Sentiment Analysis with Various Deep Learning Models on Movie Reviews

Muhammet Sinan BASARSLAN
Computer Engineering, Faculty of Engineering and Natural Sciences
Istanbul Medeniyet University
Istanbul, Turkey
e-mail: muhammet.basarslan@medeniyet.edu.tr

Fatih KAYAALP
Computer Engineering, Engineering Faculty
Duzce University
Duzce, Turkey
e-mail: fatihkayaalp@duzce.edu.tr

2022 International Conference on Artificial Intelligence of Things (ICAIoT) | 979-8-3503-9676-8/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICAIoT57170.2022.10121745

***Abstract*—Social media have led to the development of artificial intelligence tasks such as sentiment analysis to see whether people's posts have a positive or negative effect on other people. Ideas that affect society directly or indirectly about various domains, such as a movie or a meal, are very important for many business operations. This paper presents a sentiment analysis study which was carried out with 7 models based on various methods of deep learning algorithms on IMDB dataset. The best result was obtained with the model consisting of 2 Bi-LSTM and 2 dropout layers with 80%-20% train-test separation and an accuracy value of 88.21%.**

Keywords; Deep Learning; Neural Networks; Sentiment Analysis; Text Representation

I. INTRODUCTION

Social media has been a place for people to get ideas. It closely follows people's opinions about itself and its competitors by many companies. Ideas that affect society about areas that directly or indirectly concern many businesses, such as a movie or a meal, are very important for these companies. It is important to process these ideas in terms of presenting new products and gaining financial gain, especially the negatively perceived aspects. Artificial intelligence is used in this. At this point, sentiment analysis has become more important in recent years of dealing with these comments, which are included in text form on social media platforms, with Natural Language Processing (NLP) techniques.

Deep learning is a multi-layered neural network structure that scores significantly better with big data consisting of multiple hidden layers [1]. Sentiment analysis studies are used to extract the emotions in the texts. In this study, sentiment analysis was performed with deep learning algorithms, which have been widely used in recent years. An emotion classification study was conducted on reliable IMDB data collected by Stanford University and made publicly available [2]. Within the scope of the study, 7 models were created on the effect of the number of layers and optimizer functions in deep learning algorithms and the classification process was carried out.

Various machine learning methods have been used in studies on IMDB data. Amulya et. After vectorizing words with methods such as (Term Frequency-Inverse Document Frequency) TF-IDF, Keras embed, sentiment analysis with Logistic Regression (LR), Support Vector Machine (SVM), Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) they have done. RNN performed better with 88% accuracy (ACC) [3]. 87% ACC performance was obtained with RNN after Word2Vec [4]. In the model created with Multi-layer perceptron after BoW, 86.67% ACC performance was obtained [5]. After TF-IDF, models were created with Multi Nominal Naive Bayes (MNB), Bernoulli Naive Bayes (BNB), SVM, Random Forest (RF) and Stochastic Gradient Descent (SGD). Results were obtained with RF 83.66%, BNB 82.91% and MNB 83.16% ACC [6]. 87.90% of ACC results were obtained with Hierarchical ConvNets [7]. As can be seen from these studies, better results were obtained with deep learning models. For this reason, deep learning methods were used in the study.

In the second section of the study, sentiment analysis studies on IMDB data are mentioned. In the third section, methodology, the data set used in the study, data preprocessing techniques and deep learning algorithms and the ACC criteria to be used in the performance of the models created with them are explained. In the fourth section, the experiments carried out within the scope of the study and their results are given. In the last section, the interpretation of the experimental results of the study and future actions are given.

II. METHODOLOGY

The data set used in the study, data preprocessing techniques, and deep learning algorithms and the criteria used to evaluate the performance of the models created with these algorithms are explained.

A. Selecting Text Representation

The representation of documents in word processing is important for successful outcomes. In-text classification applications, texts are represented as vectors in the dataset. It is a vector consistent with the words of the document. Vectorial representation of documents a document-text matrix is created. Consequently, the words in this document are important vectors. It is computed using a variety of word weighting methods. In this study, Keras Embedding was used.

B. Dataset

The IMDB dataset, a popular dataset created by Stanford researchers to be used in sentiment analysis studies, was used in the study [1]. The dataset has an equal distribution of

emotion classes with 25,000 positive and 25,000 negative labels.

C. Data Preprocessing

Text preprocessing is the preparatory process for easier processing of texts. It includes operations such as removing stop words and special characters. NLP is used to prepare the data before the classification step.

Only the most important words are taken or existing words are cleaned from HTML tags, Special characters and numbers are removed to achieve better and more reliable classification results. All texts have been converted to lowercase by removing punctuation marks. Also the stopwords have been removed.

D. GRU

GRU is a type of Recurrent Neural Network. GRUs, which consist of only two gates as shown in Fig.1, a reset gate, and an update gate, use the hidden state to transfer information.

Update Gate: It decides the information to discard and the new information to include.

Reset Gate: This gate is used to decide how much of the past information is to be forgotten.

GRUs are slightly faster than other types of RNNs since they have fewer vector operations.

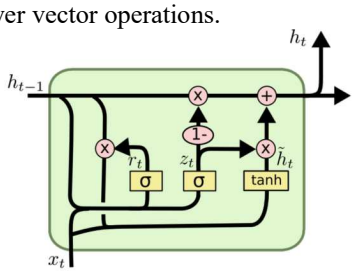

Figure 1. GRU architecture

E. LSTM

RNN architectures have an approach based on prior knowledge usage. LSTM networks are no different from RNN networks [9]. LSTMs were developed by Hochreiter and Schmidhuber in 1997 due to the need to estimate context gaps, known as the disadvantage of RNNs [10]. A structure is used in LSTM networks to calculate hidden states. LSTM contains memory cells. These memory cells are the cells that hold the input information with the previous state. These cells in the network architecture decide which data to keep or which data to delete. In the next step, they combine the previous state with the current memory and the input data. With such an approach, long-term dependencies are eliminated, making it possible to maintain datasets [9]. The LSTM cell contains these three gates:

- The entrance gate; controls the flow of input activations to the memory cell.
- The exit gate; controls the output flow of cell activation.
- The forget gate; filters the information in the input and previous output and decides which to be remembered or forgotten [10]. Besides the three gates, the LSTM cell contains cell update, which is the tenth layer, usually part of the cell state.

In each LSTM cell, three variables enter the cells:

- X_t, current input,
- h_{t-1}, previous output,
- C_{t-1}, previous cell status, on the other hand, two variables come out of the cell:
- h_t, Current output
- C_t, Current cell status

The LSTM structure is given in Fig. 2 [11].

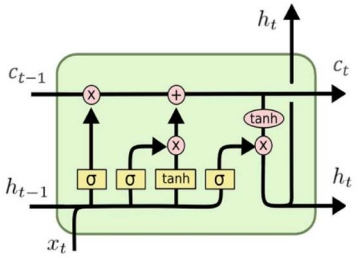

Figure 2. LSTM architecture

F. Bi-GRU

Bidirectional recurrent neural networks bring together only two independent GRUs. This structure ensures that networks always have both backward and forward information about the array in each step and shown in Fig. 3 [12].

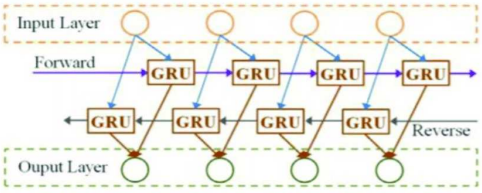

Figure 3. Bi-GRU architecture

G. Bi-LSTM

Bidirectional recurrent neural networks bring together only two independent LSTMs. This structure ensures that networks always have both backward and forward information about the array in each step as shown in Fig. 4 [13].

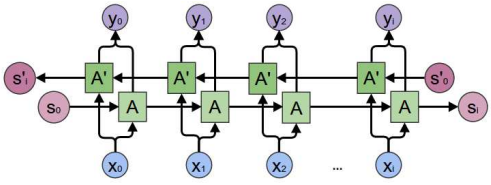

Figure 4. Bi-LSTM architecture

H. Experimental Setup

In the experimental analysis, the two data sets, whose preprocessing process and feature extraction are done, are classified with ensemble and machine learning classifiers

after the holdout (80%-20% and 70%-30%) test training separation, and then their performances are ACC Evaluated. The experiments have been coded by using the python Keras library. Experiments have been implemented on Google Colabaratory (Colab) [14].

I. Performance Metric

To evaluate the predictive performance of different feature sets, classification algorithms classification ACC is utilized as the evaluation metrics. Classification ACC is one of the most widely employed metrics in examining classifiers. It is the proportion of true positives and true negatives over the total number of instances as given by Eq. (1) [15].

$$ACC = (TP+TN) /(TP+TN+FP+FN). \quad (1)$$

III. EXPERIMENTAL RESULTS

The results obtained in the study are given in Tables I-VIII. These tables include the optimizers used, batch sizes, layers, and retrieved ACC values.

As in Table I, model 6, consisting of 2 LSTM and 2 dense layer, gave the best result with 87.26% ACC among the 7 models created with LSTM. While the increase in the number of LSTM neural network layers increased the ACC, the increase in the Dense layer did not positively affect the model performance as the increase in the number of LSTM layers did. Table I shows that the Adam optimizer has a positive effect on performance compared to other optimizer.

TABLE I. LSTM 80%-20% TRAIN-TEST RESULTS

LSTM	Model1	Model 2	Model 3	Model4	Model5	Model6	Model7
Optimizer	Adam	Adadelta	Adadelta	Adam	Adam	Adam	Adam
Activation	Sigmoid	Sigmoid	Sigmoid	Sigmoid	Sigmoid	Relu	Softmax
Batch size	128	128	128	128	128	128	128
Epoch	5	5	5	5	5	5	5
Layers	LSTM Dense	LSTM Dense	LSTM Dense	LSTM LSTM Dense	LSTM Dropout LSTM Dropout Dense	LSTM Dense Dense	LSTM Dense Dense
ACC (%)	87.04	79.05	87.11	87.69	86.66	**87.26**	82.17

As can be seen in Table II, model 4, consisting of 2 LSTM and dense layers, gave the best results with 87.85% ACC in 7 models created with the LSTM. While the increase in the number of LSTM neural network layers increased the ACC, the increase in the Dense layer did not positively affect the model performance as the number of LSTM layers increased. Table II shows that Adam optimizer has a positive effect on performance compared to other optimizer. Using the dropout layer has a positive effect on performance.

TABLE II. LSTM 70%-30% TRAIN-TEST RESULTS

LSTM	Model1	Model 2	Model 3	Model4	Model5	Model6	Model7
Optimizer	Adam	Adadelta	Adadelta	Adam	Adam	Adam	Adam
Activation	Sigmoid	Sigmoid	Sigmoid	Sigmoid	Sigmoid	Relu	Softmax
Batch size	128	128	128	128	128	128	128
Epoch	5	5	5	5	5	5	5
Layers	LSTM Dense	LSTM Dense	LSTM Dense	LSTM LSTM Dense	LSTM Dropout LSTM Dropout Dense	LSTM Dense Dense	LSTM Dense Dense
ACC (%)	87.59	87.06	87.66	**87.85**	87.70	87.81	87.16

As can be seen in Table III, model 4, consisting of 2 GRUs and a dense layer, gave the best results with 88.10% ACC in 7 models created with the GRU. While the increase in the number of GRU neural network layers increased the ACC, the increase in the Dense layer did not positively affect the model performance as the increase in the number of GRU layers did. Adam optimizer, on the other hand, has a positive effect on performance compared to other optimizer.

TABLE III. GRU 80%-20% TRAIN-TEST RESULTS

GRU	Model1	Model 2	Model 3	Model4	Model5	Model6	Model7
Optimizer	Adam	Adadelta	Adadelta	Adam	Adam	Adam	Adam
Activation	Sigmoid	Sigmoid	Sigmoid	Sigmoid	Sigmoid	Relu	Softmax
Batch size	128	128	128	128	128	128	128
Epoch	5	5	5	5	5	5	5
Layers	GRU Dense	GRU Dense	GRU Dense	GRU GRU Dense	GRU Dropout GRU Dropout Dense	GRU Dense Dense	GRU Dense Dense
ACC (%)	86.76	82.71	86.84	**88.10**	87.62	87.21	88.16

As can be seen in Table IV, model 4, consisting of 2 GRUs and a dense layer, gave the best results with 88.13% ACC in 7 models created with the GRU. While the increase in the number of GRU neural network layers increased the ACC, the increase in the Dense layer did not positively affect the model performance as the increase in the number of GRU layers did. Adam optimizer, on the other hand, has a positive effect on performance compared to other optimizer.

TABLE IV. GRU 70%-30% TRAIN-TEST RESULTS

GRU	Model1	Model 2	Model 3	Model4	Model5	Model6	Model7
Optimizer	Adam	Adadelta	Adadelta	Adam	Adam	Adam	Adam
Activation	Sigmoid	Sigmoid	Sigmoid	Sigmoid	Sigmoid	Relu	Softmax
Batch size	128	128	128	128	128	128	128
Epoch	5	5	5	5	5	5	5
Layers	GRU Dense	GRU Dense	GRU Dense	GRU GRU Dense	GRU Dropout GRU Dropout Dense	GRU Dense Dense	GRU Dense Dense
ACC (%)	87.69	81.27	87.85	**88.13**	87.69	88.04	88.26

As can be seen in Table V, model 5, consisting of 2 Bi-LSTM and 2 Dropout, gave the best result with 88.21% ACC in 7 models created with Bi-LSTM. The increase in the number of Bi-LSTM neural network layers and the number of Dropout layers did not positively affect the model performance. In addition, the effect of GRU and LSTM neural networks in two separate train-test separations on Bi-LSTM compared to the use of the Dropout layer was observed. Adam optimizer, on the other hand, has a positive effect on performance in this experiment comparable to other optimizer.

TABLE V. BI- LSTM 80%-20% TRAIN-TEST RESULTS

Bi-LSTM	Model 1	Model 2	Model 3	Model 4	Model 5	Model6	Model7
Optimizer	Adam	Adadelta	Adadelta	Adam	Adam	Adam	Adam
Activation	Sigmoid	Sigmoid	Sigmoid	Sigmoid	Sigmoid	Relu	Softmax
Batch size	128	128	128	128	128	128	128
Epoch	5	5	5	5	5	5	5
Layers	Bi-LSTM Dense	Bi-LSTM Dense	Bi-LSTM Dense	Bi-LSTM Bi-LSTM Dense	Bi-LSTM Dropout Bi-LSTM Dropout Dense	Bi-LSTM Dense Dense	Bi-LSTM Dense Dense
ACC (%)	88.08 %	80.03	87.64	86.72	**88.21**	87.60	87.70

As can be seen in Table VI, model 5, consisting of 2 Bi-LSTM and 2 Dropout, gave the best result with 87.72% ACC in 7 models created with Bi-LSTM. The increase in the number of Bi-LSTM neural network layers and the number of Dropout layers did not positively affect the model performance. In addition, it has been observed that the use of Adam optimizer and 2 Dropouts affects increasing performance, as in the previous separation (Table VI).

TABLE VI. BI- LSTM 70%-30% TRAIN-TEST RESULTS

Bi-LSTM	Model1	Model 2	Model 3	Model4	Model5	Model6	Model7
Optimizer	Adam	Adadelta	Adadelta	Adam	Adam	Adam	Adam
Activation	Sigmoid	Sigmoid	Sigmoid	Sigmoid	Sigmoid	Relu	Softmax
Batch size	128	128	128	128	128	128	128
Epoch	5	5	5	5	5	5	5
Layers	Bi-LSTM Dense	Bi-LSTM Dense	Bi-LSTM Dense	Bi-LSTM Bi-LSTM Dense	Bi-LSTM Dropout Bi-LSTM Dropout Dense	Bi-STM Dense Dense	Bi LSTM Dense Dense
ACC (%)	87.54%	79.45	87.01	87.32	87.72	87.30	86.65

As in Table VII, model 4, consisting of 2 Bi-GRU and Dense, gave the best result with 88.20% ACC in 7 models created with the Bi-GRU. The increase in the number of Bi-GRU neural network layers gave a better model performance than the increase in the number of Dropout or Dense layers. Also, Adam has given the optimizer.

TABLE VII. BI- GRU 80%-20% TRAIN-TEST RESULTS

Bi-GRU	Model1	Model 2	Model 3	Model4	Model5	Model6	Model7
Optimizer	Adam	Adadelta	Adadelta	Adam	Adam	Adam	Adam
Activation	Sigmoid	Sigmoid	Sigmoid	Sigmoid	Sigmoid	Relu	Softmax
Batch size	128	128	128	128	128	128	128
Epoch	5	5	5	5	5	5	5
Layers	Bi-GRU Dense	Bi-GRU Dense	Bi-GRU Dense	Bi-GRU Bi-GRU Dense	Bi-GRU Dropout Bi-GRU Dropout Dense	Bi-GRU Dense Dense	Bi-GRU Dense Dense
ACC (%)	88.15	86.51	87.19	**88.20**	86.63	87.43	86.57

As in Table VIII, model 4 consisting of 2 Bi-GRU and Dense gave the best result with 88.19% ACC in 7 models created with Bi-GRU. The increase in the number of Bi-GRU neural network layers gave a better model performance than the increase in the number of Dropout or Dense layers. Also, Adam has given the optimizer.

TABLE VIII. BI- GRU 70%-30% TRAIN-TEST RESULTS

Bi-GRU	Model 1	Model2	Model3	Model4	Model5	Model6	Model7
Optimizer	Adam	Adadelta	Adadelta	Adam	Adam	Adam	Adam
Activation	Sigmoid	Sigmoid	Sigmoid	Sigmoid	Sigmoid	Relu	Softmax
Batch sizess	128	128	128	128	128	128	128
Epoch	5	5	5	5	5	5	5
Layers	Bi-GRU Dense	Bi-GRU Dense	Bi-GRU Dense	Bi-GRU Bi-GRU Dense	Bi-GRU Dropout Bi-GRU Dropout Dense	Bi-GRU Dense Dense	Bi-GRU Dense Dense
ACC (%)	87.70	80.46	86.24	**88.18**	86.11	87.37	85.91

As seen in Table IX, with an ACC value of 88.21%, better results were obtained than other studies on the IMDB dataset.

TABLE IX. COMPARISON WITH RELATED WORKS

Ref	Text Representation	Model	ACC (%)
[3]	TF-IDF, Count Vectorizer, Keras embed	LR, SVM, MNB, XGBoost, CNN, RNN, LSTM	88 (RNN)
[4]	Word2Vec	RNN	87
[5]	Bow	MLP	86.67
[6]	Unigram with TF-IDF	MNB, BNB, LR, SVM, SGD, RF	83.66 (RF)
[7]	D-dimensional dense vector, n inputs, feature map of d × n in size	Hierarchical Con-vNets	87.90
Presented model	Keras embed	Bi-LSTM (Model 5, in Table V)	**88.21**

IV. CONCLUSION

People's opinions are important for every public-private company that serves and sells people, from work to health. These ideas are classified by deep learning by processing, NLP techniques in sentiment analysis tasks. For this purpose, emotion classification was performed on the widely used IMDB data set with different deep learning methods such as LSTM, GRU and Bi-GRU, Bi-LSTM. 7 models were created to investigate and compare the results obtained in different optimizers, activation functions, batch size and epoch.

The comparison of the Bi-LSTM model (model 5 in Table IV) obtained at 80%-20% train-test separation, which gave the best results in the experiments conducted within the scope of the study, with the previous studies with the IMDB data set is given in Table 8.

In this study, 7 models were created on the effect of the number of layers, and optimizer functions and the classification process was carried out.

If we list our other inferences within the scope of the study below;

- Adam performed well in Activation functions in the same batch size and epoch.
- It has been observed that the increase in the number of GRU or LSTM neural network layers increases the model performance despite the increase in the number of Dense layers.
- Although using Dropout has a positive effect on performance, better results are obtained by using Bi-GRU neural network alone.

In future studies, deep learning models will be created by digitizing the words with methods such as pre-trained embedding and transformer and choosing the most optimal hyper parameter.

REFERENCES

[1] M.S. Bakay, Ü. Ağbulut, Ü. "Electricity production based forecasting of greenhouse gas emissions in Turkey with deep learning, support vector machine and artificial neural network algorithms," *Journal of Cleaner Production*, 285, 125324,2021.

[2] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," ACL-HLT 2011 - Proc. 49th Annu. Meet. Assoc. Comput. Linguist. Hum. Lang. Technol., vol. 1, pp. 142–150, 2011.

[3] K. Amulya, S. B. Swathi, P. Kamakshi, and Y. Bhavani, "Sentiment Analysis on IMDB Movie Reviews using Machine Learning and Deep Learning Algorithms," in 2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT), Jan. 2022, pp. 814–819, doi: 10.1109/ICSSIT53264.2022.9716550.

[4] K. S. prabh and P. N. Karthikayan, "For Movie Reviews, A Sentiment Analysis using Long Short Term Memory Networks," Turkish J. Comput. Math. Educ., vol. 12, no. 9, pp. 1758–1766, 2021.

[5] Z. Shaukat, A. A. Zulfiqar, C. Xiao, M. Azeem, and T. Mahmood, "Sentiment analysis on IMDB using lexicon and neural networks," SN Appl. Sci., vol. 2, no. 2, p. 148, Feb. 2020, doi: 10.1007/s42452-019-1926-x.

[6] M. Mohaiminul and N. Sultana, "Comparative Study on Machine Learning Algorithms for Sentiment Classification," Int. J. Comput. Appl., vol. 182, no. 21, pp. 1–7, 2018, doi: 10.5120/ijca2018917961.

[7] W. Yin, K. Kann, M. Yu, and H. Schütze, "Comparative Study of CNN and RNN for Natural Language Processing," 2017, [Online]. Available: http://arxiv.org/abs/1702.01923.

[8] H. Canli and S. Toklu, “Deep Learning-Based Mobile Application Design for Smart Parking,” IEEE Access, vol. 9, pp. 61171–61183, 2021, doi: 10.1109/ACCESS.2021.3074887.

[9] S. H. Qing, X. Wenjie, and Wenfang, “Robust Support Vector Machine with Bullet Hole Image Classifi- cation,” IEEE Transactions on Systems. Man. and Cybernetics. Part C (Applications and Reviews, Vol. 32, No. 4, 2002, pp. 440–448.

[10] M. S. Basarslan and F. Kayaalp Sentiment Analysis on Social Media Reviews Datasets with Deep Learning Approach. Sakarya University Journal of Computer and Information Sciences, Vol. 4, No. 1, 2021, pp. 35–49.

[11] A. Graves, . Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," Neural networks, 18(5-6), 602-610.

[12] P.Anki, A. Bustamam, and R.A. Buyung, “Comparative Analysis of Performance between Multimodal Implementation of Chatbot Based on News Classification Data Using Categories,” Electronics, vol. 10, no. 21, 2021, pp 2696.

[13] S. Ahmed, A. F. Saif, M.S. Hanif, M. M. N, Shakil, M.M. Jaman, M.M. U, Haque, S.B., Shawkat, J. Hasan, B.S. Sonok, F. Rahman, H. M. Sabbir, Att-BiL-SL: Attention-Based Bi-LSTM and Sequential LSTM for Describing Video in the Textual Formation. Appl. Sci. 2022, 12, 317.

[14] Google, “Colab.” Access: https://colab.research.google.com/.

[15] M. S. Başarslan and F. Kayaalp, "Performance Analysis Of Fuzzy Rough Set-Based And Correlation-Based Attribute Selection Methods On Detection Of Chronic Kidney Disease With Various Classifiers," 2019 Scientific Meeting on Electrical-Electronics & Biomedical Engineering and Computer Science (EBBT), 2019, pp. 1-5, doi: 10.1109/EBBT.2019.8741688.