Opinion Mining of Reviews in E-Commerce with Convolutional Neural Network

***Ajay Butola**
****Prof. M.M.S Rauthan**

ABSTRACT

The Fourth Industrial Revolution (4.0) refers to the integration of emerging technologies such as Artificial Intelligence (AI), Robotics, the Internet of Things (IoT), Genetic Engineering, Quantum Computing, and several other digital innovations. With improvements in internet infrastructure and the reduction of data costs, a vast number of people now rely heavily on online platforms and digital services. This rapid digital adoption has reshaped how companies engage with online markets and has created new challenges as well as opportunities for businesses seeking to strengthen their presence in the digital space. E-commerce platforms such as Amazon provide both customer-to-customer (C2C) and business-to-business (B2B) services in the apparel industry. To increase profitability, companies need to closely understand customer requirements and preferences. For this reason, sentiment analysis of consumer reviews is conducted. Since manually analyzing large volumes of reviews is slow and inefficient, artificial intelligence techniques are employed to automate the process. A study conducted on a women's apparel e-commerce store revealed that applying Convolutional Neural Networks (CNN) along with word vector generation and TF-IDF achieved nearly 94% accuracy, demonstrating the effectiveness of this AI-based approach in handling sentiment classification tasks.

Keywords: Deep Learning, Convolutional Neural Networks (CNN), Opinion Mining, Long Short-Term Memory Networks (LSTM), Word Embeddings (Word2Vec), Term Frequency–Inverse Document Frequency (TF-IDF)

I. INTRODUCTION

According to a 1212 report, India had around 613 million active internet users. Compared to the previous year, this reflected an increase of nearly 47 million users, marking a growth rate of 8.2%. Considering India's total population of approximately 1.39 billion at that time, it can be inferred that nearly 64% of the country's people had access to the internet [1]. A major factor behind the rapid expansion of internet access in India is the improvement of digital infrastructure, which has extended even to rural and remote regions. Alongside this, the advancement of information technology—particularly online platforms—has significantly transformed the way people interact and communicate with one another [1]. In today's digital era, people actively use online channels to voice their thoughts, share personal experiences, and discuss matters of interest. The continuous advancement of technology has accelerated the flow of information, making it easier to distribute content instantly through platforms like Facebook, Twitter, personal blogs, and official institutional websites [2]. The rapid accessibility and speed of digital platforms have transformed how individuals consume information. Traditional sources such as newspapers and printed magazines are increasingly being replaced by online alternatives like news portals and weblogs. In the present digital age, people prefer to access news and updates from diverse sources that match their personal interests or are considered significant [3]. To capture audience attention, online platforms employ innovative strategies and continuously refine their engagement methods. These platforms also serve as spaces for expressing opinions about communities, locations, or even specific objects while reporting ongoing events [4]. Consequently, many e-commerce platforms and shopping websites now provide interactive sentiment analysis tools that categorize user feedback into neutral, negative, or positive classes [5,6]. Digital platforms generate a wide variety of discussions and viewpoints, covering areas such as economics, politics, lifestyle, health, entertainment, and knowledge sharing. Among these, materialism and financial matters are particularly significant, as they directly influence individuals, businesses, and even traditional markets, depending on the country's economic conditions. Prior studies [7] have highlighted that stock market behaviour can be predicted through articles related to stock prices, where the sentiment expressed in such content may either positively or negatively impact market fluctuations. Similarly, the tone of financial news often shapes public perception and can also guide policy directions set by local authorities. Given the strong impact of economic themes on society, this study focuses on analysing customer sentiment expressed in e-commerce product reviews. The dataset for this analysis has been obtained from Kaggle (www.kaggle.com), which provides a reliable collection of user-generated content for research purpose. Sentiment Analysis (SA), commonly referred to as opinion mining, is the process of identifying, interpreting, and classifying user opinions expressed in words, sentences, or complete documents [6]. Previous research has explored SA using a variety of models, including Deep Learning (DL)-based Emotion Recognition, Multi-Layer Perceptron (MLP), and Long Short-Term Memory (LSTM) networks. Comparative studies have shown that while the accuracy achieved by both models is relatively close—around 71.74%—the MLP model demonstrates faster processing efficiency [7]. The advancements in Deep Learning have significantly transformed multiple domains such as business, healthcare, product manufacturing, and entertainment, making it one of the most impactful technologies of recent years.Studies indicate that the application of Long Short-Term memory (LSTM) and its variant, Branch LSTM, for sentiment evaluation on Twitter achieved an accuracy of nearly 78% [8]. In another work [9], researchers utilized LSTM with an encoding mechanism to analyse user opinions expressed on social media platforms. Specifically, LSTM-based methods were employed to study public sentiment regarding COVID-11, where the model reached an accuracy of

81.15% [10]. Similarly, further investigations using LSTM have reported improved performance, with accuracy levels reaching up to 85% [11]

II. LITERATURE REVIEW

Sara Ashour Aljuhani and Norah Saleh Alghamdi, in their research work [1], evaluated the effectiveness of several machine learning algorithms, including logistic regression, convolutional neural networks (CNN), naïve Bayes, and stochastic gradient descent. Their experimental results revealed that convolutional neural networks, when combined with Word2Vec for feature extraction, achieved the highest accuracy for both balanced and imbalanced datasets. Similarly, in another study titled *"Sentiment Classification of Online Consumer Reviews Using Word Vector Representations"* by Bansal, Barkha, and Sangeet Srivastava [2], the authors employed the Word2Vec model to transform textual reviews into vector representations for sentiment classification. The dataset used contained more than 400,000 Amazon customer reviews from the mobile phone category. Using 10-fold cross-validation, along with the CBOW (Continuous Bag of Words) and Skip-gram models, they tested multiple machine learning algorithms such as SVM, Naïve Bayes, Logistic Regression, and Random Forest. Their findings indicated that the Random Forest algorithm using the CBOW model delivered the highest accuracy among all tested methods.

Another study by Lee et al. [3] proposed a model for analyzing user sentiment using a word embedding space generated from Amazon fashion product review data. In their experiments, three SVM classifiers were trained based on the proportion of positive and negative reviews. The word embedding space was created by learning from 5.7 million Amazon reviews. The study reported a maximum accuracy of 88.0% when an SVM classifier was trained using a dataset of 100,000 reviews—comprising 50,000 positive and 50,000 negative samples.

In another research work titled *"Sentiment Analysis of Product Reviews Using Supervised Learning"* by Shah and Arkesha [4], the authors utilized various machine learning approaches, including Naïve Bayes, Support Vector Machine (SVM), and a CNN deep learning model integrated with FastText word embeddings, to analyze mobile phone reviews. Among these techniques, the CNN model combined with FastText embeddings demonstrated superior performance compared to traditional machine learning methods.

Salmony et al.'s research paper from [5] used the Amazon product review dataset in conjunction with conventional machine learning techniques like NB, SVM, and RF. Ultimately, the researchers evaluate and compare the most effective approach. Of these techniques, Random Forest yields the most accurate results. In the study "Amazon Product Reviews: Sentiment Analysis Using Supervised Learning Algorithms", conducted by Hawlader et al. in [6], the classification algorithms Naïve Bayes, Support Vector Machine, Random Forest, Decision Tree, Logistic Regression, and Multi-Layer Perceptron classifier were used to analyze the Amazon electronics product reviews. MLP produces the highest yield, 92%.

In this study, Table 1 presents a comparative analysis of various research papers. The literature review enables researchers to gain a clear understanding of the core concepts, evaluate different methodologies, and identify existing limitations and key findings. Such an assessment provides valuable insights that contribute to refining the current research by integrating both the proposed approach and selected existing methods effectively.

Reference No.	Title of the Paper	Author Name	Methods Used	Best Method
[1]	A comparison of sentiment analysis methods on Amazon reviews of Mobile Phones	Aljuhani, Sara Ashour, and Norah Saleh Alghamdi	convolutional neural networks, stochastic gradient descent, naive bayes, and logistic regression	convolutional neural networks provides the best result
[2]	Sentiment classification of online consumer reviews using word vector representations	Bansal, Barkha, and Sangeet Srivastava	SVM, Nave Bayes, Logistic Regression, and Random Forest	Random Forest scores highest F1 Score
[3]	User sentiment analysis on Amazon fashion product review using word embedding	Lee, Dong-yub, Jae-Choon Jo, and Heui-Seok Lim	Support Vector Machine	Support Vector Machine provides 88.0% of accuracy
[4]	Sentiment analysis of product reviews using supervised learning	Shah, Arkesha	Naïve Bayes, Support Vector Machine and also FastText word embedded with CNN deep learning model	FastText word embedded with CNN deep learning method achieves the best result
[5]	Supervised Sentiment Analysis on Amazon Product Reviews: A survey	Salmony, Monir Yahya Ali, and Arman Rasool Faridi	NB, SVM and RF	RF gives the best performance
[6]	Amazon product reviews: Sentiment analysis using supervised learning algorithms	Hawlader, Mohibullah, Arjan Ghosh, Zaoyad Khan Raad, Wali Ahad Chowdhury, Md Sazzad Hossain Shehan, and Faisal	Naïve Bayes, Support Vector Machine, Random Forest, Decision Tree, Logistic Regression and Multi-Layer perceptron classifier	MLP yield the best result of 92%.

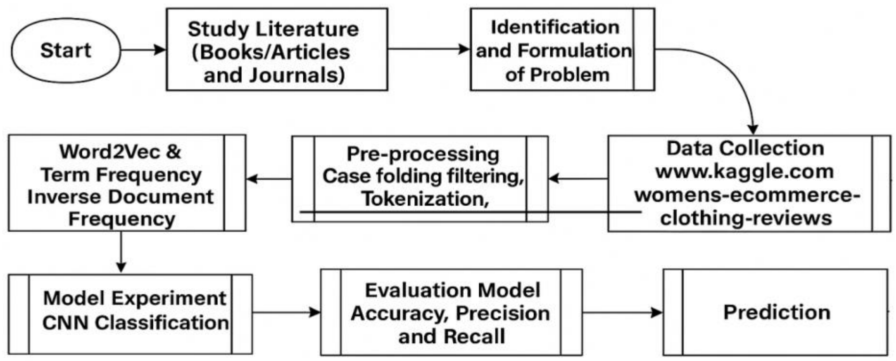

Figure 1 Research Flow

III. RESEARCH METHODOLOGY

- **Description of dataset**

The dataset used in this research is **Womens_Clothing_E-Commerce_Reviews**, which contains **13,486 instances** of customer reviews collected from an online retail platform. The total dataset is divided and categorized using **10-fold cross-validation** to evaluate detailed accuracy across different sentiment classes.

Each attribute in the dataset consists of multiple distinct values representing various aspects of customer feedback. The key attributes include **Clothing ID**, **Age**, **Title**, **Review Text**, **Rating**, **Recommended IND**, **Positive Feedback Count**, **Division Name**, **Department Name**, and **Class Name**.

Some concepts relevant to this work are outlined below:

- **TF-IDF**

Term Frequency–Inverse Document Frequency, widely known as TF-IDF, is a numerical method that determines the significance of a word in a document relative to an entire dataset. The *term frequency* part shows how often a word occurs in a single text, whereas the *inverse document frequency* decreases the weight of common words that appear repeatedly in many texts. When these two measures are combined, the method highlights terms that are frequent in one document but uncommon across the collection. This makes TF-IDF a powerful tool for identifying key terms and ranking documents effectively. The process of applying TF-IDF may vary slightly across different applications, yet the overall methodology remains quite similar. **Term Frequency (TF)** represents how many times a particular word appears within a given document. The value of TF is usually calculated using the following formula:

$TF(t) = \frac{ft,d}{\sum t,d'}$ (1)

Here, $ft_{,}d$ denotes the raw frequency of a term t within a document d, meaning the number of times that term appears in the given document. The term $\sum, d'$ represents the total count of all words in that document.

To compute the **Inverse Document Frequency (IDF)**, which measures how significant a word is across multiple documents, the following equation is applied:

$$IDF(t) = log(|D|/f_{t,d} \quad (2)$$

In this expression, |D| represents the total number of documents within the collection, while ftd indicates the number of documents in which the term t occurs (Salton & Buckley, 1118; Berger et al., 1200). For a document collection D and an individual document d ∈ D indicates the **TF-IDF** value of a term can be computed using the following formula:

$$TF - IDF(t) = TF(t) * IDF(t) \quad (3)$$

It is generally assumed that the total number of documents |D| and the document frequency of a term dft are close in scale. When the condition $\boldsymbol{1 < log(\frac{|D|}{ftD}) < c}$, where c is a small constant, is satisfied, the weight www (word embedding) becomes smaller than the raw frequency ft, d but still remains positive. This indicates that the word embedding maintains a general relationship with most documents while still preserving valuable information. For example, the TF-IDF approach may assign significance to the term *android* when it appears in a document that specifically contains articles discussing Android. This concept also applies to certain words such as conjunctions, which do not carry significant meaning in the context of a document. Such words are assigned a very low weight when using TF-IDF. On the other hand, if the term frequency ft, d in a particular document is high while the document frequency dft across the collection is low, then the value of $\boldsymbol{log(|D| / dft)}$ will also be high. As a result, the $\boldsymbol{TF - IDF(t)}$ score for that word will be large. A high TF-IDF value indicates that the word is highly relevant and important for the specific document d, but not commonly significant across the entire collection D.

- **Word2Vec**

Word2Vec is an effective word embedding technique that transforms words into numerical vectors [11,11]. For instance, the word *"India"* may be represented as a five-dimensional vector such as **[-0.3, 0.6, 0.8, 0.9, -0.2]**. These vectors capture not only the syntactic structure of words but also their semantic meaning, enabling a better understanding of relationships between words in a dataset.

- **Tensor Flow**

TensorFlow, released by Google in November 1215, has become a core component in several of its products such as Google Search, spam filtering, speech accent recognition, Google Assistant, and Google Photos. The framework allows partial sub-graph computations, which makes it suitable for distributed training and partitioned neural networks. This capability enables both parallel processing and data parallelism. TensorFlow also offers multiple APIs; at its lowest level, **TensorFlow Core** gives developers complete flexibility and control over programming[11–13].

- **Convolutional Neural Network**

The Convolutional Neural Network (CNN) is a type of Artificial Neural Network (ANN) that gives feedback while maintaining a hierarchical structure. In drawing tasks such as object detection and computer vision, CNN investigates how intrinsic features are represented and generalized. Users are not limited to images; may also be used to achieve results in Natural Language Processing (NLP) problems and speech recognition [13-14]. The architecture of the CNN is depicted in Fig. 3. The **Max Pooling layer** extracts the maximum value from each region (or patch) of the feature map, helping to reduce dimensionality while preserving important features. On the other hand, the **Dense layer** adjusts the dimensionality of vectors by connecting every neuron to the outputs of the previous layer, ensuring that the results from the preceding layer are passed to each neuron in the dense layer.

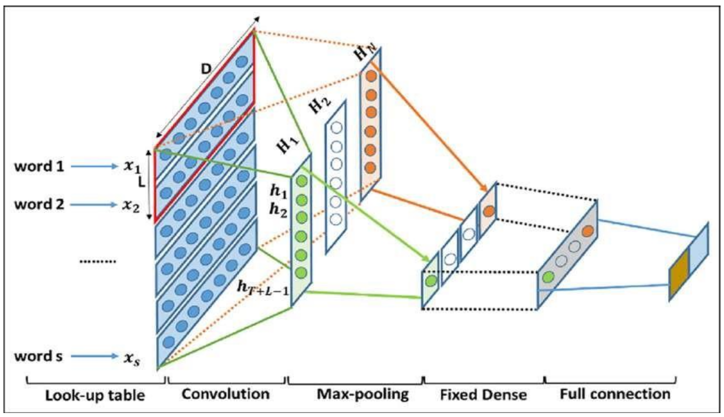

Figure 2. CNN Architecture [15]

- **Data Collection**

The dataset employed in this research consists of 13,000 customer reviews and ratings obtained from the article “Women’s E-Commerce Clothing Reviews” [14]. A probability sampling technique was adopted, specifically simple random sampling. The number of records selected for the training phase was determined using Slovin's formula [14,14], while the remaining data was utilized for testing purposes.

The sample size was calculated using Slovin's formula, which is expressed as:

$$n = \frac{N}{1+Ne^2} \quad (4)$$

The total population size is represented by N, the sample size by n, and the fault tolerance limit by e. For effective training of machine learning models, it is necessary to prepare training data using appropriate algorithms. In this case, a total of 13,000 reviews or comments are available. To divide this dataset, Slovin's formula has been applied with a 5% level of significance for selecting the testing data.

$$n = \frac{13400}{1+13400\ 0.05^2} = 5750. \quad (5)$$

Out of the total 13,000 reviews, a subset of 5,750 review titles was selected for testing. Following this, the data underwent preprocessing to ensure its suitability for further analysis.

- **Data Labeling**

In this study, a total of 13,000 datasets were utilized, with 11,140 allocated for training and 5,750 reserved for testing. Since each document reflects either positive or negative opinions and sentiments, the data was labeled accordingly by assigning positive or negative tags [14]. The labeling task was carried out by three reviewers to ensure consistency and reliability. To achieve accurate labeling, the **Majority Voting method** was adopted [14]. This approach, one of the simplest and most widely applied techniques, is based on decision-making through votes, where the final label is determined by the option receiving the majority of votes.

- **Labeling**

Table 1 shows the foundation for labeling the number of label data:

Type of Data	Sentiment	Amount
Data Training	Positive	10158
	Negative	7092
Data Testing	Positive	5133
	Negative	1495

Table 1. Labeling Results

- **Pre-processing**

In this phase, the dataset saved in CSV format is subjected to preprocessing. The main objective of this step is to clean and organize the raw text data, ensuring it is ready for further analysis. The process begins with **case folding**.

-Case folding -where every character in the text is converted into lowercase form. This step helps maintain consistency and minimizes duplication caused by differences in letter casing.

-Filtering In this stage, special characters within the dataset are eliminated to refine the text. The removed characters include punctuation marks (such as periods, commas, question marks, and exclamation points), numeric values (0–9), and additional symbols (e.g., $, %, *). Furthermore, elements that do not contribute meaningfully to the analysis are also discarded. These include user mentions beginning with the "@" symbol, hashtags (#), URLs, emoticons, and other irrelevant tokens, as they provide little to no impact on the sentiment labeling process [15].

-Tokenization Tokenization refers to the process of breaking down text into smaller components, usually words. Typically, words within a sentence are separated by spaces, and this property is utilized to identify and segment them [15]. Once tokenized, the sentence is represented as an array or list, with each element holding an individual word from the original text.

Table 2. Sentence Conversion Results

Process	Results
Word Dictionary	['awesome', 'issue', 'boycott', 'stock', 'Starbucks', 'already', 'down']
Numeric Conversion	[1063, 491, 1146, 15, 797, 89, 58]
Padding Sequences	range([[1, 0, 0, ..., 111, 59, 56], [2, 0, 1, ...,11, 57, 493], [1, 0,1, ..., 394, 54, 39], ..., [1, 0, 1, ..., 4677, 13, 388], [1, 0, 1, ..., 15, 49, 135], [1, 0, 1, ..., 11, 111, 12]], data type= integer 15)

-Sentence Conversion After the tokenization process, each sentence is transformed into an array, where every element represents an individual word from the text. The results of this sentence-to-array conversion are presented in Table 2.

```
CountVectorizer(analyzer='word', binary=False, decode_error='strict',
                dtype=<class 'numpy.int64'>, encoding='utf-8', input='conte
                lowercase=True, max_df=1.0, max_features=None, min_df=1,
                ngram_range=(1, 1), preprocessor=None, stop_words=None,
                strip_accents=None, token_pattern='(?u)\\b\\w\\w+\\b',
                tokenizer=None, vocabulary=None)
  (0, 6439)     2
  (0, 11502)    2
  (0, 7111)     1
  (0, 2522)     1
  (0, 4338)     1
```

Figure 4. Word2vec Model

Word2Vec Word2Vec [15] is an embedding technique that transforms words into vector representations within a continuous vector space. In this work, the word dictionary was generated using a collection of textual articles, processed through the Natural Language Toolkit (NLTK) available in Python via the sklearn library. The structural design of the Word2Vec model is presented in Figure 4.

-TF-IDF- The **Term Frequency–Inverse Document Frequency (TF-IDF)** method evaluates the importance of a term t within a text by multiplying its Term Frequency (TF) with its Inverse Document Frequency (IDF). Several approaches exist for computing the TF value, one of which is based on the following formula, with the corresponding results presented below [14–11]:

$$tf = 0.5 + 0.5 * \left(\frac{tf}{\max tf}\right) \quad (6)$$

And Inverse Document Frequency can be found with the formula:

$$idf_j = log\left(\frac{D}{dfj}\right) \quad (7)$$

```
TfidfTransformer(norm='l2', smooth_idf=True, sublinear_tf=False, use_idf=True)
  (0, 11502)    0.6124657626824258
  (0, 11301)    0.14069882383511612
  (0, 8669)     0.16251953114144 95
  (0, 7111)     0.3630115572293248
  (0, 6439)     0.27975358499928116
  (0, 6321)     0.1867712798055541
  (0, 6051)     0.1866182751106572
  (0, 5937)     0.30485577730784624
  (0, 4338)     0.13334766166027093
```

Figure 5. TF-IDF Model

IV. RESULTS AND DISCUSSION

- **Model Testing**

At this stage, the developed models are trained and evaluated to determine their performance based on predefined parameters. The model with the most suitable parameter values is identified through this testing process. In the case of the CNN model, evaluation produces both accuracy and loss metrics for the training and validation datasets.

The training was conducted over 100 epochs with a batch size of 146. During this process, the reliability of the training data and the minimum loss value are assessed using the calculated parameters. The model also records the optimal epoch, corresponding to the lowest loss achieved throughout the training iterations. The outcomes of this testing are illustrated in Figure 6, which presents the architecture and performance of the CNN model.

- **Model Accuracy**

Figure 7 presents the accuracy results corresponding to the model loss shown in Figure 6. At the 10th epoch, the CNN model achieves an accuracy of 0.90, while by the 40th epoch, the accuracy increases to 0.94.

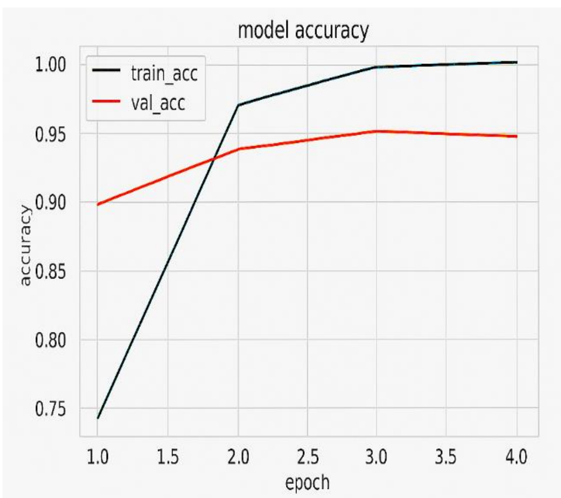

Figure 6. Model Loss Testing Using CNN

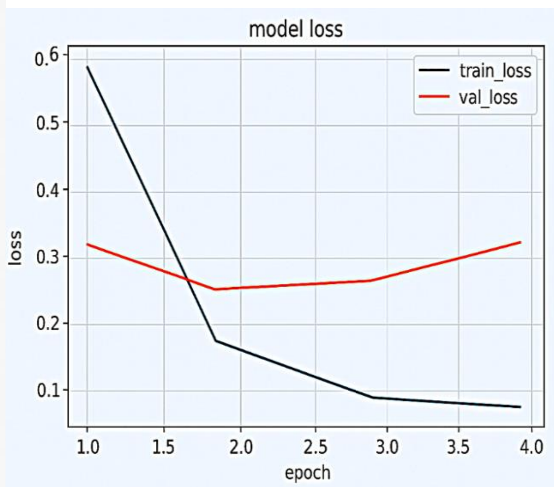

Figure 7. Model Accuracy Testing using CNN

The comparison indicates that the model performs effectively, as the accuracy values for both training and validation data remain relatively close. This consistency suggests that the model does not exhibit significant signs of overfitting. Therefore, the accuracy evaluation confirms that the proposed model is capable of analyzing sentiments in e-commerce product reviews and achieves an overall accuracy of 94%.

- **Data Testing**

At this stage, the trained model is evaluated using real-world data. A total of 5,750 reviews are considered, consisting of 1,495 reviews (15%) classified as negative and 5,133 reviews (85%) as positive. The model's performance in predicting each class is assessed by computing key evaluation metrics such as accuracy, precision, and recall, which also reflect the confidence level of the predictions. The effectiveness of the model in class classification is illustrated in Figure 8, presented through a confusion matrix.

	precision	recall	f1-score
0	0.96	0.96	0.96
1	0.91	0.94	0.93
2	0.94	0.91	0.92
accuracy			0.94
macro avg	0.94	0.94	0.94
weighted avg	0.94	0.94	0.94

Figure 8 Confusion Matrix

Furthermore, when the model makes predictions for each class, its confidence level is evaluated using accuracy, precision, and recall. The objective is to determine how reliable the accuracy percentage is for class prediction. The proposed scheme achieved strong results: for positive sentiments, it obtained a precision of **0.96**, recall of **0.96**, and F1-score of **0.96**. For negative sentiments, it reached a precision of **0.91**, recall of **0.94**, and F1-score of **0.93**. Likewise, for neutral sentiments, the model recorded a precision of **0.94**, recall of **0.91**, and F1-score of **0.92**. Overall, the scheme achieved an accuracy of **0.94**, demonstrating its effectiveness in sentiment classification.

V- CONCLUSION

The present work explored the application of a Convolutional Neural Network (CNN) for sentiment analysis in the context of women's apparel reviews within e-commerce platforms. The experimental evaluation, conducted through key performance indicators such as accuracy, precision, and recall, suggests that the designed framework provides dependable outcomes for large-scale text classification.

To begin with, the pre-processing pipeline ensured that raw customer reviews were systematically refined into structured inputs. This stage not only minimized redundant and noisy information but also enabled smoother transformation of text into features suitable for computational modelling.

Furthermore, the adoption of representation methods, including vector-based embeddings and term-weighting schemes, contributed to the effective normalization of data. These approaches allowed unstructured opinions to be converted into quantitative patterns, thereby enhancing the capacity of the CNN model to identify sentiment categories with greater consistency. Most importantly, the integrated framework that merged semantic embeddings with deep learning achieved highly promising results, reporting an overall accuracy of nearly **94%.** This outcome highlights the capability of the proposed approach to deliver reliable sentiment classification and demonstrates its potential value for e-commerce organizations seeking to understand consumer behaviour.

In summary, this study emphasizes that sentiment analysis, when supported by robust preprocessing, advanced representation techniques, and deep neural architectures, can substantially improve insights into customer preferences. While the findings validate the effectiveness of the proposed framework, future investigations may consider extending this work by employing hybrid architectures, incorporating larger multilingual datasets, and analyzing cross-domain product categories. Such directions would not only broaden the applicability of the model but also enhance its adaptability to the evolving landscape of digital commerce.

VI. FUTURE SCOPE

In future research, the scope of datasets can be expanded, and performance can be compared across larger and more varied collections. While this study focused on binary classification, future work may explore multi-class sentiment labeling to capture more detailed insights. Moreover, integrating additional algorithmic models or developing hybrid approaches could further improve accuracy and overall performance.

Hemvati Nandan Bahuguna
Garhwal University
Srinagar, Uttarakhand

References

[1] Aljuhani, Sara Ashour, and Norah Saleh Alghamdi, "A comparison of sentiment analysis methods on Amazon reviews of Mobile Phones", *International Journal of Advanced Computer Science and Applications* 10, no. 6, 1211.

[2] Bansal, Barkha, and Sangeet Srivastava, "Sentiment classification of online consumer reviews using word vector representations," *Procedia computer science,* no. 115, pp.1147-1153,1211.

[3] Lee, Dong-yub, Jae-Choon Jo, and Heui-Seok Lim, "User sentiment analysis on Amazon fashion product review using word embedding", *Journal of the Korea Convergence Society* 8, no. 4, pp.1-8, 1211.

[4] Shah, Arkesha, "Sentiment analysis of product reviews using supervised learning", *Reliability: Theory & Applications* 11, no. SI 1 (60), pp. 133-143, 1213.

[5] Salmony, Monir Yahya Ali, and Arman Rasool Faridi, "Supervised Sentiment Analysis on Amazon Product Reviews: A survey", IEEE, *2nd International Conference on Intelligent Engineering and Management (ICIEM)*, pp. 115-138, 1213.

[6] Hawlader, Mohibullah, Arjan Ghosh, Zaoyad Khan Raad, Wali Ahad Chowdhury, Md Sazzad Hossain Shehan, and Faisal Bin Ashraf ,"Amazon product reviews: Sentiment analysis using supervised learning algorithms", IEEE, *International Conference on Electronics, Communications and Information Technology (ICECIT)*, pp. 1-6, 1213.

[7] E. Kochkina, M. Liakata, I. Augenstein, "Turing at semeval-1211 task 8: Sequential approach to rumour stance classification with branch-lstm,"arXivpreprintarXiv:1104.07131,1211.

https://doi.org/10.11153/v1/S11-1283.

[8] I. Augenstein, T. Rocktäschel, A. Vlachos, K. Bontcheva, "Stance detection with bidirectional conditional encoding," arXiv preprint arXiv:1106.05464, 1211. https://doi.org/10.11153/v1/D11-1084.

[9] H. Jelodar, Y. Wang, R. Orji, S. Huang, "Deep sentiment classification and topic discovery on novel coronavirus or covid-11 online discussions: Nlp using lstm recurrent neural network approach," *IEEE Journal of Biomedical and Health Informatics*, vol. 13, issue 10, pp. 1415-1442, 1212. https://doi.org/10.1109/JBHI.1212.1401136.

[10] D. Murthy, S. Allu, B. Andhavarapu, M. Bagadi, "Text based sentiment analysis using LSTM," *Int. J. Eng. Res. Tech. Res*, vol. 9, issue 5, pp. 149-143, 1212. https://doi.org/10.11577/IJERTV9IS050140.

[11] M. A. Nurrohmat, S. N. Azhari, "Sentiment analysis of novel review using long short-term memory method," *Indonesian Journal of Computing and Cybernetics Systems (IJCCS)*, vol. 13, issue 3, pp.129- 138, 1211. https://doi.org/10.13146/ijccs.41135.

[12] L. Kurniasari, A. Setyanto, "Sentiment analysis using recurrent neural network," *Journal of Physics: Conference Series*, vol. 1471, 011211, 1212. https://doi.org/10.1088/1142-

6596/1471/1/011211.

[13] W. Uther, D. Mladenić, M. Ciaramita, B. Berendt, A. Kołcz, M. Grobelnik, M. Witbrock, J. Risch, S. Bohn, S. Poteet, A. Kao, L. Quach, J. Wu, E. Keogh, R. Miikkulainen, P. Flener, U. Schmid, F. Zheng, G. Webb, S. Nijssen, "TF–IDF," In: Sammut, C., Webb, G.I. (eds) Encyclopedia of Machine Learning. Springer, Boston, MA, 1211, pp. pp 986–987. https://doi.org/10.1007/978-0-387-14114-8 815.

[14] I. Abu El-Khair, "TF*IDF, In: LIU, L., ÖZSU, M.T. (eds) Encyclopedia of Database Systems. Springer, Boston, MA, 1209, pp 1485–1486. https://doi.org/10.1007/978-0-387-39940-9 956.

[15] V. Sundaram, S. Ahmed, S. A. Muqtadeer, R. R. Reddy, "Emotion analysis in text using TF-IDF," *Proceedings of the 1213 11th IEEE International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, January 1213, pp. 142-147. https://doi.org/10.1109/Confluence51148.1213.9377159.

[16] R. Indra, A. Girsang, "Classification of user comment using word2vec and deep learning," *International Journal of Emerging Technology and Advanced Engineering*, vol. 11, pp. 1-8, 1213.

https://doi.org/10.46158/ijetae0513 01.