Araştırma Makalesi
Adıyaman Üniversitesi Mühendislik Bilimleri Dergisi
17 (2022) 447-457

MULTI-LABEL TEXT ANALYSIS WITH A CNN AND LSTM BASED HYBRID DEEP LEARNING MODEL

Halit ÇETİNER[1*]

[1]Isparta University of Applied Sciences, Vocational School of Technical Sciences, Isparta, Turkey
Geliş Tarihi/Received Date: 21.04.2022 Kabul Tarihi/Accepted Date: 04.06.2022 DOI: 10.54365/adyumbd.1106981

ABSTRACT

In this article, it is aimed to categorize meaningful content from uncontrolled growing written social sharing data using natural language processing. Uncategorized data can disturb social sharing users with an increasing user network due to deprecating and negative content. For the stated reason, a hybrid model based on CNN and LSTM has been proposed to automatically classify all written social sharing content, both positive and negative, into defined target tags. With the proposed hybrid model, it is aimed at automatically classifying the content of the social sharing system into different categories by using the simplest embedding layer, keras. As a result of the experimental studies carried out, a better result was obtained than in the different studies in the literature using the same data set with the proposed method. The obtained performance results show that the proposed method can be applied to different multilabel text analysis problems.

Keywords: *Long Short Term Memory, Convolutional Neural Network, Multi-Label Text Classification, Social Network*

CNN VE LSTM TABANLI HİBRİT BİR DERİN ÖĞRENME MODELİ İLE ÇOK ETİKETLİ METİN ANALİZİ

ÖZET

Bu makalede doğal dil işleme kullanılarak kontrolsüz olarak büyüyen yazılı sosyal paylaşım verilerinin içerisinden anlamlı içeriklerin kategorize edilmesi amaçlanmıştır. Kategorize edilmeyen verilerin, artan kullanıcı ağına sahip sosyal paylaşım kullanıcılarını olumsuz ve negatif içerikten dolayı rahatsız edebilmektedir. Belirtilen sebepten dolayı olumlu ve olumsuz olmak üzere tüm yazılı sosyal paylaşım içeriklerinin tanımlı hedef etiketlerine otomatik olarak sınıflandırılabilmesi için CNN ve LSTM tabanlı bir hibrit model önerilmiştir. Önerilen hibrit model ile en basit gömme katmanı olan keras kullanılarak farklı kategorilere sahip sosyal paylaşım sistemi içeriklerinin otomatik sınıflandırılması hedeflenmiştir. Gerçekleştirilen deneysel çalışmalar neticesinde önerilen yöntem ile aynı veri setini kullanan literatürdeki farklı çalışmalardan daha iyi bir sonuç elde edilmiştir. Elde edilen performans sonuçları önerilen yöntemin farklı çok etiketli metin analizi problemlerine de uygulanabileceği göstermektedir.

Anahtar Kelimeler: *Uzun Kısa Süreli Bellek, Konvolüsyon Sinir Ağı, Çok Etiketli Metin Sınıflandırma, Sosyal Ağ*

1. Introduction

As a social being, human beings use social networking accounts such as WhatsApp, Facebook, Wikipedia, Instagram, Twitter to maintain their past or present friendships and establish new relationships. The number of users in cheap and fast information sharing systems, especially in the mentioned social sharing systems, is increasing day by day [1]. The increase in the use of social sharing systems also increases data sharing. Within these data increases, professional fake news content can be

* e-posta[1] : halitcetiner@isparta.edu.tr ORCID ID: https://orcid.org/0000-0001-7794-2555 (Sorumlu Yazar)

produced, and it is seen that data that can be considered negative in content are quickly shared on social platforms [2].

Insecurity in data and news on social sharing systems affects users negatively, as well as negatively affecting comments with negative and obscene content. Such content can cause changes in user attitudes [3]. The main reason for the increase in such content is that social sharing systems, which have the ability to easily influence people, are out of control. Uncontrolled sharing of negative thoughts and behaviors on social media causes such useless thoughts to increase. This article provides an automatic categorization of classes in the comments of a particular social networking system on the Internet. Thanks to the automatic categorization and classification process, users will be informed before reading the message content. In line with this purpose, as a result of experimental studies on user comments of the Wikipedia social networking system, automatic classification of content with one or more target tags has been achieved over the same data.

Multi-label text classification refers to dealing with multiple target labels simultaneously [4]. In multilabel text classification approaches, deep learning algorithms give better results than classical LDA [5] and KNN [6] algorithms for different reasons. LDA is an algorithm that uses information contained in unlabeled content to augment the small set of unlabeled samples at the beginning of the method's run [5]. Based on the traditional K Nearest Neighbor (KNN) algorithm, the proximity values of texts are measured [6]. CNN structures, unlike these classical structures, perform one-dimensional filtering of the vectors of words in a sentence. It can use different filtering sizes while doing this [7]. In the first of these, automatic feature extraction processes are performed in deep learning algorithms instead of classical machine learning algorithms [8]. The second is that it can be easily used in large-scale data sets [9]. For the stated reasons, researchers who want to offer the best solution to the multilabel text classification problem have turned to deep learning methods [10]. Existing word, word and translation applications are being replaced by deep learning models instead of machine learning-based methods [11]. When the performance metrics obtained as a result of these transformations are examined, it is stated that deep learning-based models give better results than classical machine learning methods [12].

Deep learning models proposed in the literature to solve multilabel text classification problems are examined. It is seen that a deep learning model consisting of 29 convolution layers, also known as the feature extraction layer, is assembled in succession [13]. Similar to this developed model, [14] proposed a model consisting of five successive convolution layers. Similarly to the models mentioned, a deep learning model consisting of convolution and maximum pooling layers, which is considered the most important layer of deep learning, has been proposed [15]. Kim performed text classification with the help of Word2Vec pretrained networks using 1D CNN structures [15]. Developed a nonstatic text classification model using 1D CNN [14]. It also performs text classification with two multichannel pretrained word embedding structures. Zhou et al. proposed a model that uses CNN and RNN structures together for text classification [16]. In natural language processing applications, an effective increase in performance criteria is observed when the parameters that make up deep learning models are appropriately adjusted [17–19]. In this article, a hybrid method is proposed by combining CNN and LSTM [9,20] methods that are frequently used together with natural language processing methods. For this purpose, the parameters of the model, which combine the strengths of CNN and LSTM methods, have been determined by experimental studies.

Deep learning models based on natural language processing and recurrent neural networks are used in multilabel text classification. In a data set open to researchers, preprocesses such as cleaning numbers, converting them to lowercase, removing spaces, and special characters were applied to bring all texts to the same standard. The preprocessed text chunks are divided into training and testing. The text chunks are trained with the help of pre-trained networks using the Keras embedding layer. As a result of the test data given to the model created with the help of trained networks, test performance results were obtained. The performance results obtained as a result of the training and testing processes are presented using metrics commonly used in the literature. The contributions of the study to the literature as a result of the processes and the proposed models are given below.

- CNN and LSTM deep learning frameworks are combined to perform effective multilabel text analysis by emphasizing the structure of words in social sharing comments.

- CNN and LSTM deep learning models were compared in terms of F1 score, precision, recall, and accuracy metrics.
- A light hybrid model is proposed due to the large data set used and the long training time.
- The comparison of the proposed deep learning model with the studies in the literature was made according to general performance metrics such as F1 score, precision, recall, and accuracy.
- The automatic classification of inappropriate Wikipedia comments was carried out in a very short time with the proposed models.

The next steps of the article consist of three parts. In the first part, information is given about the material to be used to analyze and classify Wikipedia comments. At the same time, the foundations of deep learning architectures used within the scope of this article are given. In the second part, the results obtained according to the general performance metrics are given. In the last part, the study is concluded.

2. Material and Methods

In the Material and Methods section, information is given about the Wikipedia dataset used within the scope of the article. After the data set used is detailed, the theoretical foundations of the deep learning models that will train this data set are examined.

2.1. Material

Data sets that can be determined by experts to determine which text belongs to which class are rare [21]. As a result of the research, a Wikipedia user comment dataset, which is open to all researchers and tagged by text experts, was used [22]. This data set consists of 20,000 words with six different class labels [9]. The data set used was divided into three parts, 80% training, 10% testing, and 10% validation data. Ninety percent of the class labels in the data set do not have any class labels. While 9.58% of the rest belong to the toxic class, the rest belong to other classes.

In the data set used, there are six different target tags: toxic, severe toxic, obscene, threat, insult, and identity hate. The toxic, severe toxic, obscene, threat, insult, and identity hate values defined as the target tag are the tag names given according to the content in the comments. Separation of the data sets used according to the target tags was carried out by an artificial intelligence team of 5000 people [23]. The data set with the specified target tags is an unbalanced data set. This uneven distribution is shown in Table 1. Apart from the distributions shown in Table 1, there are 124473 unclassified observations.

Table 1. Dataset label distributions [23]

Target Tags	Number of observations
Toxic	15294
Severe Toxic	1595
Obscene	8449
Threat	478
Insult	7877
Identity Hate	1405

A comment can have multiple tags at the same time. In this case, the problem changes from a multiclass text problem to a multilabel text problem. Natural language processing and CNN and LSTM methods from deep neural networks were used in the classification process of defined target classes within the specified scope. In the multilabel text classification problem, firstly, preprocesses such as cleaning the numbers, converting to lowercase, removing spaces and special characters are applied to bring all texts in the data set to the same standard. Afterwards, the preprocessed text chunks are divided into training and testing. In the third operation, the text chunks are trained with the help of pre-trained networks using the keras embedding layer. This training process also allows one to draw result graphs directly from the top of the curve in accuracy graphs. Instead of training and test graphics from zero and near zero points, it is ensured that performance results are obtained from above in the first step.

2.2. Methods

2.2.1.LSTM

Feedforward neural networks with hidden states and whose hidden states are triggered by previous states at a given time are defined as RNN architecture [20]. For the stated reason, RNN architectures can model contextual information by processing sequences of variable length [24]. LSTM methods are a method that offers a solution to the disappearing gradient problem in RNN structures. It gives better results than other RNN-based models in long-term recall of the data to be remembered [25,26]. LSTM has a chain structure in time series with sequential data entry [21]. As in the the GRU structure, in LSTM structures, the information from outside is represented by the symbols x_t, the structure h_{t-1} expressing the precurrent state, and the current state h_t.

$$f_t = \sigma\left(W_f X_t + U_f h_{t-1} + b_f\right) \tag{1}$$

$$i_t = \sigma(W_i X_t + U_i h_{t-1} + b_t) \tag{2}$$

$$o_t = \sigma(W_o X_t + U_o h_{t-1} + b_o) \tag{3}$$

$$g_t = tanh\left(W_g x_t + R_g h_{t-1} + b_g\right) \tag{4}$$

In Equation 1-4, W represents weight value, R represents repetitive weight value and b represents bias. The terms f_t, i_t, and o_t in Equations 1-4 represent forgetting, input and output gates at time t, respectively. g_t represents the state layer. The standard LSTM uses only the historical context, not the future temporal context [20].

2.2.2.CNN

Although fully connected neural networks are known to be good at classification problems, they can cause many connectivity problems. Sentences that are far from each other should not be evaluated together with sentences that are close to each other. However, fully connected neural networks treat far and near sentences alike. In order to solve this problem, CNN methods have been used in multi-label text analysis. In this structure, instead of connecting all input neurons between the layers, a small structure is connected. As a result of this fastening process, there is a great reduction in calculation costs.

While RNN is used to recognize words in time series, CNN is trained to recognize words in space [27]. CNN in text classification problems was first proposed in [14]. The maximum pooling is used, which selects the largest value in hovering windows over digitized text data. It is stated that a word matrix is created for each word in the embedding layer they use. Subsequently, a feature map was created

that contains short- and long-term relationships of digitized and weighted sentences from the embedding layer, the convolution layer, and the maximum pooling layer. Finally, text classification was performed on target tags using the sigmoid activation function.

In the convolution process, to detect the edges of an object in the image or to determine the distinguishing features in the text, the filters must be navigated on the basis of rows and columns within the specified area. During the filtering process, the values defined in the convolution kernel are multiplied by the values in the hovering area, and the convolution operation is performed. The stated situation is expressed in Equation 5. I in Equation 5 shows the data matrix. A filter square matrix named K of size [i,j] is circulated over the defined data matrix. After this navigation process, a new I*K data matrix of [x,y] dimension is obtained.

$$(I * K)_{xy} = \sum_{i=1}^{h} \sum_{j=1}^{w} K_{ij} \cdot I_{x+i-1,y+j-1} \tag{5}$$

The formula representing the most important layer of CNN structure is shown in Equation 5. After the convolution layer, layers such as batch normalization, dropout, density, and fully connected are applied in a certain order and successively [28]. The parameters and ordering of these layers can change the classification success rates.

2.2.3. CNN and LSTM Based Hybrid Model

In this section, the structures of two different deep learning architectures based on CNN and LSTM are combined to classify Wikipedia data. The hybrid model layer structure created by combining, the number of steps, the performance metrics obtained, and the running times are presented in this section. The number of layers, structure and features of the model are set to be the same in order to be able to compare the obtained model correctly.

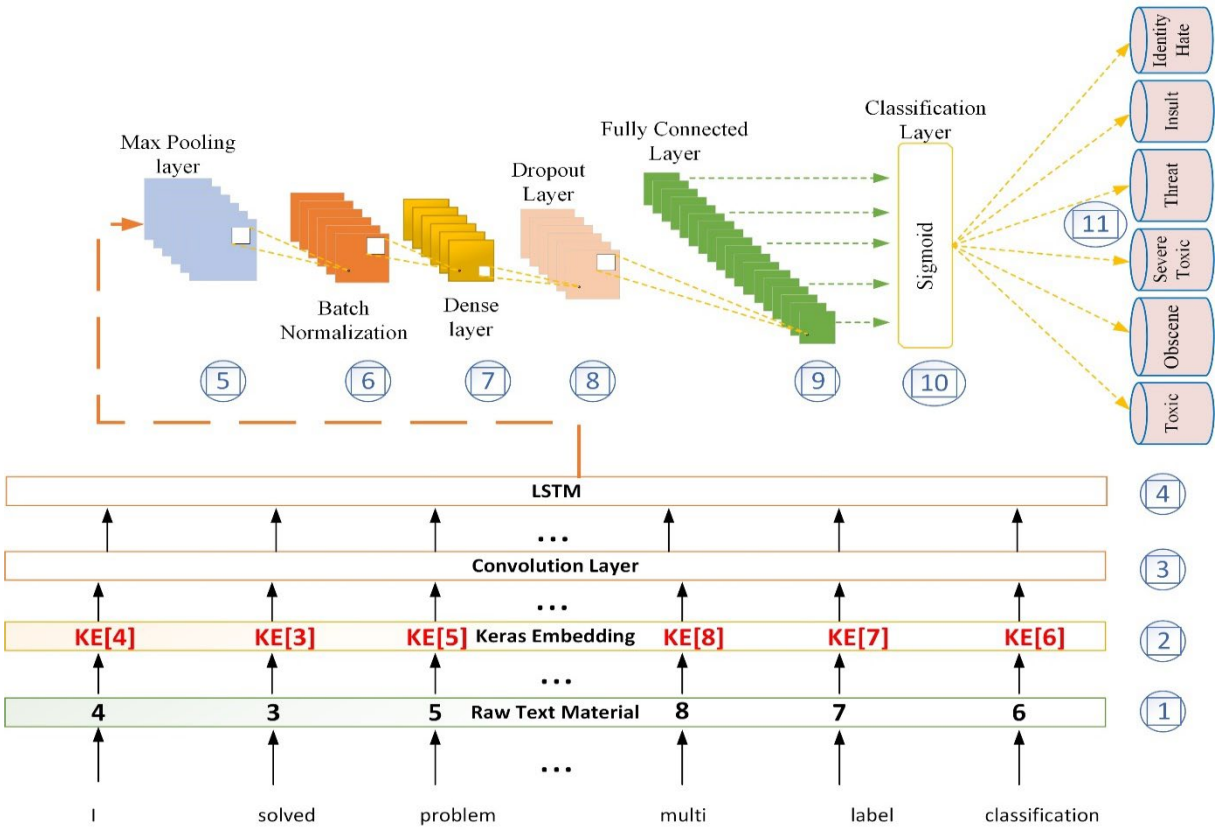

Figure 1. Proposed CNN LSTM deep learning model

The maximum word length that the network can process is 20,000, the maximum attribute is 200,000, and the batch size value is 256. Training data and target data are set to fill margins up to 200 sentences long. The proposed hybrid CNN LSTM deep learning model is shown in Figure 1. In the first layer of the model seen in Figure 1, raw text data are given as input. In the second layer, a layer of 148243 long keras embedding layer is defined. In this layer, the words are weighted. Due to this process, words are digitized and kept as vectors.

In the third layer, a convolution layer with 256 filters with ReLU activation function is defined using 4x4 windows. In this layer, many filters with different window sizes act on word placements to perform one-dimensional convolution. As the filters move, semantic, contextual and syntactic meanings are captured and many sequence structures are created. By combining these structures, feature maps of word embeddings are created. In this layer, although sentences of different lengths are taken as input, it is ensured that vectors of fixed length are obtained as output.

In the fourth layer, a 128 neuron LSTM layer was added. In this layer, the semantic data of the feature maps are kept in the memory for a longer time. In the fifth layer, the most distinctive local features on the feature map were captured. In the sixth layer, vector-type features were converted to two-dimensional matrix by batch normalization. In the seventh layer, there is the den layer that connects with the previous feature networks. In the dense layer with 256 neurons, a full connection was achieved to the next layer with the previous layers. In the ninth layer, the data from the forgetting layer to the eighth layer are made into a one-dimensional array.

In the tenth layer, the sigmoid activation function was applied to the data from the ninth layer. This activation function is shown in Equation 6 [29]. The z in the equation represents the values in the fully connected layer coming from the classification layer. These values are brought to a certain scale. After this layer, the result of multi-label text classification is obtained. A value is formed for each target tag value. Comparison with the actual value can be made using the estimated value above a certain threshold value.

$$\sigma(z) = \frac{1}{1 + e^{-z}} \quad , z \in \mathbb{R} \tag{6}$$

3. Experimental Results and Discussion

Experimental studies were carried out using tensorflow libraries in the Python 3.8 environment. Proposed models are run using GPU on a graphics card with NVIDIA GeForce RTX 3060 version. The Adam optimization method was used in the six iteration trials of the layer structure detailed in Figure 1. The man optimization method was run with a learning rate of 0.01. As a hyper parameter, the best results were obtained with batch size 256 value. Performance evaluations of the proposed model were performed according to the formulas of Equations 7-10. The result values obtained from the deep learning model are given in Table 2. The measurement results given were calculated according to the basic formulas in the study [30].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{7}$$

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

$$Precision = \frac{TP}{TP + FP} \tag{9}$$

$$F1 = 2x \frac{PrecisionxRecall}{Precision + Recall} \tag{10}$$

In Figure 2, the training accuracy and loss graphs and test accuracy and loss graphs are presented. In the hybrid CNNLSTM model proposed in Figure 2, it is seen that it starts from high level training and test accuracy rates with the effect of keras embedding layer. The F1 score, recall, and precision values of the performance results obtained are presented in Table 2. In the article, the Adam optimization method was used to obtain the performance results according to the formulas in Equations 7-10. Among the reasons for using this is that although the SGD optimization method has a fast gradient descent, it is difficult to reach the targeted values such as precision, F1 score, recall, and accuracy. Rapid gradient descent, which increases the learning speed of deep learning models, has problems with overshooting the slope. In addition to these, RMSProp calculates the learning rate by dividing the exponentially decreasing square gradient mean and decreases the learning rate as the gradients approach the minimum.

Due to the disadvantages of SGD and RMSProp optimization methods, Adaptive Moment Estimation (Adam) [31] optimization method was used as an optimization method in the training processes in this article. The Adam optimization method proceeds by calculating an exponentially decreasing gradient mean, such as momentum, as the gradient can decrease exponentially with the square mean [32]. For these reasons, the Adam optimization method was used as a single optimization method in the multilabel text analysis of Wikipedia data.

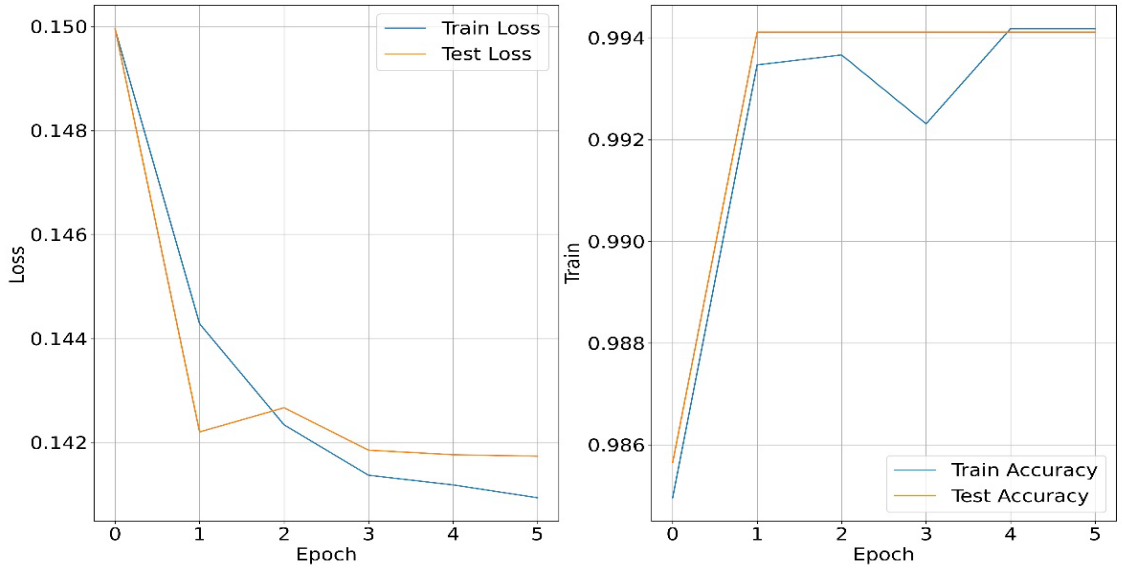

Figure 2. Proposed CNN-LSTM model training, test accuracy and loss plot

When Table 2 is examined, it is seen that the training and test results are very close to each other and the loss values are plotted in a way that is proportional to the accuracy values. The results obtained show that the proposed study has a sufficient success rate. Performance analysis can be detailed by comparing these results with different studies using the same data set in the literature.

Table 2. Performance results of the proposed hybrid CNNLSTM model

Model	Precision	Recall	F1 score	Accuracy
CNNLSTM model (training)	0.8764	0.7976	0.8213	0.9942
CNNLSTM model (test)	0.8681	0.7950	0.8112	0.9941

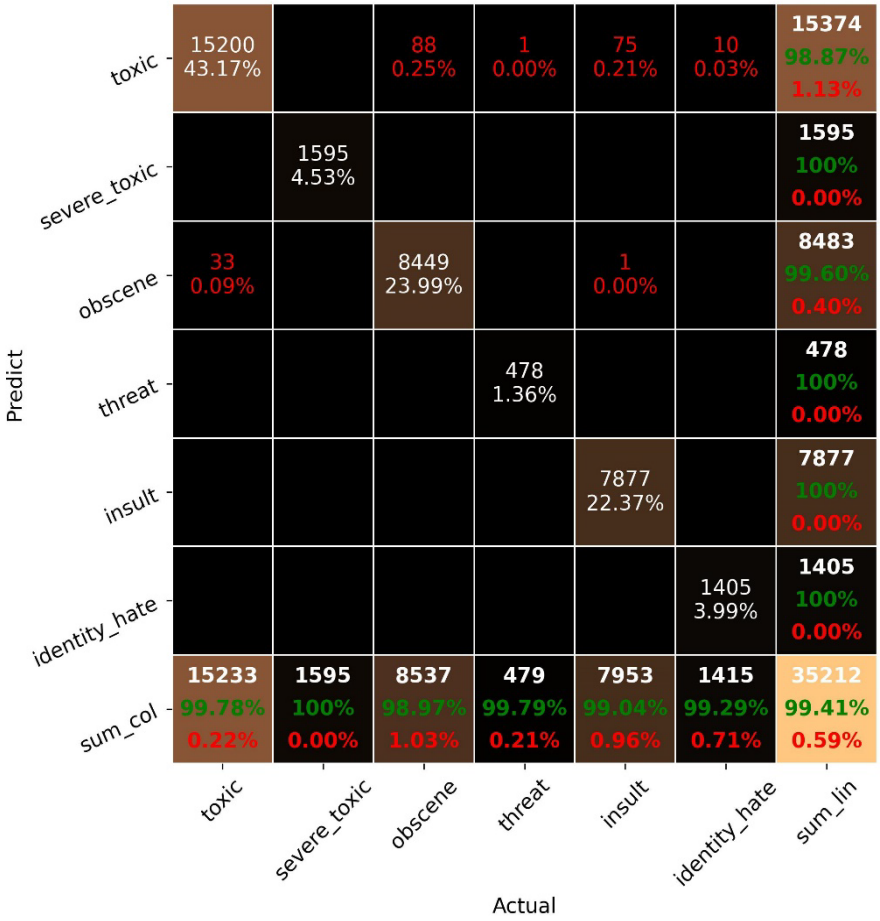

Figure 3. The confusion matrix of proposed model

Class-based accuracy rates obtained with the test data of the proposed model are shown in Figure 3. According to the proposed model, 98.87%, 100%, 99.60%, 100%, 100%, 100%, 100% success rates were achieved in the classification of toxic, severe_toxic, obscene, threat, insult, identity_hate target tags, respectively. The average success rate was determined as 99.41%.

When different studies using the same data set are examined, it can be directly decided whether the results obtained are good or bad. In this sense, the performance results of deep learning models using different word placement algorithms to classify the Wikipedia dataset are presented in Table 3. In CNN, LSTM and BiLSTM deep learning models, it is observed that GloVe word placement algorithm gives much better results than random word placement algorithm [33]. They have shown that they will be more successful when the word sequences have weight in a certain order. At the same time, it is stated that word placement algorithms with a certain starting point can be considered to be more successful [33].

Although LSTM models generally give good results in natural language processing problems, the authors express that they are surprised that the CNN model gave as successful results as the LSTM model in their study. Although LSTM is successful in solving the vanishing gradient problem, it is mentioned that the CNN method accurately captures local dependencies at different points of the inputs. The BiLSTM model, on the other hand, is shown to have more layers than other models as the main reason for its success. At the same time, it was reported that after the number of layers in the BiLSTM model increased, the training time increased significantly compared to the CNN model. Furthermore, the CNN model is stated to be more prone to overfitting than the BiLSTM model. It is stated that when the BiLSTM model is used with GloVe word insertion, it provides better results than the LSTM and CNN methods.

Table 3. Model performance results with different Word Embeddings

Model	Word Embeddings	Precision	Recall	F1 score	Accuracy
[33] CNN	GloVe	-	-	-	0.9174
[33] LSTM	GloVe	-	-	-	0.9396
[33] BiLSTM	GloVe	-	-	-	0.9756
[33] CNN	Random	-	-	-	0.9145
[33] LSTM	Random	-	-	-	0.8841
[33] BiLSTM	Random	-	-	-	0.9222
[34] CNN	FastText	0.86	0.83	0.778	0.904
[34] CNN	GloVe	0.85	0.76	0.795	0.912
[34] CNN	Word2Vec	0.82	0.73	0.739	0.889
[34] LSTM	FastText	0.87	0.75	0.796	0.917
[34] LSTM	GloVe	0.86	0.76	0.803	0.930
[34] LSTM	Word2Vec	0.84	0.73	0.744	0.896

Another study in which the results obtained with the hybrid CNNLSTM model were compared in the article is [34]'s study. According to this study, the LSTM method is stated to give a better result than the CNN and GRU methods. The fluctuation in the results is stated to be due to the fluctuation in the Wikipedia data set used in this article. It is declared that the use of pretrained weight methods does not affect the results too much. The F1 score penalizes incorrect class predictions in proposed models. The study [34] performed better than the study [33] in terms of the F1 score.

4. Conclusions

With an increase in the number of Internet users, there is a great increase in the use of social sharing systems. This increase can have positive as well as negative sides. One of the negative aspects is that user comments must be manually or automatically classified according to different class tags. A CNN LSTM-based hybrid deep learning model has been developed to perform this process. With the proposed deep learning model, a 99.41% test success rate has been achieved. The success rate was compared with studies [34] and [33] available in the literature. As a result of word insertion using the Keras embedding layer, the hybrid CNNLSTM method outperformed all accuracy results of the study [29]. At the same time, the proposed study provided a good performance by providing better results in terms of precision, recall, F1 score, and study precision [34]. In this study, the parameters of the CNNLSTM model were also determined by experimental studies. In future academic research, an academic study will be carried out to automatically determine the parameters of the deep learning model with the best results with optimization methods.

References

[1] Sahoo SR, Gupta BB. Multiple features based approach for automatic fake news detection on social networks using deep learning. Appl Soft Comput 2021;100:106983. doi:10.1016/j.asoc.2020.106983.

[2] Horne B, Adali S. This just in: Fake news packs a lot in title, uses simpler, repetitive content in text body, more similar to satire than real news. Proc. Int. AAAI Conf. web Soc. media, 2017; 11: 759–66.

[3] Balmas M. When fake news becomes real: Combined exposure to multiple news sources and political attitudes of inefficacy, alienation, and cynicism. Communic Res 2014;41:430–54.

[4] Liu B, Liu X, Ren H, Qian J, Wang Y. Text multi-label learning method based on label-aware attention and semantic dependency. Multimed Tools Appl 2022;81:7219–37. doi:10.1007/s11042-021-11663-9.
[5] Pavlinek M, Podgorelec V. Text classification method based on self-training and LDA topic models. Expert Syst Appl 2017;80:83–93. doi:10.1016/j.eswa.2017.03.020.
[6] Feng Y, Wu Z, Zhou Z. Multi-label text categorization using k-Nearest Neighbor approach with M-Similarity. Int. Symp. String Process. Inf. Retr., Springer 2005; 155–60.
[7] Gong J, Teng Z, Teng Q, Zhang H, Du L, Chen S, Bhuiyan MZA, Li J, Liu M, Ma H. Hierarchical graph transformer-based deep learning model for large-scale multi-label text classification. IEEE Access 2020;8:30885–96.
[8] Nam J, Kim J, Loza Mencía E, Gurevych I, Fürnkranz J. Large-Scale Multi-label Text Classification — Revisiting Neural Networks BT - Machine Learning and Knowledge Discovery in Databases. In: Calders T, Esposito F, Hüllermeier E, Meo R, editors., Berlin, Heidelberg: Springer Berlin Heidelberg; 2014; 437–52.
[9] Jayaraman AK, Murugappan A, Trueman TE, Cambria E. Comment toxicity detection via a multichannel convolutional bidirectional gated recurrent unit. Neurocomputing 2021;441:272–8. doi:10.1016/j.neucom.2021.02.023.
[10] Bahdanau D, Cho K, Bengio Y. Neural Machine Translation by Jointly Learning to Align and Translate. ArXiv 2014;1409.
[11] Yadav V, Bethard S. A Survey on Recent Advances in Named Entity Recognition from Deep Learning models. 2019.
[12] Lauriola I, Lavelli A, Aiolli F. An introduction to Deep Learning in Natural Language Processing: Models, techniques, and tools. Neurocomputing 2022;470:443–56. doi:10.1016/j.neucom.2021.05.103.
[13] Conneau A, Schwenk H, Barrault L, Lecun Y. Very deep convolutional networks for text classification. ArXiv Prepr ArXiv160601781 2016.
[14] Kalchbrenner N, Grefenstette E, Blunsom P. A convolutional neural network for modelling sentences. ArXiv Prepr ArXiv14042188 2014.
[15] Kim Y. Convolutional Neural Networks for Sentence Classification 2014.
[16] Zhou C, Sun C, Liu Z, Lau F. A C-LSTM neural network for text classification. ArXiv Prepr ArXiv151108630 2015.
[17] Johnson R, Zhang T. Semi-supervised convolutional neural networks for text categorization via region embedding. Adv Neural Inf Process Syst 2015;28:919.
[18] Rakhlin A. Convolutional neural networks for sentence classification 2016.
[19] Chen Y. Convolutional neural network for sentence classification 2015.
[20] Liu G, Guo J. Bidirectional LSTM with attention mechanism and convolutional layer for text classification. Neurocomputing 2019;337:325–38. doi:10.1016/j.neucom.2019.01.078.
[21] Cao J, Zhang Z, Luo Y, Zhang L, Zhang J, Li Z, Tao F. Wheat yield predictions at a county and field scale with deep learning, machine learning, and google earth engine. Eur J Agron 2021;123:126204. doi:https://doi.org/10.1016/j.eja.2020.126204.
[22] Wulczyn E, Thain N, Dixon L. Ex Machina: Personal Attacks Seen at Scale. Proc. 26th Int. Conf. World Wide Web, Republic and Canton of Geneva, CHE: International World Wide Web Conferences Steering Committee 2017; 1391–1399. doi:10.1145/3038912.3052591.
[23] Guo X. Multi-label Classification and Sentiment Analysis on Textual Records 2019.
[24] Pang Z, Niu F, O'Neill Z. Solar radiation prediction using recurrent neural network and artificial neural network: A case study with comparisons. Renew Energy 2020;156:279–89. doi:https://doi.org/10.1016/j.renene.2020.04.042.
[25] Wang JQ, Du Y, Wang J. LSTM based long-term energy consumption prediction with periodicity. Energy 2020;197:117197.
[26] Srinivasu PN, SivaSai JG, Ijaz MF, Bhoi AK, Kim W, Kang JJ. Classification of Skin Disease Using Deep Learning Neural Networks with MobileNet V2 and LSTM. Sensors 2021;21. doi:10.3390/s21082852.
[27] LeCun Y, Bengio Y, Hinton G. Deep learning. Nature 2015;521:436–44. doi:10.1038/nature14539.
[28] Çetiner H. Classification of Apple Leaf Diseases Using The Proposed Convolution Neural Network

Approach. J Eng Sci Des 2021;9:1130–40. doi:10.21923/jesd.980629.
[29] Langer S. Approximating smooth functions by deep neural networks with sigmoid activation function. J Multivar Anal 2021;182:104696. doi:https://doi.org/10.1016/j.jmva.2020.104696.
[30] Goutte C, Gaussier E. A Probabilistic Interpretation of Precision, Recall and F-Score, with Implication for Evaluation. Lect. Notes Comput. Sci. 2005; 3408:345–59. doi:10.1007/978-3-540-31865-1_25.
[31] Kingma D, Ba J. Adam: A Method for Stochastic Optimization. Int Conf Learn Represent 2014.
[32] Ruder S. An overview of gradient descent optimization algorithms. ArXiv Prepr ArXiv160904747 2016.
[33] Magalhaes A, Small H. Deep Learning Approaches to Classifying Types of Toxicity in Wikipedia Comments.
[34] Mohammed HH, Dogdu E, Görür AK, Choupani R. Multi-Label Classification of Text Documents Using Deep Learning. 2020 IEEE Int. Conf. Big Data (Big Data), IEEE 2020; 4681–9.