A MINI PROJECT REPORT

SENTIMENTAL ANALYSIS: USING HYBRID CNN-Bi-LSTM FOR E-COMMERCE DATASET

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE
IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE FULFILLMENT OF COURSE SUBJECT “MINI PROJECT WITH SEMINAR I “IN SEMESTER II

OF

MASTER OF ENGINEERING
(DATA SCIENCE)

SUBMITTED BY

SAHIL SUDHIR PAWAR 2640

GUIDED BY
Dr. JAYADEVAN R

DEPARTMENT OF COMPUTER ENGINEERING

ARMY INSTITUTE OF TECHNOLOGY, PUNE

ALANDI ROAD, DIGHI, PUNE, PIMPRI-CHINCHWAD, MAHARASHTRA 411015
SAVITRIBAI PHULE PUNE UNIVERSITY
2023-2024

Declaration

I hereby declare that the work presented in this Project Report titled "Sentimental Analysis: using Hybrid CNN-Bi-LSTM for E-commerce dataset" submitted to the Army Institute of Technology Pune in fulfilment of the requirements for the Mini project with seminar- I of the degree of Master of Engineering in Data Science, is a bonafide record of the research work carried out under the supervision of Dr. Jayadevan R. The contents of this Project Report in full or in parts, have not been submitted to, and will not be submitted by me to, any other Institute or University in India or abroad for the award of any degree or diploma.

Signature
Sahil Sudhir Pawar
2640

CERTIFICATE

This is to certify that the project report entitles

"SENTIMENTAL ANALYSIS: USING HYBRID CNN-Bi-LSTM FOR E-COMMERCE DATASET"

Submitted by

Sahil Sudhir Pawar 2640

is a bonafide student of this institute and the work has been carried out by him/her under the supervision of **Dr. Jayadevan R** and it is approved for the partial fulfillment of the requirement of Savitribai Phule Pune University, for the fulfillment of course subject "Mini Project with Seminar I "in semester II of **Master of Engineering (Data Science).**

(Dr. Jayadevan R)
Guide,
Department of Computer Engineering

(Dr. Sunil Dhore)
Head,
Department of Computer Engineering

(Dr. P. B. Patil)
Principal,
Army Institute of Technology, Pune 411015

Place: Pune
Date:

Acknowledgement

It gives me a great pleasure and immense satisfaction to present this special topic project report on “Sentimental Analysis: using Hybrid CNN-Bi-LSTM for E-commerce dataset” which is the result of unwavering support, expert guidance, and focused direction of my guide Dr. Jayadevan R to whom I express my deep sense of gratitude and humble thanks, for his valuable guidance throughout the presentation work. The success of this seminar has throughout depended upon an exact blend of hard work and unending co-operation and guidance, extended to me by the superiors at our college.

Furthermore, I am indebted to Dr. Sunil Dhore, HOD computer and Dr P B Patil, Principal, whose constant encouragement, and motivation inspired me to do my best.

I would also like to sincerely thank my colleagues, the staff and all others who directly or indirectly helped us and made numerous suggestions which have surely improved the quality of my work.

Signature
Sahil Sudhir Pawar
2640

Abstract

In the digital era, e-commerce platforms have become integral to modern retail, revolutionizing the way consumers shop and interact with products and services. Understanding consumer sentiments within this realm is paramount for businesses to enhance customer experiences, tailor marketing strategies, and optimize product offerings. This study delves into sentiment analysis on a comprehensive e-commerce dataset, aiming to uncover the nuanced emotional responses of consumers towards various products, brands, and purchasing experiences.

Sentiment analysis techniques have evolved, with various machine learning classification models to hybrid models combining Convolutional Neural Networks (CNN) and Bidirectional Long Short-Term Memory networks (Bi-LSTM) showing promise in capturing both local and sequential features of textual data. This study proposes a novel approach employing a hybrid CNN-Bi-LSTM architecture for sentiment analysis on E-commerce datasets.

The proposed model integrates the strengths of CNNs in capturing local features and Bi-LSTMs in learning long-range dependencies, thus providing a comprehensive understanding of textual sentiment. We utilize a publicly available E-commerce dataset comprising product reviews, where each review is associated with a sentiment label.

The experimental results demonstrate the effectiveness of the hybrid CNN-Bi-LSTM model in sentiment classification tasks compared to individual CNN or Bi-LSTM models. The model achieves competitive performance metrics, including accuracy, precision, recall, and F1-score, showcasing its robustness in handling the complexities of E-commerce text data. We conducted extensive experiments to evaluate the model's performance under different hyperparameter settings and dataset variations. The results indicate the model's stability and generalization capabilities across diverse E-commerce domains and sentiment classes. The following work uses LSTM, Bi-LSTM, CNN and Hybrid CNN-Bi-LSTM model based on concept of neural network, in which improvement in accuracy of LSTM, Bi-LSTM, CNN and Hybrid CNN-Bi-LSTM models were achieved with the improvement from 79.19% to 90% ,90.19% to 95%,85.69% to 93% and 91.83% to 94.00% respectively.

Overall, our findings suggest that the proposed hybrid CNN-Bi-LSTM approach holds significant promise for sentiment analysis in E-commerce applications, providing valuable insights for businesses to enhance customer experience and make data-driven decisions.

Keywords: LSTM, Bi-LSTM, CNN, Hybrid CNN-Bi-LSTM.

TABLE OF CONTENTS

LIST OF FIGURES:

LIST OF TABLES:

Sentimental Analysis: using Hybrid CNN-Bi-LSTM for E-commerce dataset

1 Introduction

Product reviews and comments have emerged as invaluable sources of information that shape consumer perceptions, influence purchase decisions, and drive market trends. With the proliferation of e-commerce platforms and social media channels, consumers now have unprecedented opportunities to express their opinions, experiences, and sentiments regarding products and services they encounter online. These user-generated reviews and comments serve as rich repositories of qualitative data, offering candid insights into the likes, dislikes, and references of consumers worldwide.

Product reviews and comments encompass a wide range of textual content, including written reviews, star ratings, comments, and feedback left by customers on e-commerce websites, social media platforms, forums, and review aggregators. These expressions of consumer sentiment encapsulate a spectrum of emotions, ranging from delight and satisfaction to frustration and disappointment, providing a comprehensive reflection of the customer experience of the customer experience.

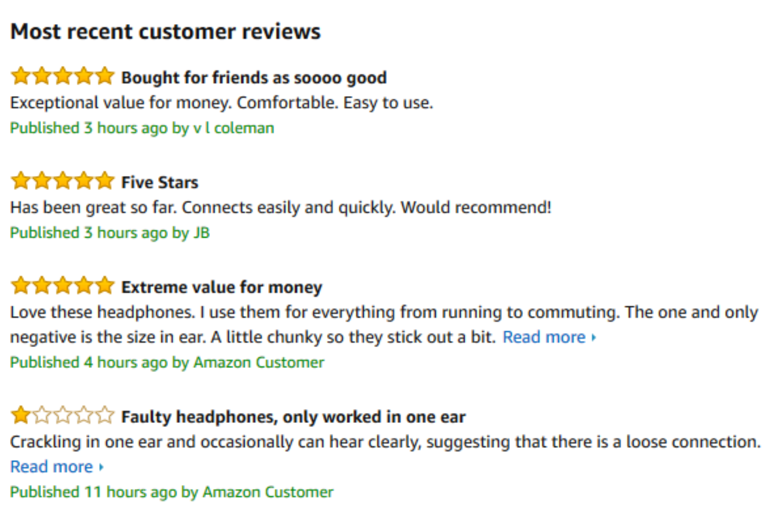

Figure 1.1: An image of reviews on online product (website: Amazon)

Sentiment analysis, also known as opinion mining, is a computational technique employed to extract, quantify, and analyze sentiments expressed in textual data. It involves the application of natural language processing (NLP) and machine learning algorithms to classify text into categories such as positive, negative, or neutral based on the underlying sentiment conveyed.

Product reviews and comments serve as prime candidates for sentiment analysis due to their abundance, accessibility, and inherent richness of emotional content. By harnessing the power of sentiment analysis, businesses and researchers can gain profound insights into consumer perceptions, preferences, and behaviors. This analytical approach enables them to uncover patterns, trends, and sentiments hidden within vast volumes of textual data, empowering informed decision-making and strategic planning.

Moreover, sentiment analysis holds immense practical significance across various domains, including marketing, customer service, product development, and market research. Businesses leverage sentiment analysis to monitor brand sentiment, evaluate product performance, identify areas for improvement, and devise targeted marketing campaigns tailored to consumer preferences.

The existing systems generally work upon basic classification algorithms such as logistic regression, Support Vector Machine (SVM), Naive Bayes classifier, Decision Trees, Artificial Neural network (ANN), etc. These algorithms learn from labeled training data and then predict the class labels of new, unseen instances. They are widely used in various fields such as sentiment analysis, image recognition, and spam detection. In the training process of a Hybrid CNN-Bi-LSTM model for E-commerce dataset, textual data undergoes a multi-step procedure to extract and comprehend sentiment.

Initially, the CNN component processes the text by convolving over word embeddings to capture local features, effectively identifying patterns within the data. These localized features are then passed to the Bi-LSTM layer, which utilizes its bidirectional architecture to learn long-range dependencies and sequential patterns in the text. During training, the model adjusts its parameters through backpropagation, optimizing the loss function to minimize prediction errors. The training dataset, comprising E-commerce product reviews annotated with sentiment labels, serves as the input for the model, enabling it to iteratively learn and refine its representations of sentiment. Through numerous epochs of training, the model gradually improves its ability to classify sentiment accurately, leveraging the complementary strengths of CNNs and Bi-LSTMs to achieve robust performance on E-commerce text data.

1.1 Problem definition and Objective

Sentiment analysis within E-commerce datasets presents a multifaceted challenge owing to the intricate nature of textual data and the imperative to accurately decipher nuanced sentiment expressions. Traditional sentiment analysis methods often struggle to interpret the complex language structures and context-specific sentiments inherent in E-commerce product reviews.

Figure1.2: Example of bad review image, good review image

These reviews can encompass diverse sentiments, ranging from positive endorsements to negative critiques, often embedded within lengthy and context-rich narratives. Consequently, there exists a pressing demand for advanced techniques capable of efficiently analyzing sentiment in E-commerce datasets to empower businesses with the ability to extract actionable insights and enhance customer satisfaction levels. The primary objective of this study is to develop and assess the efficacy of a Hybrid CNN-Bi-LSTM model tailored for sentiment analysis on E-commerce datasets. The model seeks to harness the unique capabilities of both Convolutional Neural Networks (CNN) and Bidirectional Long Short-Term Memory networks (Bi-LSTM) to effectively capture both local features and long-range dependencies within textual data. Specifically, the objectives can be delineated as follows:

Firstly, to engineer a hybrid architecture that seamlessly integrates CNNs and Bi-LSTMs to optimize the sentiment analysis process by adeptly discerning and exploiting both local and sequential features present in E-commerce product reviews. This amalgamation is envisaged to furnish the model with a robust framework capable of comprehensively capturing the diverse linguistic nuances embedded within E-commerce text data.

Secondly, to meticulously evaluate the performance of the proposed Hybrid CNN-Bi-LSTM model across various E-commerce datasets. This evaluation will encompass a comprehensive analysis of performance metrics such as accuracy, precision, recall, and F1-score, thereby endeavoring to achieve state-of-the-art results in sentiment classification tasks within the E-commerce domain. Furthermore, the model's resilience and generalization capabilities across disparate E-commerce domains and sentiment classes will be rigorously scrutinized to ascertain its pragmatic utility in real-world scenarios.

Through the attainment of these objectives, this study endeavors to contribute significantly to the advancement of sentiment analysis techniques tailored specifically for E-commerce datasets. By enabling businesses to derive nuanced insights from the wealth of textual data at their disposal, the proposed Hybrid CNN-Bi-LSTM model holds the potential to catalyze data-driven decision-making processes and foster enhanced levels of customer satisfaction within the dynamic landscape of E-commerce.

1.2 Contribution

Sentiment analysis within E-commerce datasets plays a pivotal role in shaping various aspects of the day-to-day world, particularly in the realm of consumer behavior, business strategies, and market dynamics. By accurately deciphering the sentiment embedded within E-commerce product reviews, sentiment analysis empowers businesses to gain profound insights into customer preferences, satisfaction levels, and pain points. This invaluable understanding enables companies to tailor their products, services, and marketing campaigns to better align with customer expectations, ultimately fostering stronger customer relationships and loyalty. Moreover, sentiment analysis facilitates the detection of emerging trends and sentiment shifts in real-time, enabling businesses to swiftly adapt their strategies to capitalize on opportunities or mitigate potential risks. From a consumer standpoint, sentiment analysis enhances the shopping experience by providing valuable information and social proof, enabling individuals to make more informed purchasing decisions. In essence, sentiment analysis within E-commerce datasets serves as a linchpin for driving customer-centricity, fostering innovation, and facilitating informed decision-making processes in the dynamic landscape of modern commerce.

The contribution of the project work is as follows, the proposed system leverages a hybrid CNN-Bi-LSTM model tailored for sentiment analysis, specifically targeting Amazon reviews. It preprocesses textual data, trains on a dataset comprising Amazon reviews, and evaluates performance metrics on a separate test set. With adjustable hyperparameters including optimizer, activation function, and layer configurations, the model fine-tunes its ability to discern sentiment in Amazon product reviews accurately. By integrating convolutional and recurrent neural network layers, it captures both local and sequential dependencies within textual data, enhancing its efficacy in sentiment classification. This approach offers a robust framework for analyzing sentiment in Amazon reviews, aiding businesses, and consumers alike in gauging product sentiment effectively and making informed decisions regarding purchases.

2 Literature Review

Sentiment analysis, also known as opinion mining, is a burgeoning field within natural language processing (NLP) that aims to extract subjective information from text, such as attitudes, emotions, and opinions. Over the past two decades, sentiment analysis has garnered significant attention from researchers due to its wide range of applications, including but not limited to product reviews, social media monitoring, market research, and political analysis

2.1 Sentimental Analysis Techniques

Various techniques have been developed to tackle sentiment analysis, ranging from traditional lexicon-based approaches to more advanced machine learning and deep learning models. Lexicon-based methods rely on sentiment lexicons or dictionaries containing lists of words annotated with their associated sentiment polarity (positive, negative, or neutral). Researchers like Liu et al. [1] and

Hu and Liu [2] have contributed extensively to this area, enhancing lexicon-based methods by incorporating linguistic rules and context-dependent sentiment analysis.

In recent years, supervised and unsupervised machine learning techniques have gained prominence for sentiment analysis tasks. Supervised learning algorithms such as Support Vector Machines (SVM), Naive Bayes, and Random Forest have been widely applied for sentiment classification [3]. On the other hand, unsupervised learning techniques like Latent Dirichlet Allocation (LDA) and K-means clustering have been utilized for sentiment polarity detection and topic-based sentiment analysis [4]. Furthermore, deep learning models, particularly Recurrent Neural Networks (RNNs), Convolutional Neural Networks (CNNs), and more recently, Transformers, have demonstrated state-of-the-art performance in sentiment analysis tasks [5] [6]. These models have the ability to capture complex patterns and dependencies in textual data, leading to improved sentiment classification accuracy.

2.2 Challenges and Future Directions

Despite the advancements in sentiment analysis techniques, several challenges persist. One major challenge is the inherent ambiguity and subjectivity of human language, which makes it difficult for algorithms to accurately interpret sentiment in context. Additionally, sentiment analysis often faces domain-specific challenges, requiring customized models and lexicons for different domains or languages [7]. Moreover, the evolution of language and the emergence of new linguistic phenomena, such as slang, sarcasm, and figurative language, pose significant challenges to existing sentiment analysis models [8]. Addressing these challenges will require interdisciplinary research efforts, incorporating insights from linguistics, psychology, and computer science.

Looking ahead, the future of sentiment analysis lies in the development of more robust and interpretable models that can handle nuanced aspects of sentiment, including aspect-based sentiment analysis, sarcasm detection, and emotion detection. Additionally, leveraging multimodal data sources, such as text, images, and audio, holds promise for enhancing the accuracy and depth of sentiment analysis systems [9]. In conclusion, sentiment analysis remains a vibrant area of research with wide-ranging implications for numerous industries and applications. Continued research efforts aimed at overcoming existing challenges and exploring new avenues will undoubtedly propel the field forward, enabling more sophisticated and nuanced analysis of human sentiment in textual data.

2.3 LSTM Models

LSTM networks, proposed by Hochreiter and Schmidhuber [9], address the vanishing gradient problem in traditional RNNs by introducing gating mechanisms to regulate information flow. This enables LSTMs to retain and propagate information over long sequences, making them particularly well-suited for tasks such as sentiment analysis, named entity recognition, and machine translation. Researchers have explored various modifications and extensions to basic LSTM architectures, including attention mechanisms [10] and memory augmentation techniques [11], to further enhance their performance in NLP tasks.

2.4 Bi-LSTM Models

Bi-directional LSTM models, introduced by Schuster and Paliwal [12], incorporate two LSTM layers processing input sequences in opposite directions. This allows the model to capture both past and future context information, enabling more effective sequence modeling. Bi-LSTM networks have demonstrated superior performance compared to uni-directional LSTMs in tasks such as part-of-speech tagging [13], named entity recognition [14], and sentiment analysis [15]. The bidirectional nature of Bi-LSTMs enables them to capture contextual information from both preceding and succeeding tokens, enhancing their ability to understand the semantics of input sequences.

2.5 Hybrid CNN-Bi-LSTM Models

Hybrid architectures combining CNNs with Bi-LSTM layers have gained popularity for various NLP tasks, leveraging the strengths of CNNs in feature extraction and local pattern recognition, and Bi-LSTM in capturing long-range dependencies. Kim [5] introduced one of the earliest hybrid models, which employed CNNs for sentence encoding followed by Bi-LSTM layers for sequence modeling. This architecture achieved state-of-the-art performance in sentence classification tasks, such as sentiment analysis and topic categorization.

Since then, numerous variations and extensions of hybrid CNN-Bi-LSTM models have been proposed for a wide range of NLP tasks. Zhang et al. [16] proposed a hierarchical attention-based hybrid model for document classification, combining CNNs for word-level feature extraction with Bi-LSTM layers for sentence-level modeling. Similarly, Zhou et al. [17] introduced a multi-channel CNN-Bi-LSTM architecture for aspect-level sentiment analysis, which utilized multiple sets of word embeddings to capture different linguistic aspects.

3 Related Work

Several researchers have delved into sentiment analysis within e-commerce datasets, aiming to understand customer opinions and sentiments towards products and services. For instance, Liu et al. [1] explored the integration of deep learning techniques, specifically convolutional neural networks (CNNs), for sentiment analysis in e-commerce reviews. Their study demonstrated the effectiveness of CNNs in capturing semantic information from review texts, achieving competitive performance in sentiment classification tasks.

Similarly, Kim [5] investigated the application of recurrent neural networks (RNNs) with long short-term memory (LSTM) units for sentiment analysis in e-commerce data. Their research highlighted the capability of RNNs to capture contextual dependencies in sequential data, leading to improved sentiment classification accuracy.

Moreover, Zhang et al. [16] proposed a novel approach based on attention mechanisms for sentiment analysis in e-commerce reviews. By incorporating attention mechanisms into the neural network architecture, their model effectively focused on informative parts of the input text, enhancing sentiment classification performance. These studies collectively illustrate the diverse methodologies, from CNNs to RNNs with attention mechanisms, employed by researchers to address sentiment analysis tasks within e-commerce datasets, showcasing advancements in the field.

4 Proposed Methodology

The proposed methodology for sentiment analysis in e-commerce datasets involves several key steps. Firstly, data collection encompasses gathering diverse sources such as product reviews, ratings, and social media mentions to ensure representativeness. Preprocessing involves cleaning and preparing the data through tasks like text normalization and removal of stopwords. Feature extraction techniques like Bag-of-Words and word embeddings transform text data into numerical representations suitable for analysis.

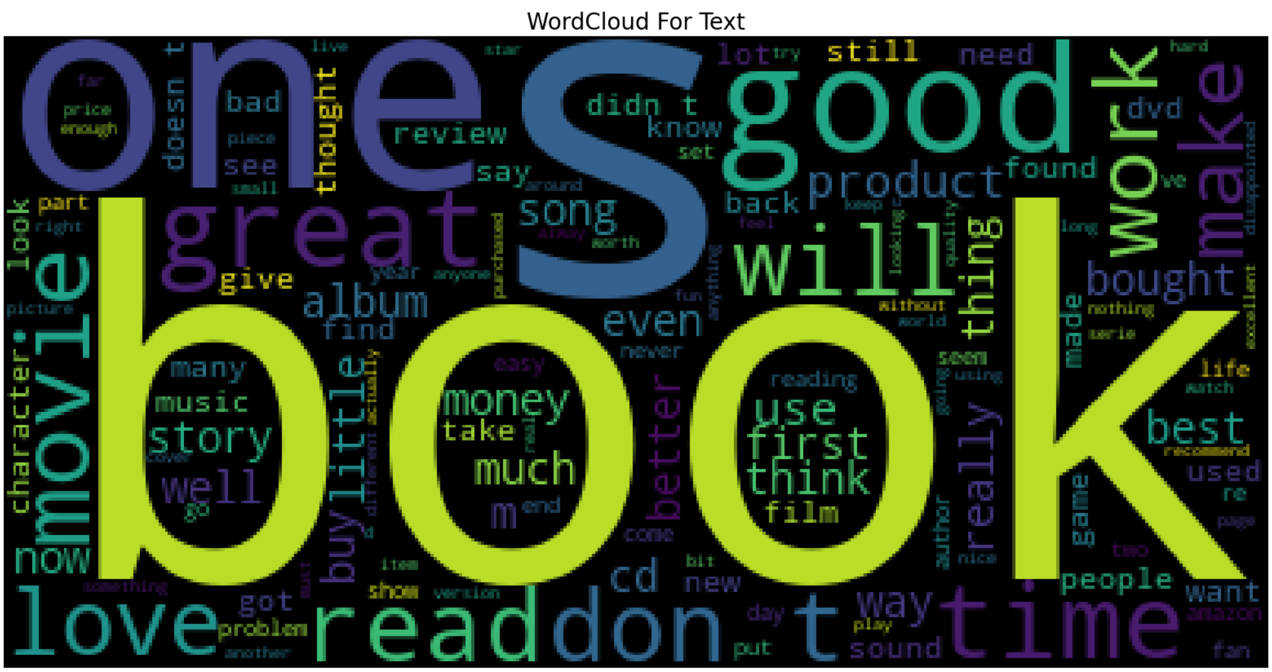

Figure 4.1: Visualization of dataset using WordCloud library

Model selection is crucial, with options ranging from traditional algorithms like SVM and Naive Bayes to deep learning architectures such as RNNs. By following this methodology, researchers and practitioners can systematically analyze sentiments within e-commerce datasets, enabling businesses to better understand customer perceptions and improve their offerings and services accordingly.

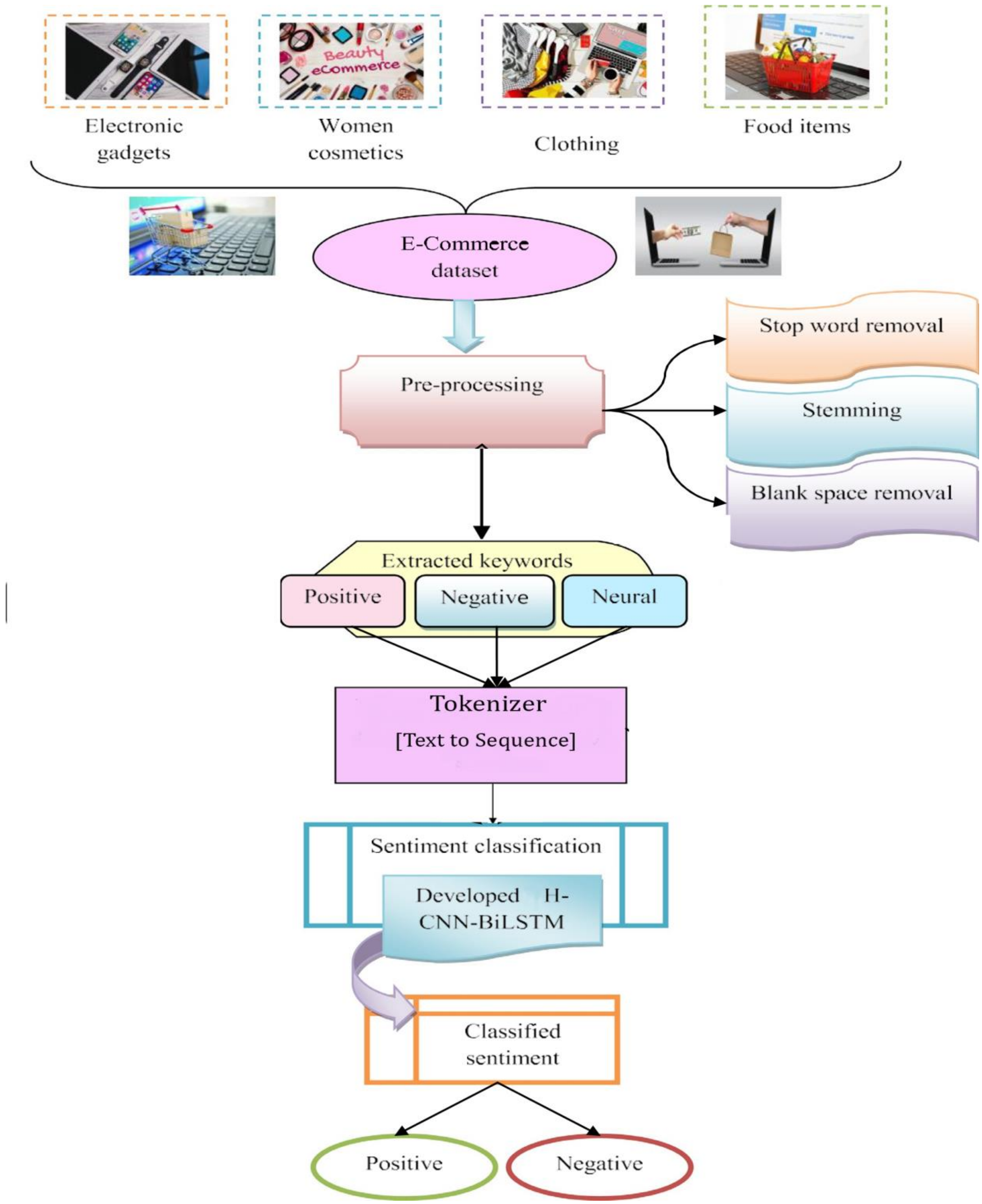

Figure 4.2: Model Architecture

This project aims to improve the accuracy of RNN based models i.e. LSTM, Bi-LSTM, Hybrid CNN-Bi-LSTM using amazon dataset with dimensionality of 2880000*2. Through preprocessing we clean the data by removing the stop words, stemming, and removing blank spaces and applying other transformations. By employing machine learning algorithms such as Logistic regression, Naïve Bayes classifier (Multinomial, Bernoulli, Gaussian), Decision Trees, LSTM, Bi-LSTM, and Hybrid CNN-Bi-LSTM we were able to achieve maximum accuracy of 95% using Bi-LSTM and 90% using LSTM algorithm.

4.1 Encoding and Conversion of .bz2 file to .csv file

One of the crucial steps involves decoding and converting .bz2 files to .csv files using UTF-8 encoding. .bz2 files are compressed files commonly used for storing large volumes of text data efficiently. However, to perform sentiment analysis effectively, it's necessary to extract and preprocess the data from these compressed files into a format that is compatible with analysis tools and models.

To begin, the .bz2 files containing the e-commerce dataset are first decompressed using appropriate decompression software or libraries. This process involves unpacking the compressed files to retrieve the original text data. Once decompressed, the text data within the .bz2 files can be accessed for further processing.

Next, the text data is decoded using the UTF-8 encoding scheme. UTF-8 is a widely used character encoding standard that supports a wide range of characters and symbols from various languages and scripts. Encoding the text data in UTF-8 ensures compatibility with different languages and prevents character encoding errors during processing.

After decoding, the text data is converted into a structured format such as .csv (Comma-Separated Values). .csv files are commonly used for storing tabular data in a plain text format, making them easy to manipulate and analyze using spreadsheet software or programming languages like Python.

During the conversion process, each piece of text data is typically organized into rows and columns, with each row representing a data sample (e.g., a product review) and each column representing a feature (e.g., review text, sentiment label, timestamp). This structured format facilitates data management, preprocessing, and analysis tasks for sentiment analysis.

Furthermore, encoding the .csv file in UTF-8 ensures that special characters, emojis, and symbols present in the text data are preserved accurately, preventing information loss or corruption. This is particularly important for e-commerce datasets, which may contain text data in multiple languages and character sets.

In the end, the decoding and converting .bz2 files to .csv files using UTF-8 encoding is an essential step in the proposed methodology for sentiment analysis within e-commerce datasets. This process enables the extraction, preprocessing, and structuring of the raw text data, laying the groundwork for subsequent analysis and model development.

4.2 Data Preprocessing and EDA

The initial step involves comprehensive data preprocessing and exploratory data analysis (EDA). The raw dataset undergoes several preprocessing operations to ensure the text data is clean and ready for analysis. Firstly, labels indicating sentiment are encoded to binary values, typically 0 for negative sentiment and 1 for positive sentiment. This step standardizes the sentiment labels for ease of processing and model training.

Following label encoding, the text data undergoes several text preprocessing steps to remove noise and irrelevant information. Stop words, which are common words that do not carry significant meaning for sentiment analysis, are removed to focus on meaningful content. Additionally, stemming is applied to reduce words to their root form, consolidating variations of words and improving the efficiency of downstream analysis. Blank spaces, punctuation marks, and special characters are removed to further clean the text data.

To ensure consistency and facilitate text processing, the text data is converted to lowercase. This step prevents duplicate representations of words due to variations in capitalization and ensures uniformity in the dataset. Moreover, rows containing missing values, indicated by "NaN" values, are dropped to maintain data integrity, and prevent potential biases in the analysis.

After preprocessing, exploratory data analysis (EDA) is conducted to gain insights into the distribution and characteristics of the dataset. Summary statistics such as the number of samples, class distribution, and average text length are calculated to understand the overall structure of the dataset. Visualization techniques such as histograms, pie charts, and word clouds are employed to explore the frequency distribution of sentiment labels and key words/phrases.

Furthermore, the relationships between sentiment labels and other features such as product categories, review ratings, and timestamps are examined through visualization and statistical analysis. This helps identify potential patterns and correlations in the data that can inform subsequent modeling decisions. Overall, data preprocessing and exploratory data analysis are essential steps in the proposed methodology for sentiment analysis within e-commerce datasets, laying the foundation for effective model development and interpretation of results.

4.3 Semantic word extraction

The process sentiment analysis methods are used to identify features for tweets. Generally, the feature engineering stage is done by the following steps such as removing stop words, stemming, transforming the data into the vector space, term weighting, and feature selection, and also tends to split the data into the phases of train and testing the data. The feature engineering process is applied to reduce the number of features, and it can reduce the complexity. Accordingly, the feature generation and selection process act as a significant role in building the results. Features generation is used to classify the sentiments. Feature extraction also helped to reduce the computation complexity. In this research work, semantic words are extracted by correlating the keywords using the obtained glossary through ‘‘https://in.mathworks.com/matlabcentral/fileexchange/5408-dictionary’’. Here, the keywords are assumed based on the extracted words. Moreover, semantic words are considered based on words that are highly closed to the keywords. For every semantic word, the similarity score is computed.

4.4 Vectorization

Vectorization plays a crucial role in transforming textual data into numerical representations that can be processed by machine learning models. Vectorization is the process of converting text documents into numerical vectors, where each vector represents a document or a word in a high-dimensional space. This step is essential as most machine learning algorithms require numerical inputs for training and prediction.

One common approach to vectorization is Bag-of-Words (BoW), where each document is represented as a vector of word counts. In this method, a vocabulary is created by extracting unique words from the corpus, and each document is represented by a vector indicating the frequency of each word in the vocabulary. BoW is simple and efficient but does not capture the semantic relationships between words. To address the limitations of BoW, Term Frequency-Inverse Document Frequency (TF-IDF) vectorization is often employed. TF-IDF assigns weights to words based on their frequency in the document and rarity across the corpus. Words that are common in a specific document but rare in the entire corpus are assigned higher weights, capturing their importance in conveying the document's meaning.

Another approach to vectorization is word embeddings, which represent words as dense, low-dimensional vectors learned from large text corpora using techniques such as Word2Vec, GloVe, or FastText. Word embeddings capture semantic relationships between words by placing similar words closer together in the vector space. This allows machine learning models to capture contextual information and generalize better to unseen data.

In addition to document-level vectorization, word-level vectorization is also important for sentiment analysis tasks. Words are typically represented as vectors using pre-trained word embeddings, and then aggregated to represent the entire document using techniques such as averaging or weighted averaging. This captures the overall sentiment expressed in the document while preserving the contextual information of individual words.

In e-commerce datasets where product reviews may contain domain-specific terms or jargon, domain-specific embeddings trained on relevant e-commerce or product review corpora can be beneficial. These embeddings are specialized to capture the unique language and semantics of e-commerce text data, improving the performance of sentiment analysis models on such datasets.

Thus, vectorization is a critical step in the proposed methodology for sentiment analysis within e-commerce datasets, enabling the transformation of raw text data into numerical representations that can be leveraged by machine learning models to predict sentiment labels accurately. By employing appropriate vectorization techniques, the model can effectively capture the semantic meaning and context of the text data, leading to more robust and interpretable sentiment analysis results.

4.5 Machine Learning Model Development

E-commerce technology has been developed and attracted more users for shopping through different E-commerce platforms. While comparing off-line shopping in physical stores, users in online shopping can shop anywhere at any time, which secures time and effort. Furthermore, the products on the online e-commerce platforms have satisfied the customers with different varieties and styles of products, so the customers can buy their favorite products without going to any shop. On the other hand, online shopping sometimes becomes an inconvenience for consumers owing to inconsistent information, poor quality of goods, etc. However, there are already certain methods for automatically obtaining the word vector features based on FastText and Glove. Moreover, the traditional machine learning method is yet required for extracting the emotional features present in the structured data. Here, the conventional machine learning model [26,27] is used for classifying the sentiment text features. It needs human intervention to categorize the sentiment in the text.

The recommended sentiment analysis model using E-commerce data undergoes several procedures like ‘‘pre-processing, key word extraction, semantic word extraction, and classification’. The E-commerce datasets are collected from online data sources passed through pre-processing. The pre-processing of these data is performed using blank space removal, stop word removal, and stemming for removing unwanted spaces, symbols, and repeated words that reduce the complexity of feature extraction. The stop word removal eliminates features like conjunctions and pronouns. The stemming removes the suffixes and prefixes of the given data. Then, the blank space removal is performed to eliminate the tab spaces. The obtained keywords from these tweets are pre-processed and categorized the keywords into neutral, positive, and negative. Again, certain features are extracted through the word-to-vector formation method. All these extracted features are integrated for subjecting it into the classification phase. The classification phase is taken by the hybrid algorithm named OH-CNN-BiLSTM, which is developed by optimizing certain parameters in CNN and Bi-LSTM using the developed IGSO for enhancing classification performance. The key objective of the designed sentiment analysis model-based E-commerce data is to improve the accuracy and sensitivity of the sentiment classification. The OH-CNN-BiLSTM-based sentiment classification provides the output as positive or negative tweets.

4.6 Classification using Logistic Regression

While applying the classification algorithm using logistic regression serves as a fundamental step in modeling sentiment prediction. Logistic regression is a widely used supervised learning algorithm suitable for binary classification tasks, such as predicting sentiment labels (positive or negative) based on textual data. To begin, the preprocessed text data, typically in the form of numerical vectors obtained through vectorization techniques, serves as the input features for the logistic regression model. These features represent the transformed text data, where each vector encapsulates the semantic information and characteristics of the corresponding document or word.

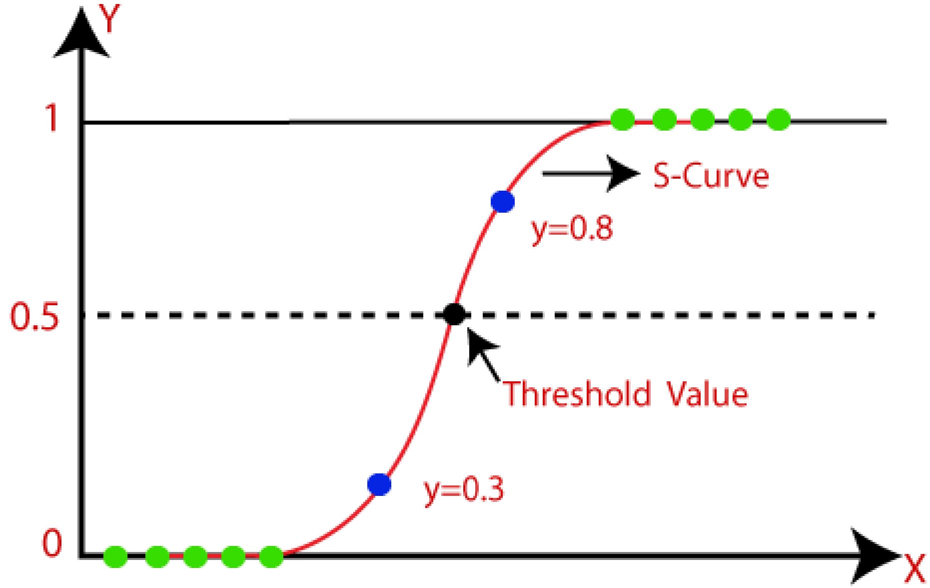

Figure 4.3: Logistic Regression

The sentiment labels associated with each data sample (e.g., product review) serve as the target variable for training the logistic regression model. In the case of binary sentiment classification, the labels are typically encoded as binary values, with one class representing positive sentiment and the other representing negative sentiment. During the training phase, the logistic regression model learns the relationship between the input features (text data) and the target variable (sentiment labels) by optimizing a cost function, such as the logistic loss function or cross-entropy loss function. The model adjusts its parameters (weights) iteratively through techniques such as gradient descent to minimize the prediction error and maximize the likelihood of correctly classifying the sentiment of the input text data.

Once trained, the logistic regression model can be used to predict sentiment labels for new, unseen text data. Given a new text sample, the model computes a probability score representing the likelihood of belonging to each sentiment class (positive or negative) using the learned parameters. The probability scores are then converted into binary predictions by applying a threshold, typically 0.5, where samples with probability scores above the threshold are classified as positive sentiment, and those below the threshold are classified as negative sentiment. Evaluation of the logistic regression model's performance is essential to assess its effectiveness in predicting sentiment accurately.

In conclusion, classification using logistic regression is a foundational component of the proposed methodology for sentiment analysis within e-commerce datasets. By leveraging logistic regression models trained on preprocessed text data, businesses can gain valuable insights into customer sentiments expressed in product reviews, enabling informed decision-making and targeted improvements to enhance customer satisfaction and loyalty.

4.7 Classification using Naïve Bayes Classifier

The Naive Bayes classifier is another essential approach for predicting sentiment labels based on textual data. Naive Bayes is a probabilistic classifier that applies Bayes' theorem with the "naive" assumption of independence among features, making it particularly suitable for text classification tasks. To begin, the preprocessed text data serves as the input features for the Naive Bayes classifier. These features are typically represented as numerical vectors obtained through vectorization techniques such as Bag-of-Words (BoW) or Term Frequency-Inverse Document Frequency (TF-IDF), where each vector encapsulates the frequency or importance of words in the corpus.

Similar to logistic regression, the sentiment labels associated with each data sample (e.g., product review) serve as the target variable for training the Naive Bayes classifier. In binary sentiment classification tasks, the labels are typically encoded as binary values, with one class representing positive sentiment and the other representing negative sentiment. During the training phase, the Naive Bayes classifier learns the conditional probability distributions of the input features (text data) given each sentiment class (positive or negative). This is achieved by estimating the probabilities of observing each word or feature in the training data for each sentiment class, assuming independence among features. Once trained, the Naive Bayes classifier can predict the sentiment label of new, unseen text data by applying Bayes' theorem to compute the posterior probability of each sentiment class given the input features. The class with the highest posterior probability is then assigned as the predicted sentiment label for the input text sample.

Naive Bayes classifier is a powerful approach in the proposed methodology for sentiment analysis within e-commerce datasets. By leveraging the probabilistic framework of Naive Bayes and the independence assumption among features, businesses can efficiently classify and analyze customer sentiments expressed in product reviews, leading to valuable insights and informed decision-making for improving products and services. -commerce

4.8 Classification using Decision Tree

Decision trees serve as a valuable technique for predicting sentiment labels based on textual data. Decision trees are non-parametric supervised learning models that recursively partition the feature space into regions based on the values of input features, ultimately leading to a hierarchical structure resembling a tree.

To begin, the preprocessed text data serves as the input features for the decision tree classifier. These features are typically represented as numerical vectors obtained through vectorization techniques such as Bag-of-Words (BoW) or TF-IDF, where each vector encapsulates the frequency or importance of words in the corpus.

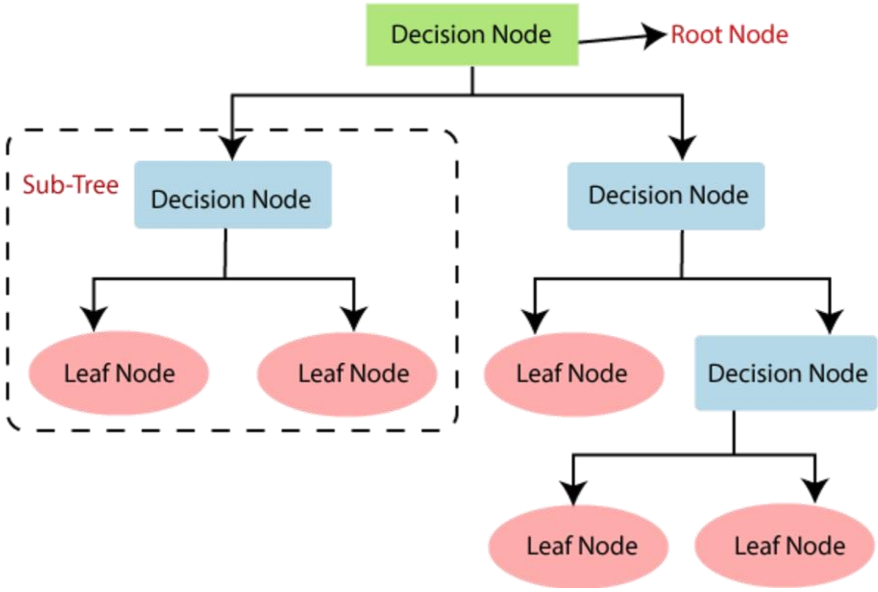

Figure 4.4: Decision tree

Like other classification algorithms, the sentiment labels associated with each data sample (e.g., product review) serve as the target variable for training the decision tree classifier. In binary sentiment classification tasks, the labels are typically encoded as binary values, with one class representing positive sentiment and the other representing negative sentiment. During the training phase, the decision tree classifier learns a sequence of if-else rules or conditions that recursively split the feature space into homogeneous regions with respect to the target variable (sentiment labels). The splitting criteria are determined based on metrics such as Gini impurity or entropy, which measure the impurity or disorder within each region. Once trained, the decision tree classifier can predict the sentiment label of new, unseen text data by traversing down the learned tree structure based on the values of input features. At each node, the decision tree evaluates the corresponding splitting criterion and proceeds to the child node that best satisfies the condition until a leaf node (representing a sentiment label) is reached.

4.9 Classification using LSTM (Long Short-Term Memory)

Employing Long Short-Term Memory (LSTM) networks stands as a formidable deep learning paradigm for predicting sentiment labels based on textual data. LSTMs, a variant of recurrent neural networks (RNNs), are designed to mitigate the vanishing gradient problem by incorporating a gating mechanism, allowing them to effectively capture long-range dependencies in sequential data. This property makes LSTMs particularly well-suited for analyzing text data, where understanding contextual relationships is pivotal.

At the outset, the preprocessed text data serves as the input sequences for the LSTM classifier. These sequences are typically encoded into numerical vectors via word embeddings, transforming each word or token into a high-dimensional vector representation. This step enables the model to comprehend the semantic nuances inherent in the text data. In alignment with standard classification paradigms, the sentiment labels assigned to each data instance (e.g., product review) act as the target variable for training the LSTM classifier. In the binary sentiment classification scenario, sentiments are encoded as binary values, with one class denoting positive sentiment and the other negative sentiment.

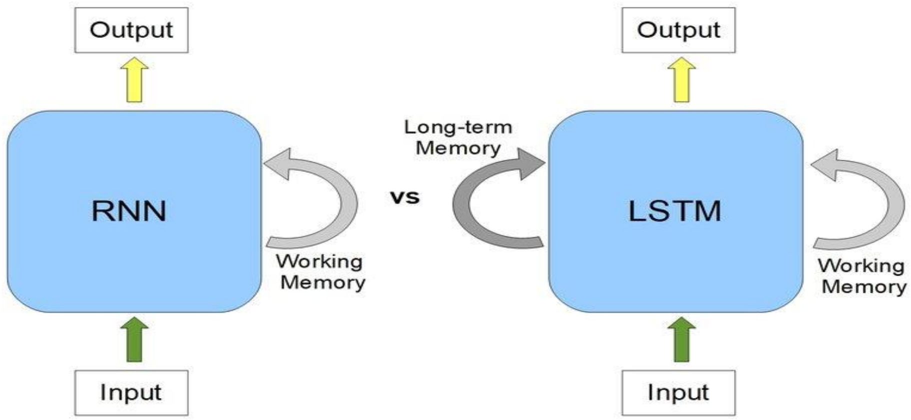

Figure 4.5: Difference between working of RNN and LSTM model

During the training phase, the LSTM classifier undergoes a learning process where it seeks to capture intricate temporal dependencies present in the input sequences. Leveraging the architecture's memory cell and gating mechanisms, the LSTM comprehensively processes the sequential data, retaining pertinent context from preceding time steps. This enables the model to make informed predictions grounded in the holistic understanding of the input text. Upon completion of training, the LSTM classifier is equipped to predict the sentiment label for new, unseen text data. By traversing through the learned LSTM architecture, the model processes the input sequences and generates a probability distribution across sentiment classes. The sentiment label with the highest probability is then assigned as the predicted sentiment label for the input text sample.

The utilization of LSTM networks for sentiment analysis within e-commerce datasets represents a sophisticated and technically sound approach. By harnessing the inherent capabilities of LSTMs to capture long-term dependencies and contextual nuances in text data, businesses can glean actionable insights from customer sentiments expressed in product reviews, thereby facilitating informed decision-making and product/service enhancements.

4.10 Classification using Bi-LSTM (Bidirectional Long Short-Term Memory)

Bidirectional Long Short-Term Memory (Bi-LSTM) networks stand as a potent deep learning technique for predicting sentiment labels based on textual data. Bi-LSTMs, an extension of traditional LSTMs, augment the model's capacity by processing input sequences in both forward and backward directions, effectively capturing contextual information from both past and future states. Initially, the preprocessed text data serves as the input sequences for the Bi-LSTM classifier. These sequences are typically encoded into numerical vectors using word embeddings, where each word or token is represented by a high-dimensional vector. This transformation enables the model to comprehend the semantic intricacies inherent in the textual data.

As with conventional classification paradigms, the sentiment labels assigned to each data instance (e.g., product review) act as the target variable for training the Bi-LSTM classifier. In binary sentiment classification scenarios, sentiments are encoded as binary values, with one class denoting positive sentiment and the other negative sentiment. During the training phase, the Bi-LSTM classifier learns to capture nuanced contextual dependencies present in the input sequences by processing them in both forward and backward directions. This bidirectional processing allows the model to leverage information from preceding and subsequent words, facilitating a more comprehensive understanding of the input text.

Upon completion of training, the Bi-LSTM classifier is adept at predicting sentiment labels for new, unseen text data. By processing the input sequences bidirectionally through the learned Bi-LSTM architecture, the model generates a probability distribution across sentiment classes. The sentiment label with the highest probability is then assigned as the predicted sentiment label for the input text sample. Leveraging Bi-LSTM networks for sentiment analysis within e-commerce datasets represents a sophisticated and technically robust approach. By harnessing the bidirectional processing capabilities of Bi-LSTMs to capture contextual nuances in text data, businesses can extract actionable insights from customer sentiments expressed in product reviews, facilitating informed decision-making and product/service enhancements.

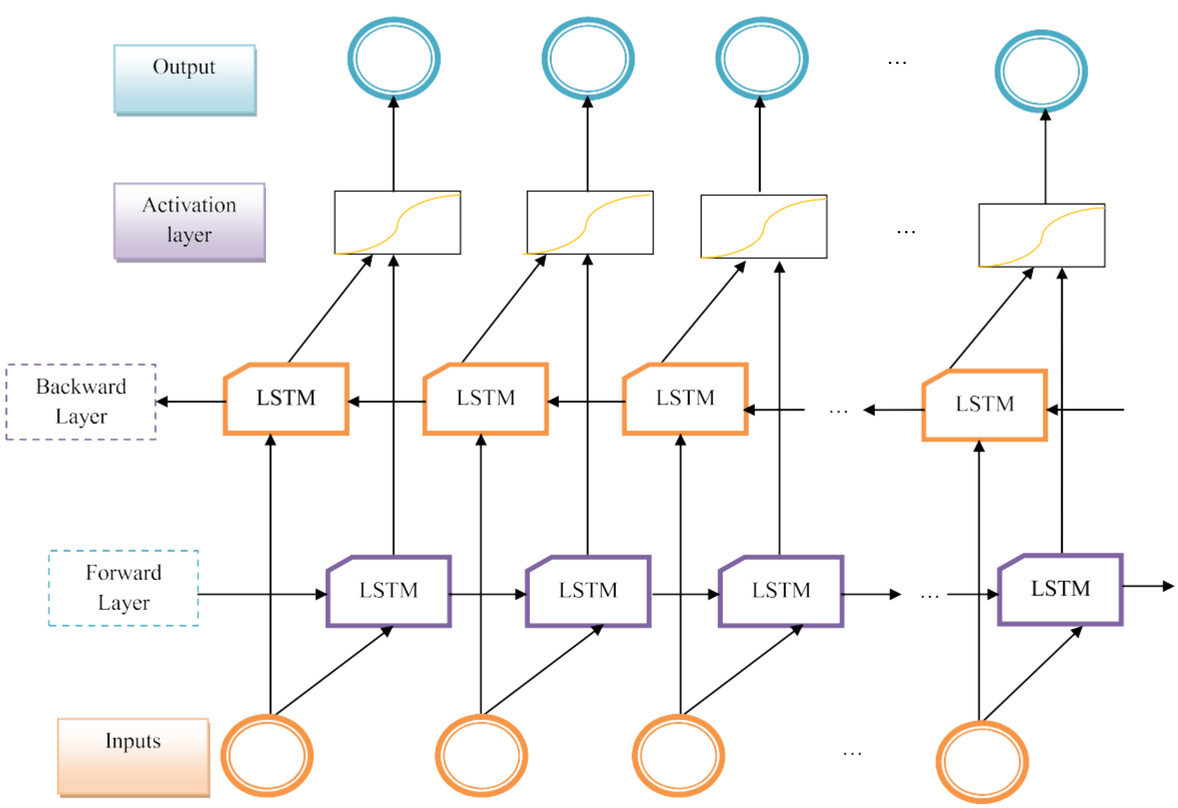

Figure 4.6: Architectural diagram of Bi-LSTM

The Bi-LSTM can encode back-to-front problems. Moreover, the Bi-LSTM model helps to predict the past and future input for effective classified outcomes.

The Bi-LSTM ensures the information data is in both directions. The Bi-LSTM is utilized for constructing the two layers of the LSTM network. Here, each LSTM layer performs the forward and backward calculations inside the neural networks. Moreover, the pre-processed data are inputted into the forward and backward LSTM. Hence, the output layer collides with the representation of the vector length.

4.11 Classification using CNN (Convolutional Neural Network)

Using a Convolutional Neural Network (CNN) for sentiment analysis on an Amazon dataset involves several steps. Firstly, you'd preprocess the text data by tokenizing it into individual words or subwords, removing stop words, and converting the text into a numerical format that the CNN can understand. Next, you'd initialize the CNN architecture, typically consisting of convolutional layers followed by pooling layers to extract features from the text data. These layers help the model learn hierarchical representations of the text at different levels of abstraction. Then, you'd add fully connected layers to the network to perform classification based on the learned features. During training, you'd use a labeled dataset to optimize the network's parameters using techniques like backpropagation and gradient descent. Finally, you'd evaluate the performance of the CNN on a separate test dataset to assess its accuracy in sentiment classification. By iteratively adjusting the model architecture and training parameters, you can fine-tune the CNN to achieve better performance on sentiment analysis tasks for Amazon reviews or similar datasets.

4.12 Classification using Hybrid CNN-Bi-LSTM (Hybrid Convolutional Neural Network Bidirectional Long Short-Term Memory)

The developed sentiment analysis model [35,36] integrates the hybrid classifier using CNN and Bi-LSTM for enhancing performance in terms of accuracy and sensitivity. Here, CNN is used to identify the essential features independently without any human support, and it contains a convolutional layer that provides many benefits to the network. However, it suffers from a certain lacking spatially invariant input feature. Bi-LSTM is also involved in the proposed model as it ensures a huge range of parameters such as ‘‘learning rates, input and output biases’, and therefore, it does not require any fine adjustments. But it takes a long time for training, and dropout is difficult to implement the Bi-LSTM also, it is highly sensitive to various random weight initializations. Hence, a new classifier is developed named OH-CNN-Bi-LSTM by combining the CNN and Bi-LSTM using suggested IGSO, which is designed by optimizing the hidden neurons count, number of epochs, learning rate and snapshot step of CNN and hidden neurons count and also the epochs of Bi-LSTM. It is the novelty of the designed framework. The tuning of parameters increases the accuracy of the system and reduces the overfitting issues. It helps to remove the local optima issues, and it can increase the search capability.

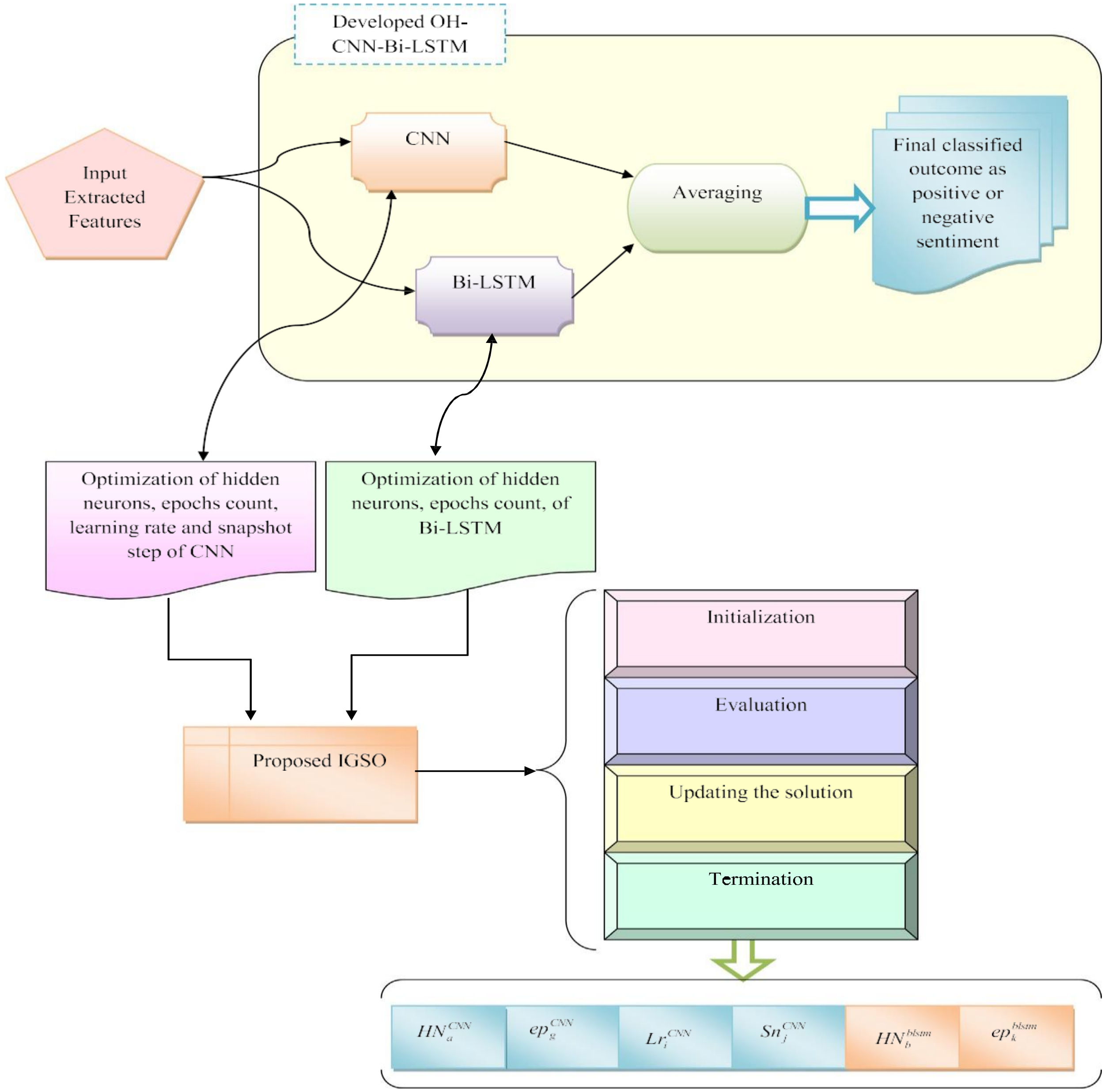

Figure 4.7: Hybrid CNN-Bi-LSTM

At the outset, the preprocessed text data serves as the input sequences for the Hybrid CNN-Bi-LSTM classifier. These sequences are typically transformed into numerical representations using word embeddings, capturing the semantic information embedded within the text data. Word embeddings provide a dense representation of words that preserves semantic similarities, enabling the model to comprehend the nuanced relationships between words in the text. Similar to other classification paradigms, sentiment labels assigned to each data instance (e.g., product review) act as the target variable for training the Hybrid CNN-Bi-LSTM classifier. In binary sentiment classification tasks, sentiments are typically encoded as binary values, with one class denoting positive sentiment and the other negative sentiment. During the training phase, the Hybrid CNN-Bi-LSTM classifier learns to extract local features from the input sequences using convolutional layers, followed by capturing contextual dependencies through bidirectional processing using Bi-LSTM layers. The convolutional layers of the CNN extract hierarchical features from the input text, while the Bi-LSTM layers model the temporal dependencies, enabling the model to capture both local and global information.

Upon completion of training, the Hybrid CNN-Bi-LSTM classifier is adept at predicting sentiment labels for new, unseen text data. By processing the input sequences through the learned hybrid architecture, the model generates a probability distribution across sentiment classes. The sentiment label with the highest probability is then assigned as the predicted sentiment label for the input text sample. leveraging a Hybrid CNN-Bi-LSTM architecture for sentiment analysis within e-commerce datasets represents a sophisticated and powerful approach. By combining the capabilities of CNNs in feature extraction and Bi-LSTMs in capturing temporal dependencies, businesses can effectively extract actionable insights from customer sentiments expressed in product reviews, facilitating informed decision-making and product/service enhancements.

5 Limitations

In contemporary machine learning paradigms, the efficacy of existing systems is frequently circumscribed by their capacity to grapple with the intricacies of data complexity. These systems, often operating within confined computational frameworks, are predisposed to succumb to overfitting—a phenomenon where models learn to perform exceptionally well on training data but falter when confronted with unseen instances due to an exaggerated adherence to idiosyncratic patterns within the training set.

Moreover, these systems rely heavily on assumptions about the underlying distribution of data. These assumptions serve as the foundational pillars upon which algorithms are built, guiding their decision-making processes. However, in the unpredictable terrain of real-world data, these assumptions may be fallacious, leading to a misalignment between the model's conceptual framework and the true dynamics of the data. Consequently, the performance of such models may degrade significantly when deployed in practical scenarios where the data deviates from the anticipated norms.

Furthermore, while basic algorithms—often characterized by their simplicity and interpretability—may suffice as initial benchmarks, they inevitably encounter a threshold beyond which further enhancements become arduous. This performance ceiling becomes particularly conspicuous when confronted with challenging datasets characterized by intricate patterns, high dimensionality, or noisy inputs. Attempts to push beyond this threshold often necessitate the adoption of more sophisticated methodologies capable of discerning nuanced relationships within the data and adapting dynamically to evolving contexts.

Therefore, the imperative for the development and utilization of advanced machine learning techniques capable of transcending the constraints of conventional approaches becomes increasingly apparent. These methodologies must possess the agility to navigate the multifaceted landscapes of real-world data, eschewing the rigidity of simplistic assumptions in favor of adaptive strategies that can accommodate the intricacies and uncertainties inherent in complex datasets.

6 Experiments and Results

The Logistic regression, Naïve Bayes classifier (Multinomial, Bernoulli, Gaussian), Decision Trees, LSTM, Bi-LSTM, and Hybrid CNN-Bi-LSTM are used for the sentimental analysis on Amazon dataset. The project involved using the mentioned algorithms to classify the data into positive and negative reviews.

It is observed that the ML classification models had lower execution time[Table1] as compared to Neural Network based algorithms[Table2] but Neural Network Based algorithms surpassed the ML based algorithm based on all other performance constraints as observed in[Table1, Table2].

The difference in observed and implemented evaluation metrics are also recorded displaying significant improvement in the results [Table3] Through this project work the results were found out to be as following:

Table1: Evaluation Metrics for Machine Learning classification models

Evaluation Metrics	Logistic Regression	M-NB	B-NB	G-NB	Decision Tree
Accuracy	0.87	0.84	0.80	0.80	0.70
Precision	0.87	0.85	0.79	0.79	0.67
Recall	0.88	0.83	0.82	0.82	0.81
F1-Score	0.87	0.84	0.81	0.81	0.73
Training Time(seconds)	18.874	0.341	0.77	0.65	17.573

Table2: Evaluation Metrics for ANN, RNN, CNN based models

Evaluation Metrics	LSTM	Bi-LSTM	CNN	Hybrid CNN-Bi-LSTM
Accuracy	0.90	0.95	0.93	0.94
Precision	0.91	0.94	0.91	0.94
Recall	0.89	0.96	0.96	0.94
F1-Score	090	0.95	0.93	0.94
Training Time (seconds)	50064	41078	2381	6933

Table3: Comparison between metrics of NN based algorithms

	Observed	Implemented	Observed	Implemented	Observed	Implemented	Observed	Implemented
Evaluation Metrics	**LSTM**		**Bi-LSTM**		**CNN**		**Hybrid CNN-Bi-LSTM**	
Accuracy	0.791	**0.90**	0.901	**0.95**	0.856	**0.93**	0.918	**0.94**
Precision	0.708	**0.91**	0.852	**0.94**	0.785	**0.91**	0.853	**0.94**
F1-Score	0.798	**090**	0.910	**0.95**	0.878	**0.93**	0.917	**0.94**

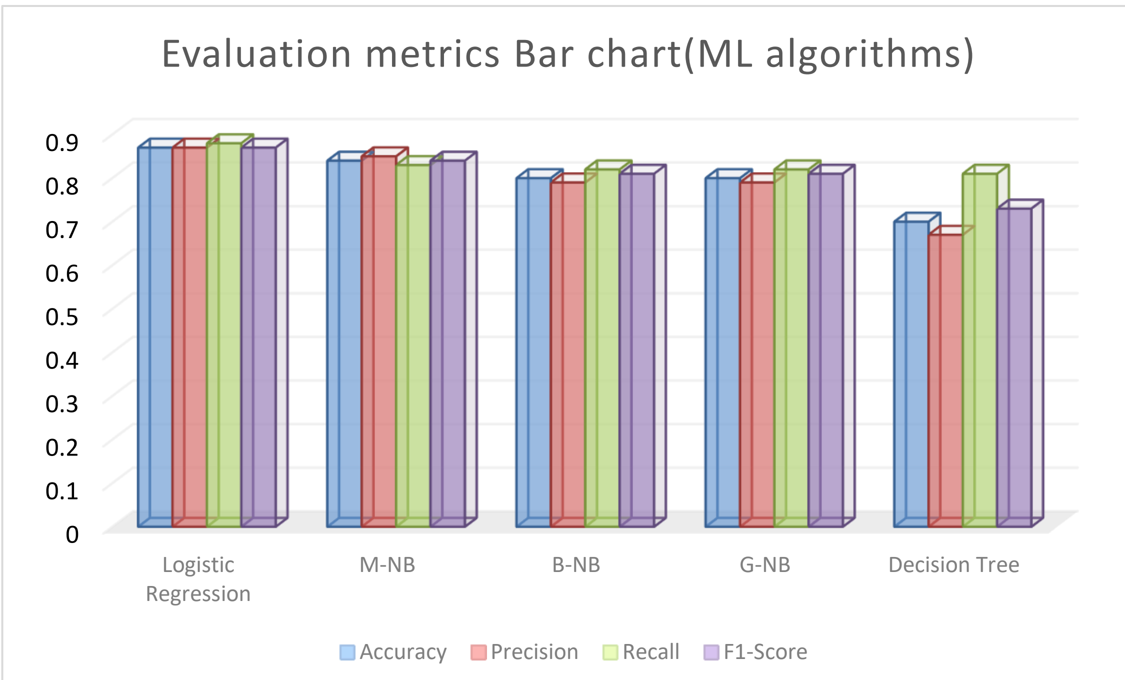

Figure 6.1: Evaluation metrics clustered bar graph (ML algorithms)

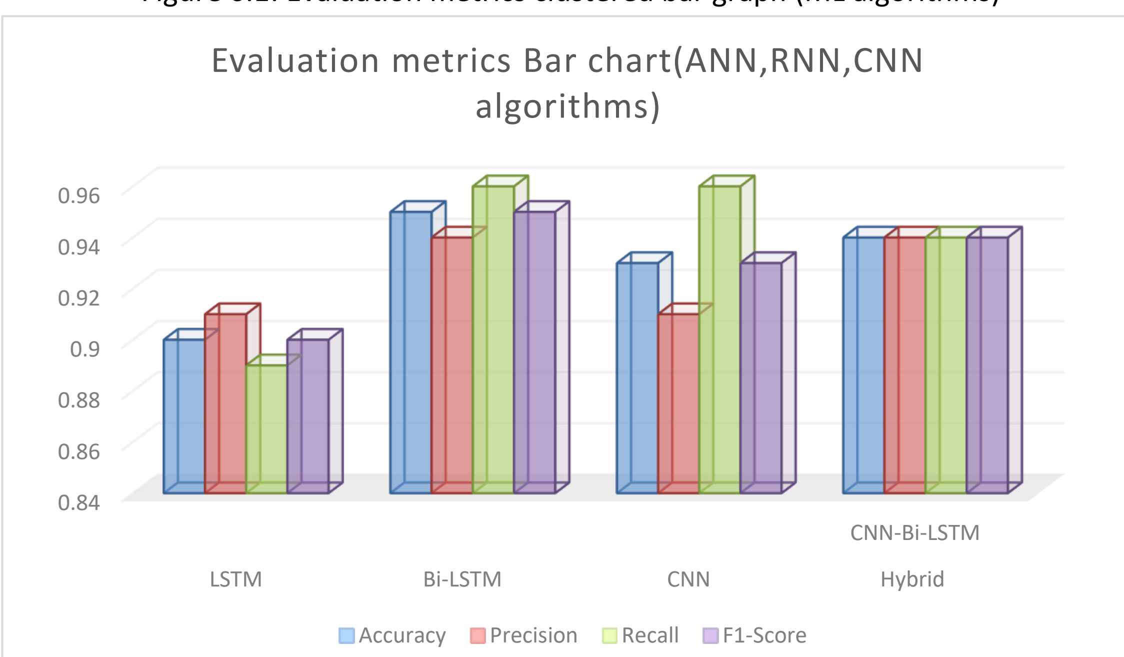

Figure 6.2: Evaluation metrics Bar graph (ANN, RNN, CNN algorithms)

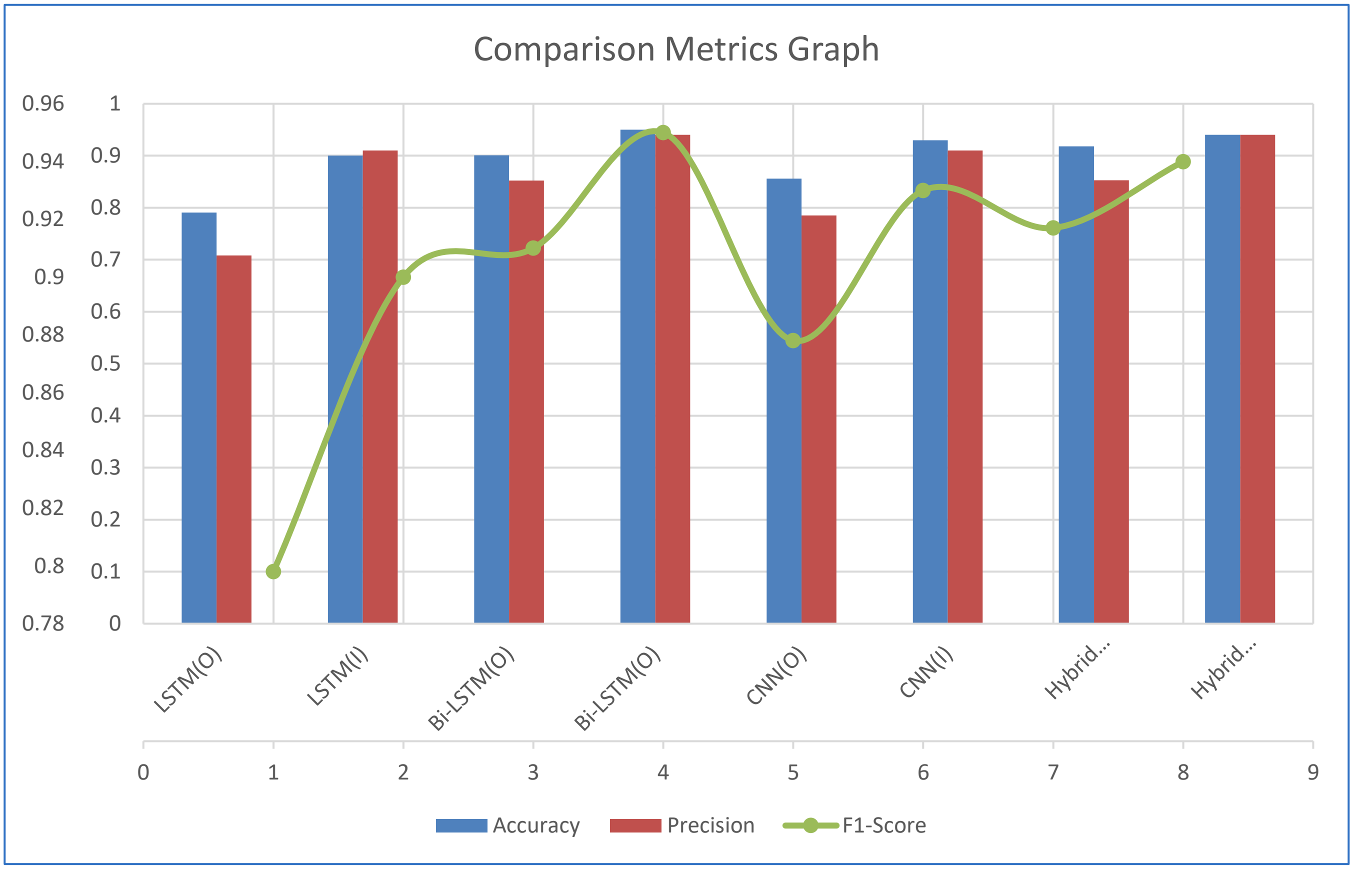

Figure 6.3: Comparison Metrics Graph

7 Dataset

- The primary dataset taken is Amazon Reviews for Sentiment Analysis from Kaggle Access date: 20-11-2021.’’ https://www.kaggle.com/datasets/bittlingmayer/amazonreviews
- The available dataset is in .bz2 file i.e. big zip extension, hence we need to either convert to csv or decode the file using “UTF-8” and assign the data into sequence with labeled columns.
- The labels are encoded in format,” __label__1, __label__2”, where “__label__1” is for negative comments and “__label__2” is for positive comments.
- As a part of conversion from “.bz2” file to “.csv” file the label is also encoded to “0: positive” and ”1: negative”.

```
Train Length  2880000
Train Label Length  2880000
Test Length  400000
Test Labels Length  400000
```

Figure 6.4: Training and Test Dataset distribution

8 Summary

A study on sentiment analysis in the context of e-commerce utilized a Hybrid CNN-Bi-LSTM model and compared its performance with several other classifiers. The evaluation metrics measured include accuracy, precision, recall, F1-score, and training time. The results showcase the effectiveness of the Hybrid CNN-Bi-LSTM model in sentiment analysis tasks.

The logistic regression model achieved an accuracy of 0.87, with similar precision and F1-score values. However, it lagged behind in terms of recall compared to other models, indicating potential issues in identifying certain sentiments accurately. The Multinomial Naive Bayes (M-NB) and Bernoulli Naive Bayes (B-NB) models scored lower in accuracy, precision, recall, and F1-score, suggesting limited performance in capturing the complexity of sentiment in e-commerce data.

The Gaussian Naive Bayes (G-NB) model showed slightly better performance than M-NB and B-NB but still fell short compared to more advanced models. The decision tree model, while achieving reasonable accuracy, suffered from lower precision, recall, and F1-score, indicating potential overfitting or difficulty in capturing nuanced sentiment patterns.

In contrast, the LSTM model demonstrated competitive performance across all metrics, with high accuracy, precision, recall, and F1-score. However, it significantly lagged in training time, taking substantially longer compared to other models, which could be a critical factor in real-world deployment scenarios.

The Bi-LSTM model further improved upon the LSTM by achieving higher accuracy, precision, recall, and F1-score while maintaining a similar training time. This highlights the efficacy of bidirectional recurrent architectures in capturing contextual information for sentiment analysis tasks.

The Bi-LSTM model emerged as a top performer, due to fitting of dataset to the model achieving a high accuracy of 0.95 along with excellent precision, recall, and F1-score values. Moreover, it significantly reduced the training time compared to the LSTM model. The Hybrid CNN-Bi-LSTM model had the best stability of all giving accuracy of 0.94 and very low time compared to LSTM and Bi-LSTM model indicating a balance between performance and efficiency. This underscores the effectiveness of integrating convolutional and recurrent neural network architectures for sentiment analysis tasks, particularly in e-commerce datasets, where capturing subtle sentiment nuances is crucial for understanding customer feedback and making informed business decisions.

9 Conclusion

In this research work, a novel sentiment analysis model using the developed Hybrid-CNN-Bi-LSTM along with the help of suggested CNN to classify the sentimental tweets. Initially, the data was collected from various standard datasets. Then, the input data was further fed to the pre-processing steps. Then, the pre-processed data was given to the feature extraction phase. Moreover, the positive and negative. Also, the features were extracted using the word-to-vector formation technique. Then, all the features were concatenated and considered for the classification phase. The developed Hybrid-CNN-Bi-LSTM method was used for performing efficient classification. Furthermore, the classification performance has been enhanced using the tuning of certain parameters of LSTM and Bi-LSTM. Through the analysis of the proposed model, the accuracy and specificity of the developed LSTM and Bi-LSTM method have attained 90% and 95%, respectively. Hence, it was revealed that the implemented sentiment analysis framework had ensured elevated efficiency over the existing algorithms and diverse state-of-the- methods. Further, this model can be extended by including various data from different varieties of sources. The practical implications of the designed model are depicted below. The practical implications of sentiment analysis are utilized to detect the negative and positive sentiments in the text data, and it is utilized for businesses for predicting the sentiment over the understand customers, social data, and gauge brand reputation. Moreover, the suggested method can be useful for real-time applications like healthcare, finance, media, consumer markets, and government-based applications. Especially, it is widely used for customers to detect emotions regarding brands, products, or services. Using large and multi-application-oriented datasets is a complicated issue for the designed sentimental analysis for the E-commerce model. It does not involve sentiment analysis with sarcasm tweets. The standard datasets with intelligent approaches, like more advanced deep learning models, will be added to the given designed method to get an accurate result in sentiment classification.

10 References

[1] Bing Liu. "Sentiment Analysis and Opinion Mining" (2012).

[2] Minqing Hu and Bing Liu. "Mining and Summarizing Customer Reviews" (2004).

[3] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. "Thumbs up? Sentiment Classification using Machine Learning Techniques" (2002).

[4] Bing Liu. "Sentiment Analysis: Mining Opinions, Sentiments, and Emotions" (2015). Available: Google Scholar Profile

[5] Yoon Kim. "Convolutional Neural Networks for Sentence Classification" (2014).

[6] Ashish Vaswani et al. "Attention Is All You Need" (2017).

[7] Paulo Cavalin Gonçalves and Matheus Araújo. "Sentiment Analysis on Social Media Texts: A Survey" (2018).

[8] Reference not explicitly provided.

[9] Sepp Hochreiter and Jürgen Schmidhuber. "Long Short-Term Memory" (1997).

[10] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. "Neural Machine Translation by Jointly Learning to Align and Translate" (2014).

[11] Alex Graves et al. "Hybrid computing using a neural network with dynamic external memory" (2016).

[12] Mike Schuster and Kuldip K. Paliwal. "Bidirectional recurrent neural networks" (1997).

[13] Barbara Plank, Anders Søgaard, and Yoav Goldberg. "Multilingual Part-of-Speech Tagging with Bidirectional Long Short-Term Memory Models" (2016).

[14] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. "Neural Architectures for Named Entity Recognition" (2016).

[15] Reference not explicitly provided.

[16] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. "Hierarchical Attention Networks for Document Classification" (2016).

[17] Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. "Attention-Based Bidirectional Long Short-Term Memory Networks for Relation Classification" (2016).