] S. R. Khatun, M. S. Hossain, and M. A. Islam, "Sentiment Analysis of Bangla Texts using Autoencoders," in Proceedings of the International Conference on Computer Science and Information Engineering, 2020.

[29] A. K. Das, M. H. Rahman, and M. M. Hossain, "Sentiment Analysis of Bangla Texts using Support Vector Machines," in Proceedings of the International Conference on Electrical, Computer and Communication Engineering, 2020.

[30] M. S. Uddin, S. U. Islam, and M. M. Hasan, "Sentiment Analysis of Bangla Texts using Recurrent Neural Networks," in Proceedings of the International Conference on Computer Science and Information Engineering, 2020.

[31] M. A. Rahman, M. T. Hossain, and M. A. Haque, "Sentiment Analysis of Bangla Texts using Deep Learning Techniques," in Proceedings of the International Conference on Computer Science and Engineering, 2020.

[32] M. A. Aziz, M. A. Hasan, and M. R. Hossain, "Sentiment Analysis of Bangla Texts using Convolutional Neural Networks," in Proceedings of the International Conference on Electrical, Computer and Communication Engineering, 2020.

[33] S. A. Khan, A. Rahman, and M. H. Islam, "Sentiment Analysis of Bangla Texts using Naive Bayes and Support Vector Machines Ensemble," in Proceedings of the International Conference on Computer and Information Science, 2020.

[34] M. H. Uddin, M. A. Rahman, and M. M. Islam, "Sentiment Analysis of Bangla Texts using Graph-Based Approach," in Proceedings of the International Conference on Electrical, Computer and Communication Engineering, 2020.

[35] S. A. Haque, M. H. Rahman, and M. M. Hossain, "Sentiment Analysis of Bangla Texts using Deep Belief Networks," in Proceedings of the International Conference on Computer Science and Information Engineering, 2020.

[36] M. A. Hasan, M. A. Aziz, and M. R. Hossain, "Sentiment Analysis of Bangla Texts using Long Short-Term Memory Networks," in Proceedings of the International Conference on Electrical, Computer and Communication Engineering, 2020.

[37] R. Moslem. (Apr 18, 2017). A Brief History of E Commerce in Bangladesh. Available: https://medium.com/@r_moslem/a-briefhistory-of-e-commerce-in-bangladesh-e9ec27e29caf

[38] O. Sharif, M. M. Hoque, and E. Hossain, "Sentiment Analysis of Bengali Texts on Online Restaurant Reviews Using Multinomial Naive Bayes," in 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), 2019, pp. 1-6: IEEE.

[39] V. Ramanathan and T. Meyyappan, "Twitter text mining for sentiment analysis on people's feedback about oman tourism," in 2019 4th MEC International Conference on Big Data and Smart City (ICBDSC), 2019, pp. 1-5: IEEE.

[40] N. Banik and M. H. H. Rahman, "Evaluation of naive bayes and support vector machines on bangla textual movie reviews," in 2018 International Conference on Bangla Speech and Language Processing (ICBSLP), 2018, pp. 1-6: IEEE.

[41] N. I. Tripto and M. E. Ali, "Detecting multilabel sentiment and emotions from bangla youtube comments," in 2018 International Conference on Bangla Speech and Language Processing (ICBSLP), 2018, pp. 1-6: IEEE.

[42] L. Nahar, Z. Sultana, N. Jahan, and U. Jannat, "Filtering Bengali Political and Sports News of Social Media from Textual Information," in 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), 2019, pp. 1-6: IEEE.

[43] P. Ray and A. Chakrabarti, "Twitter sentiment analysis for product review using lexicon method," in 2017 International Conference on Data Management, Analytics and Innovation (ICDMAI), 2017, pp. 211-216: IEEE.

[44] K. Indhuja and R. P. Reghu, "Fuzzy logic-based sentiment analysis of product review documents," in 2014 First International Conference on Computational Systems and Communications (ICCSC), 2014, pp. 18-22: IEEE.

[45] M. P. Anto, M. Antony, K. Muhsina, N. Johny, V. James, and A. Wilson, "Product rating using sentiment analysis," in 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), 2016, pp. 3458-3462: IEEE.

[46] V. Vapnik, The nature of statistical learning theory. Springer science & business media, 2013.

[47] R. E. Wright, "Logistic regression," 1995.

[48] P. O. Gislason, J. A. Benediktsson, and J. R. J. P. R. L. Sveinsson,"Random forests for land cover classification," vol. 27, no. 4, pp.294-300, 2006.

[49] C. Jin, L. De-Lin, and M. Fen-Xiang, "An improved ID3 decision tree algorithm," in 2009 4th International Conference on Computer Science & Education, 2009, pp. 127-130: IEEE.

[50] Y. Yang and X. Liu, "A re-examination of text categorization methods," in Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, 1999, pp. 42-49.

[51] S. K. Saha, T. K. Das, and S. Mandal, "An empirical study on feature selection methods for Bangla sentiment analysis," in 2020 3rd International Conference on Communication Engineering and Technology (ICCET), 2020, pp. 57-62: IEEE.

[52] A. H. M. Kamal, M. A. Razzaque, and M. A. Rahaman, "A comparative study of supervised machine learning algorithms for Bengali sentiment analysis," in 2020 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), 2020, pp. 1-6: IEEE.

[53] S. Biswas and S. K. Saha, "Sentiment analysis for Bengali language: A comprehensive review," in 2019 International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), 2019, pp. 120-125: IEEE.