Date of publication JUN-30, 2022, date of current version MAR-21, 2022.
www.computingonline.net / computing@computingonline.net

Print ISSN 1727-6209
Online ISSN 2312-5381
DOI 10.47839/ijc.21.2.2592

Sentiment Analysis on E-Commerce Apparels using Convolutional Neural Network

KUSUM MEHTA, SUPRIYA P. PANDA
Manav Rachna International Institute of Research and Studies, Faridabad, India

Corresponding author: Kusum Mehta (e-mail: kusumprerak@gmail.com).

ABSTRACT The Fourth Industrial Revolution (4.0) is a fusion of advances in Artificial Intelligence (AI), Robotics, the Internet of Things (IoT), Genetic Engineering, Quantum Computing, and other technologies. A large number of people are using internet-based services as a result of enhanced internet infrastructure and decreased costs. As a result, such businesses' attempts to penetrate internet media are disrupted. The e-commerce company, like Amazon, offers both customer-to-customer and business-to-business services in the apparel sector. Companies must understand the needs of buyers to maximize their profits. As a result, consumer sentiment analysis is carried out. However, because this procedure is time-consuming, it is made automatically utilizing artificial intelligence approaches. According to the findings of a study on sentiment analysis on an E-Commerce-based web store for women, the apparels review dataset using the CNN method with the word vector generator and TF-IDF can produce a higher accuracy of 94%.

KEYWORDS Deep Learning; Convolutional Neural Networks; Sentiment Analysis; Long Short Term Memory (LSTM); Word2Vec; Term Frequency – Inverse Document Frequency (TF-IDF).

I. INTRODUCTION

THE community statement declared in 2020 that there were 624.0 million internet users in India. In contrast to the earlier year, there was a boost of 8.2% or 47 million internet consumers in this nation. Based on the Indian entire populace of 1,391.99 million, it means that 64% of the Indian populace has experienced admittance to cyberspace [1]. One of the main keys to the success of internet penetration in India is the development of infrastructure that has reached remote areas in India. With the growth of information technology, especially online media, the way humans communicate with each other has changed a lot [1]. The use of online media is widely used by the public to express their opinions, experiences, and other things that concern them. Technological developments are currently growing very rapidly, this makes the dissemination of information easier and faster through online media (Facebook, Twitter), blogs, or the official website of an institution [2].

The ease and speed of online media can change the way people consume news. Newspapers, physical magazines are being replaced by online media such as online news, weblogs. In this era and age, citizens desire to devour news to as large extent as possible from various resources that they think are important, or according to their interests [3]. Online media have developed effective approaches to attract people's considerations. Online media articulate opinions about an entity, which can consist of community, places, or even objects when reporting events that occur [4]. For this explanation, interactive emotion ranking services are presented by the different channels of e-commerce and shopping websites, the opinions can be classified as neutral, negative, and positive [5, 6]. Online media construct a lot of diverse kinds of observations and opinions such as materialistic, economic, political, fitness, games, or knowledge. Amongst, materialism and finances are fascinating themes to argue. The materialistic aspect and market has an undeviating blow on citizens, corporations, and even conventional markets depending on the financial conditions in the motherland. Researchers [7] explained that stock trends can be predicted with articles related to stock prices; these articles will positively or negatively affect market share prices. The sentiment contained in the news can affect the public's view of a subject or direction policy by local governments. The subject of money matters is a fascinating subject for investigation since it has an undeviating force on the public. Therefore, this research is devoted to analyzing the sentiment of e-commerce opinions or reviews obtained from the webstore using www.kaggle.com.

Sentiment Analysis (SA) is often recognized as opinion mining, which is used to analyze or classify users from words, sentences, or documents [6]. Several studies researched SA using different models or Emotion Recognition using Deep Learning (DL), Multi-Layer Perceptron (MLP), and Long Short-Term Memory (LSTM) models where the accuracy results of the two mResearch on Pre-processing of Twitter Data models are not much different, namely 71.74% but MLP shows a faster process [7]. Deep Learning has made revolutionary change in our lives due to its prominence in business, products manufacturing, health and entertainment, etc.

The accuracy of using LSTM and Branch LSTM to assess sentiment or remarks on Twitter was just 78 percent [8]. Another study [9] applied the LSTM encoding condition to the processing of social media opinions. The LSTM technique was used to determine public opinion on COVID-19, and the results were 81.15 percent accurate [10]. Likewise, using LSTM researchers were able to get an accuracy of 85% [11].

II. RESEARCH METHODOLOGY

The methodology used in this study consists of several stages, shown in Fig. 1.

This research starts from the literature study stage. Furthermore, the process of defining needs is carried out by identifying the required data, looking at the current procedure, analyzing the current system, and making the results of the evaluation of the system. The second stage is preprocessing; many reviews collect data that includes unstructured language like acronyms, emoticons, symbols, and numbers, which necessitates the use of preprocessing tools. Preprocessing is very important and critical in data mining [12].

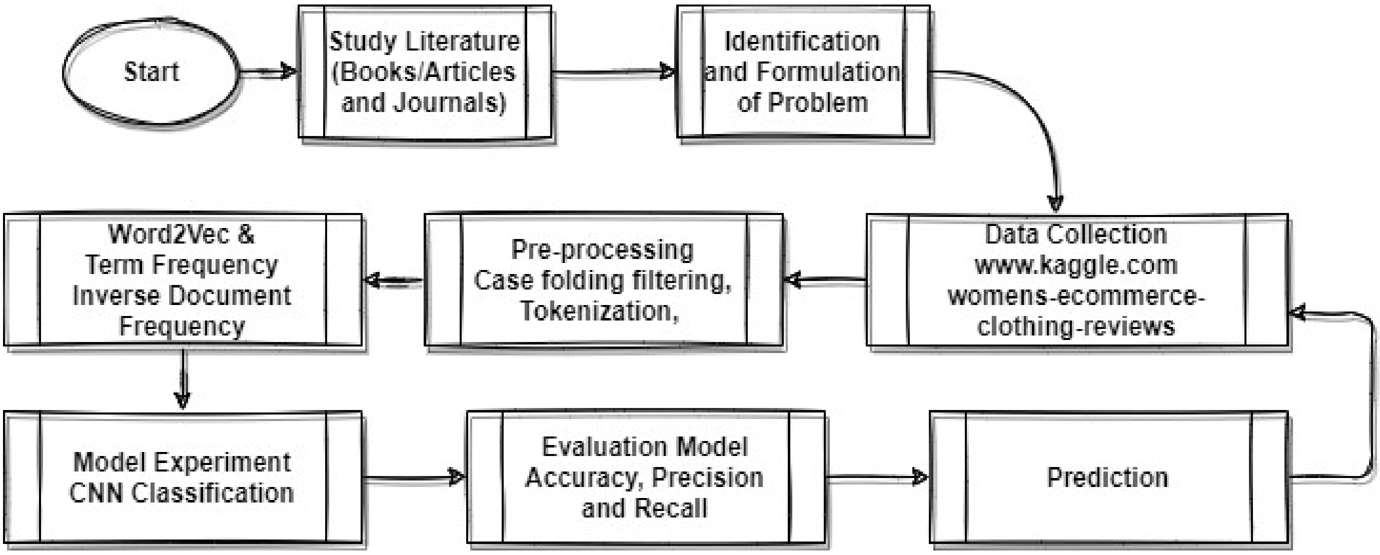

Start
Study Literature (Books/Articles and Journals)
Identification and Formulation of Problem
Data Collection www.kaggle.com womens-ecommerce-clothing-reviews
Pre-processing Case folding filtering, Tokenization,
Word2Vec & Term Frequency Inverse Document Frequency
Model Experiment CNN Classification
Evaluation Model Accuracy, Precision and Recall
Prediction

Figure 1. Research Flow

Following the preprocessing, the next stage is sentence conversion. Creating dictionary terms, translating words to numbers, and padding are all processes in the sentence conversion process. This procedure is followed to provide input to the system [12]. Word2vec and TF-IDF, a word embedding method for expressing words in a vector, is the next step. The next stage contains the system and software design process such as CNN modeling methods. After that, the model evaluation is carried out; the model evaluation is carried out with calculations, the level of accuracy, precision, and recall.

Confusion Matrix is used to measure performance which has several parameters. Confusion Matrix is also known as error matrix. The parameters used are True Positive (TP), False Negative (FN), True Negative (TN), and False Positive (FP) in the training data [13]. At this final stage, the process of documentation and publication of the results of the research has been carried out. As demonstrated in NLP Flowchart & CNN Modeling Flowchart in Fig. 2, this research analyses combinations of algorithms such as NLP (Word2Vector, TF-IDF) and CNN.

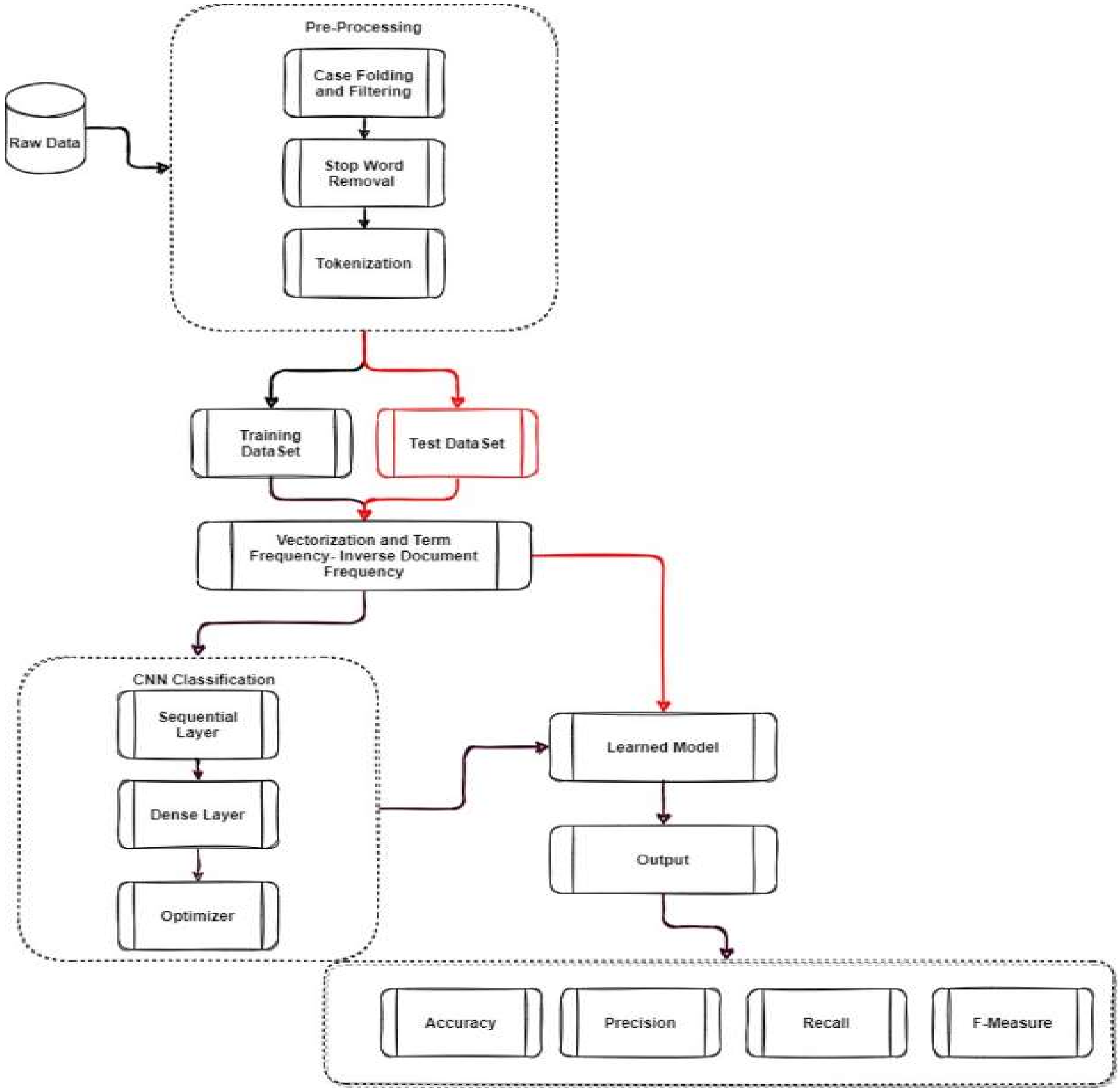


Figure 2. Flowchart Design for Sentiment Analysis using NLP and CNN Modeling

III. RELATED WORK

Some of the concepts used in this investigation are listed below.

- **TF-IDF**

The TF-IDF is used to determine the frequency of an expression in a document as well as its occurrence rate in an article. The Terms Documents Frequency (TDF) and frequency, in particular, are used extensively to rank documents [15]. After considering the word-based model [14-16], the authors propose that words appearing in documents should be scored in proportion to word frequency and inversely proportional to document frequency for the spatial model [14]. A weighting scheme that approximates this approach is called Term Frequency × Inverse Document Frequency (TF × IDF). The procedures for implementing TF-IDF differ slightly in all applications, but the approaches are more or less the same. TF is the frequency with which a word occurs in a document. The value of the TF is obtained using the equation:

$$\mathrm{TF(t)} = \frac{\mathrm{f_{t,d}}}{\sum t, d}, \tag{1}$$

where $\mathrm{f_{t,d}}$is the raw count of a term in a document, i.e., the number of times that term t occurs in document d, while $\sum t, d$ are the total words contained in the document. Then to calculate the IDF, value of a word in a document is determined using the equation:

$$IDF(t) = \log\left(|D|/\mathrm{f_{t,d}}\right), \tag{2}$$

where | D | is the number of documents in the collection, while $f_{t,d}$ is the number of documents where t appears in D (Salton & Buckley, 1988, Berger, et al., 2000). In document collection D, and document individual d ϵ D, the TF-IDF value can be calculated using the formula:

$$TF - IDF\ (t) = TF\ (t) * IDF\ (t). \tag{3}$$

It is assumed that | D | ~ ft,D, the size of the document and is almost the same with document d contains a term t as the relative document frequency. If 1 <log (| D | / ft,D) <c for a constant c with a small value, w (word embedding) is smaller than $f_{t,d}$ but still has a value positive. This means that w (word embedding) has the usual relationship with all documents but still retains some important information. An example is when the TF-IDF method finds the word android in a document containing articles about android. This is also true for some words such as conjunctions which have no relevant meaning against a document. The conjunction is given a low value using TF-IDF. If ft, d are large and ft, D is small, then log (| D | / ft, D) tends to be high, and also TF-IDF (t) will be high. A word with a high TFIDF (t) value means that it is a very strong word important in document d but not in document D.

- **Word2vec**

Word2vec is a useful word embedding method that represents the word to be a vector [17, 18]. For example, a word "India" represents a vector with length 5, namely: [-0.3, 0.6, 0.8., 0.9, -0.2]. A vector not only represents the word syntactically, but also semantically.

- **TensorFlow**

Google published TensorFlow in November 2015, and it has since been used in a variety of Google products, including Google Search, spam recognition, accent detection, Google Supporter, Google Now, and Google Photos. Tensor Flow has the power to do partial sub-graph calculations, thus, enabling distributed training with assistance partition neural networks. As a result, TensorFlow supports parallel processing and data parallelism. TensorFlow provides several APIs. At the lowest level of the API, namely TensorFlow Core provides complete programming control [19-21].

- **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 [22-25]. The architecture of the CNN is depicted in Fig. 3.

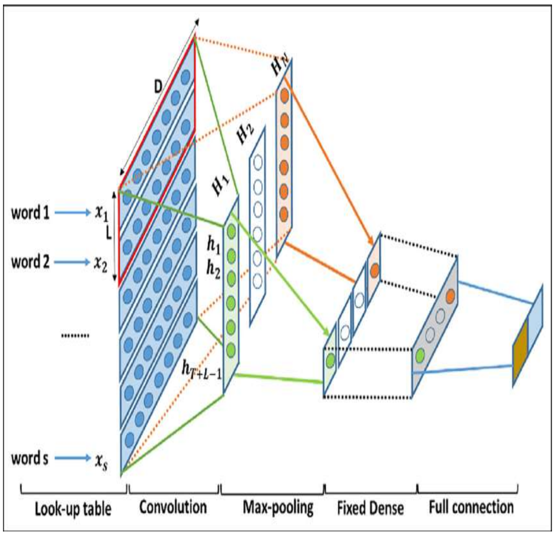


Figure 3. CNN Architecture [37]

Max pooling layer calculates maximum value for every patch of feature map. Dense layer is used for changing dimension of vector by using each neuron and preceding layer result goes to each neuron of dense layer.

- **Data Collection**

The data set used was 23,000 Customer Reviews and Ratings from the article" Women's E-Commerce Clothing Reviews" [26]. In this study, the sampling method is probability sampling with simple random sampling. The amount of data used for the learning phase will be chosen using Slovin's formula [27, 28], and the remaining data will be used for testing. The sample size calculation is based on the Slovin's formula as follows:

$$n = \frac{N}{1 + Ne^2}, \tag{4}$$

where:

N = population size
n = sample size
e = fault tolerance limit.

Machine learning performance training necessitates the creation of train data using Machine Learning algorithms. There are 23,000 reviews or comments available in total. The available data is additionally divided using the Slovin's formula with a significance level of 5% for testing data.

$$n = \frac{23000}{1 + 23000\ 0.05^2} = 5750. \tag{5}$$

Subsequently, as many as 5750 review titles are tested from the 23000 of total data. Thereafter the data preprocessing process is carried out.

- **Data Labeling**

The total amount of data sets used is 23000 with 17250 for training and 5750 for testing. Because a document contains positive or negative ideas and sentiments, labeling is done by assigning negative or positive labels to the data [29]. Three reviewers will be in charge of the labeling. The labeling methodology necessitates the employment of a unique approach, such as the Majority Voting method [30]. The Majority Voting method is the simplest and most widely used of the numerous approaches. This method employs the concept of decision-making as a result of voting gained from the greatest number of each vote option accessible.

- **Labeling**

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

Table 1. Labeling Results

Type of Data	Sentiment	Amount
Data Training	Positive	10158
	Negative	7092
Data Testing	Positive	5213
	Negative	1495

- **Preprocessing**

Preprocessing is performed on the dataset stored in CSV format at this step. This procedure is used to clean the unstructured text data, and the steps include the following:

- **Case folding**

Case folding, or rendering all letters lowercase, is the initial stage in cleaning the data.

- **Filtering**

At this stage, it is done by deleting special characters in the dataset, while special characters that are removed are punctuation symbols (period (.), comma (,), question symbols, exclamation symbols, numeric numbers, and supplementary characters. In this procedure, regulations are prepared by eradicating special typeset in the review as punctuation marks (period (.), comma (,), question mark (?), exclamation point (!) and so on), numeric information (0-9), and other characters ($,%, *, etc.). This process also removes words that do not match the parsed result, such as client names origination with the "@" symbol, hashtags "#", Uniform Resource Locator (URL), and emoticons or this number is omitted because it does not have much effect on labeling [32-35].

- **Tokenization**

This approach divides each word that makes up a text in theory. In general, a space character separates one word from the next; hence the tokenization technique relies on a character space in the text fork to separate words [32, 35]. The sentence becomes a set of arrays after going through the tokenization procedure, with each cell storing the words in the text.

- **Sentence Conversion**

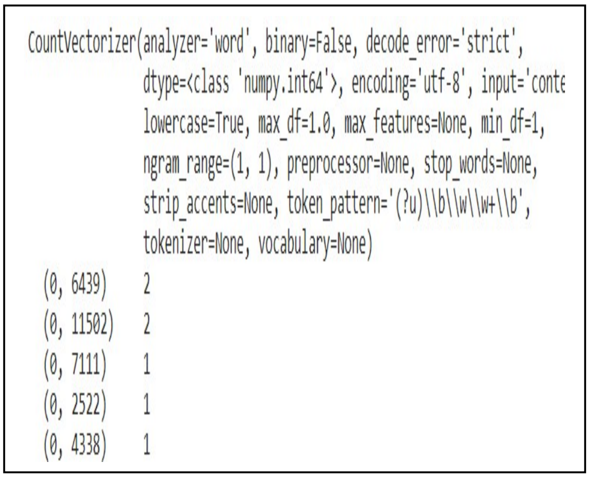
The sentence becomes a set of arrays after going through the tokenization procedure, with each cell storing the words in the text. Table 2 shows the outcomes of the sentence conversion.

Table 2. Sentence Conversion Results

Process	Results
Word Dictionary	['awesome', 'issue', 'boycott', 'stock', 'starbucks', 'already', 'down']
Numeric Conversion	[1063, 491, 1266, 36, 797, 89, 58]
Padding Sequences	range([[1, 0, 0, ..., 191, 59, 56], [2, 0, 1, ...,18, 57, 493], [1, 0,1, ..., 394, 54, 39], ..., [1, 0, 1, ..., 4677, 13, 388], [1, 0, 1, ..., 32, 49, 245], [1, 0, 1, ..., 16, 186, 12]], data type= integer 32)

- **Word2vec**

Word2vec [36] is a word embedding approach that can be used to represent words as a vector. The process of creating a word dictionary was carried out utilizing a set of articles from the Python sklearn module's natural language tool kit. Fig. 4 depicts the Word2vec architectural drawing.



```
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

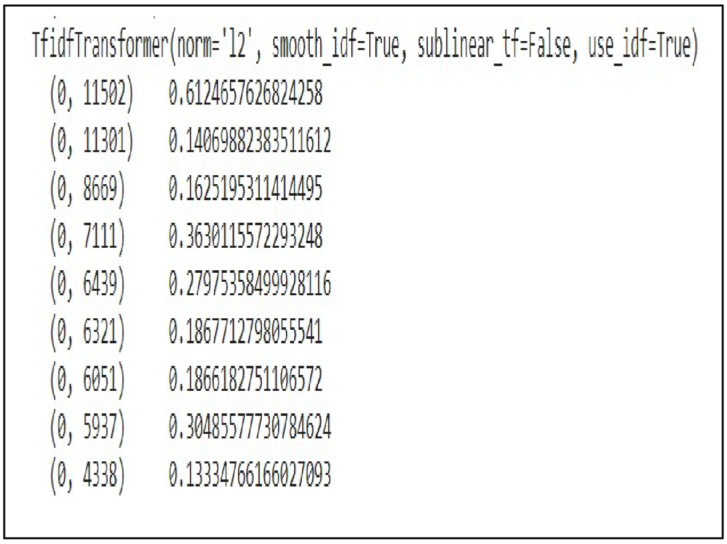
- **TF-IDF**

The computation of the term t in a text is finished in this methodology by multiplying the Term Frequency (TF) value by the Inverse Document Frequency (IDF). There are a number of methods for calculating the TF value, one of which is to use the formula and the results shown below [14-16]:

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

And Inverse Document Frequency can be found with the formula:

$$idf_j = \log\left(\frac{D}{df_j}\right). \tag{7}$$



```
TfidfTransformer(norm='l2', smooth_idf=True, sublinear_tf=False, use_idf=True)
  (0, 11502)	0.6124657626824258
  (0, 11301)	0.1406988238351612
  (0, 8669)	0.1625195311414495
  (0, 7111)	0.3630115572293248
  (0, 6439)	0.27975358499928116
  (0, 6321)	0.186771279805541
  (0, 6051)	0.1866182751106572
  (0, 5937)	0.30485577730784624
  (0, 4338)	0.13334766166027093
```

Figure 5. TF-IDF Model

IV. RESULTS AND DISCUSSION

- **Model Testing**

A training model is carried out at this stage with the goal of testing the model on training data with parameters on each model that has been created. The model with the best parameter value is obtained based on the testing parameters that have been tested: the CNN-Model testing produces accuracy and loss values in training data and validation.

The training process uses 100 epochs and 256 batch sizes. It will be determined how reliable the training data is and what the lowest loss value is using the parameters that have been calculated. The model will store the optimal epoch at its loss value lowest during the epoch process. The following are the model testing results, which may be seen in the figure below. Fig. 6 depicts the CNN model.

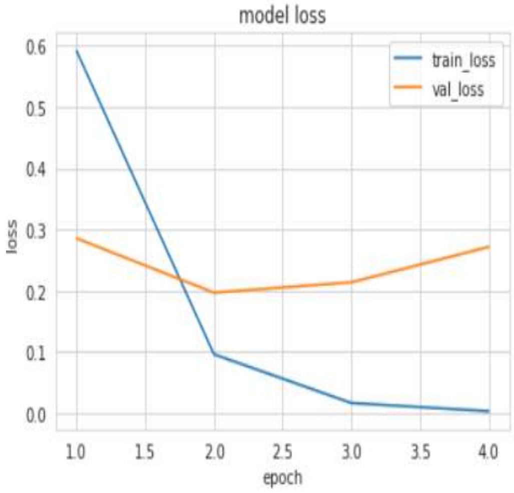


Figure 6. Model Loss Testing using CNN

- **Model Accuracy**

Fig. 7 shows the best findings based on the test image in Fig. 6. Consequently, in the 10th epoch, the CNN model generates the lowest value accuracy 0.90. Thereinafter, at epoch 40th CNN models produce the value accuracy of 0.94 respectively. The comparison demonstrates that the model is pretty good, as the accuracy of data training with validation is not significantly different, indicating that it does not suffer from overfitting.

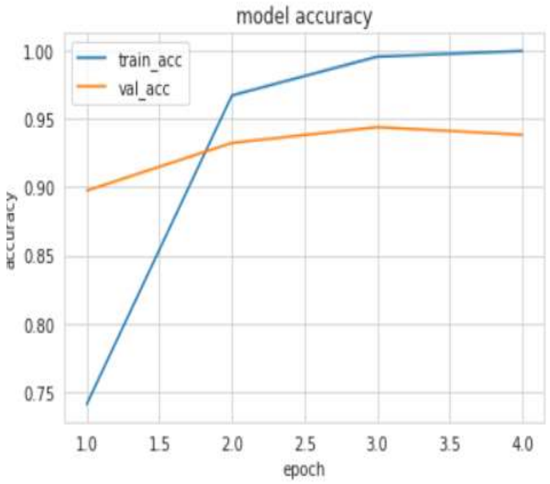
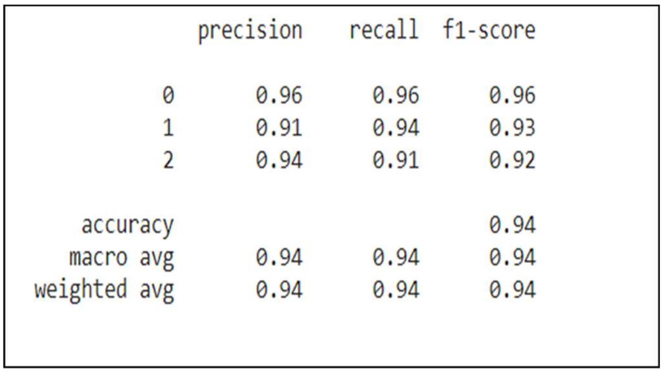


Figure 7. Model Accuracy Testing using CNN

Consequently, based on model accuracy testing, the scheme can evaluate the sentiments of e-commerce product reviews and the model is able to produce the accuracy of 94%.

Data Testing

This stage involves putting the model to the test with real-world data. There are 5750 reviews to be tested with 1495 reviews (15%) in the negative class and 5213 reviews (85%) in the positive class. Furthermore, when the model generates its predictions in each class, the confidence level of the model is calculated by examining the accuracy, precision, and recalls. The goal is to see how well the model is at predicting class. Fig. 8 depicts the outcomes of data testing in the form of 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 its predictions in each class, the model's confidence level is calculated by examining accuracy, precision, and recall. The goal is to find out how much the percentage of accuracy can be trusted as a model in prediction class. Therefore, the proposed scheme achieved the precision of 0.96, 0.96 of recall, and 0.96 of F1-score for positive sentiments. Similarly, for negative sentiments, precision of 0.91, 0.94 recall, and 0.93 F1-score, and for neutral sentiments, precision of 0.94, 0.91 recall, and 0.92 F1-score is achieved. Consequently, the 0.94 of accuracy was achieved by the scheme.

V. CONCLUSION

After going through the stage of testing, the level of performance with parameters of accuracy, precision, and recall on the CNN model in determining the sentiment analysis on the e-commerce apparel products review for women, conclusions may be derived as:

1. The pre-processing strategies are successful in producing well-formed raw data for feature extraction.
2. The data can be normalized for classification using features extraction techniques such as Word2Vector and TF-IDF.
3. The proposed scheme, which employs the Word2Vec, TF-IDF, and CNN models, achieves a 94 percent accuracy rate.

VI. FUTURE SCOPE

In further research, one can add the number of datasets used and perform accuracy comparisons based on a larger number of datasets.

In this study, binary classification is used on the labeling made; three classifications on labeling can be made.

In further research other algorithm models can be combined to get better accuracy results.

References

[1] Digital 2021: India. [Online]. Available at: https://datareportal.com/reports/digital-2021-india

[2] K. K. Kapoor, K. Tamilmani, N. P. Rana, P. Patil, Y. K. Dwivedi, S. Nerur, "Advances in social media research: Past, present and future," *Information Systems Frontiers*, vol. 20, issue 3, pp.531-558, 2018. https://doi.org/10.1007/s10796-017-9810-y.

[3] How Social Media has Changed how we Consume News. [Online]. Available at: https://www.forbes.com/sites/nicolemartin1/2018/11/30/how-social-media-has-changed-how-we-consume-news.

[4] Y. K. Dwivedi, E. Ismagilova, D. L. Hughes, J. Carlson, R. Filieri, J. Jacobson, V. Kumar, "Setting the future of digital and social media marketing research: Perspectives and research propositions," *International Journal of Information Management*, 102168, 2020. https://doi.org/10.1016/j.ijinfomgt.2020.102168.

[5] P. Devadas, Sentiment Analysis using Contextual Approach for E-Commerce Reviews, 2021.

[6] M. H. Munna, M. R. I. Rifat, A. S. M. Badrudduza, "Sentiment analysis and product review classification in e-commerce platform," *Proceedings of the 2020 23rd IEEE International Conference on Computer and Information Technology ICCIT*, 2020, pp. 1-6. https://doi.org/10.1109/ICCIT51783.2020.9392710.

[7] J. Kalyani, P. Bharathi, P. Jyothi, "Stock trend prediction using news sentiment analysis," arXiv preprint arXiv:1607.01958, 2016.

[8] E. Kochkina, M. Liakata, I. Augenstein, "Turing at semeval-2017 task 8: Sequential approach to rumour stance classification with branch-lstm," arXiv preprint arXiv:1704.07221, 2017. https://doi.org/10.18653/v1/S17-2083.

[9] I. Augenstein, T. Rocktäschel, A. Vlachos, K. Bontcheva, "Stance detection with bidirectional conditional encoding," arXiv preprint arXiv:1606.05464, 2016. https://doi.org/10.18653/v1/D16-1084.

[10] H. Jelodar, Y. Wang, R. Orji, S. Huang, "Deep sentiment classification and topic discovery on novel coronavirus or covid-19 online discussions: Nlp using lstm recurrent neural network approach," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, issue 10, pp. 2733-2742, 2020. https://doi.org/10.1109/JBHI.2020.3001216.

[11] 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. 299-303, 2020. https://doi.org/10.17577/IJERTV9IS050290.

[12] 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.209-218, 2019. https://doi.org/10.22146/ijccs.41236.

[13] L. Kurniasari, A. Setyanto, "Sentiment analysis using recurrent neural network," *Journal of Physics: Conference Series*, vol. 1471, 012018, 2020. https://doi.org/10.1088/1742-6596/1471/1/012018.

[14] 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, 2011, pp. pp 986–987. https://doi.org/10.1007/978-0-387-30164-8_832.

[15] I. Abu El-Khair, "TF*IDF, In: LIU, L., ÖZSU, M.T. (eds) Encyclopedia of Database Systems. Springer, Boston, MA, 2009, pp 3085–3086. https://doi.org/10.1007/978-0-387-39940-9_956.

[16] V. Sundaram, S. Ahmed, S. A. Muqtadeer, R. R. Reddy, "Emotion analysis in text using TF-IDF," *Proceedings of the 2021 11th IEEE International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, January 2021, pp. 292-297. https://doi.org/10.1109/Confluence51648.2021.9377159.

[17] 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, 2021. ttps://doi.org/10.46338/ijetae0521_01.

[18] V. K. Ayyadevara, "Word2vec," *Pro Machine Learning Algorithms*, Apress, Berkeley, CA, 2018, pp. 167-178. https://doi.org/10.1007/978-1-4842-3564-5_8.

[19] A. K. Jha, A. Ruwali, K. B. Prakash, G. R. Kanagachidambaresan, "Tensorflow basics," In: Prakash, K.B., Kanagachidambaresan, G.R. (eds) *Programming with TensorFlow. EAI/Springer Innovations in Communication and Computing*, Springer, Cham, 2021, pp. 5-13. https://doi.org/10.1007/978-3-030-57077-4_2.

[20] I. Hull, I. "TensorFlow 2," In: *Machine Learning for Economics and Finance in TensorFlow 2*, Apress, Berkeley, CA, 2021, pp. 1-59. https://doi.org/10.1007/978-1-4842-6373-0_1.

[21] N. Silaparasetty, "Programming with Tensorflow," In: *Machine Learning Concepts with Python and the Jupyter Notebook Environment*, Apress, Berkeley, CA, 2020, pp. 173-189. https://doi.org/10.1007/978-1-4842-5967-2_9.

[22] D. Basler, Convolutional Neural Networks, 2021. https://doi.org/10.3139/9783446464261.006.

[23] N. Ketkar, J. Moolayil, N. Ketkar, J. Moolayil, "Convolutional neural networks," *Deep Learning with Python: Learn Best Practices of Deep Learning Models with PyTorch*, 2021, pp.197-242. https://doi.org/10.1007/978-1-4842-5364-9_6.

[24] Y. V. R. Nagapawan, K. B. Prakash, G. R. Kanagachidambaresan, "Convolutional Neural Network," In: *Programming with TensorFlow*, Springer, Cham, 2021, pp. 45-51. https://doi.org/10.1007/978-3-030-57077-4_6.

[25] D. Paper, "Convolutional neural networks," In: *TensorFlow 2.x in the Colaboratory Cloud*, Apress, Berkeley, CA. 2021, pp. 153-181. https://doi.org/10.1007/978-1-4842-6649-6_7.

[26] Women's E-Commerce Clothing Reviews. Dataset. [Online]. Available at: https://www.kaggle.com/nicapotato/womens-ecommerce-clothing-reviews

[27] Slovin's Formula: What is it and When do I use it? [Online]. Available at: https://www.statisticshowto.com/how-to-use-slovins-formula/

[28] A. M. Adam, "Sample size determination in survey research," *Journal of Scientific Research and Reports*, pp. 90-97, 2020. https://doi.org/10.9734/jsrr/2020/v26i530263.

[29] R. Cowie, C. Cox, J. C. Martin, A. Batliner, D. Heylen, K. Karpouzis, "Issues in data labelling," In: *Emotion-oriented Systems*, Springer, Berlin, Heidelberg, 2011, pp. 213-241. https://doi.org/10.1007/978-3-642-15184-2_13.

[30] M. Desmond, M. Muller, Z. Ashktorab, C. Dugan, E. Duesterwald, K. Brimijoin, C. Finegan-Dollak, M. Brachman, A. Sharma, N. N. Joshi, Q. Pan, "Increasing the speed and accuracy of data labeling through an AI assisted interface," *Proceedings of the 26th International Conference on Intelligent User Interfaces*, 2021, pp. 392-401. https://doi.org/10.1145/3397481.3450698.

[31] R. Lischner, "Case-folding," In: *Exploring C++ 11*, Apress, Berkeley, CA, 2013, pp. 111-113. https://doi.org/10.1007/978-1-4302-6194-0_19.

[32] A. Kulkarni, A. Shivananda, "Advanced natural language processing," In: *Natural Language Processing Recipes*, Apress, Berkeley, CA, 2019, pp. 97-128. https://doi.org/10.1007/978-1-4842-4267-4_4.

[33] C. Ng, J. Alarcon, *Artificial Intelligence in Accounting: Practical Applications*, Routledge, 2020. https://doi.org/10.4324/9781003003342.

[34] U. Qamar, M. S. Raza, "Text mining," In: *Data Science Concepts and Techniques with Applications*, Springer, Singapore, 2020, pp. 133-151. https://doi.org/10.1007/978-981-15-6133-7_7.

[35] H. Hassani, C. Beneki, S. Unger, M. T. Mazinani, and M. R. Yeganegi, "Text Mining in Big Data Analytics," Big Data and Cognitive Computing, vol. 4, no. 1, p. 1, 2020, https://doi.org/10.3390/bdcc4010001.

[36] G. Di Gennaro, A. Buonanno, F. A. Palmieri, "Considerations about learning Word2Vec," The Journal of Supercomputing, vol. 77, issue 11, pp. 1-16, 2021.

[37] V. Q. Nguyen, T. N. Anh, and H.-J. Yang, "Real-time event detection using recurrent neural network in social sensors," International Journal of

Distributed Sensor Networks, vol. 15, no. 6, p. 155014771985649, 2019, https://doi.org/10.1177/1550147719856492.

***KUSUM MEHTA** received her Master's Degree in Computer Science and Engineering in the year 2006. Currently, she is doing her Ph.D. in the area of Big Data under the guidance of Dr. Supriya P. Panda in the Department of Computer Science and Engineering at Manav Rachna International Institute of Research and Studies, Faridabad, India.*

***DR. SUPRIYA P. PANDA** received her Doctorate in Computer Science in the year 1990. She has around thirty four years and 9 months of Academic experience in the field of Computer Science. Currently she is working as the Professor and Head of the Department of Computer Science and Engineering at Manav Rachna International Institute of Research and Studies, Faridabad, India. She has been the best Teaching Fellow at BGSU, Ohio, USA during MS and Ph.D (1985-90).*