Computer & Robotics

Optimization and Improvement of Spam Email Detection Using Deep Learning Approaches

Mohsen Nooraee [a], Hamidreza Ghaffari [a,*]

Department of Computer Engineering, Ferdows Branch, Islamic Azad University, Ferdows. Iran

Received 14September 2022; Accepted 14 October 2022

Abstract

Today, one of the widely used fields in artificial intelligence is text mining methods, which due to the expansion of virtual space and the increase in the use of media and social messengers, and on the other hand, the ability of these methods to extract the desired information from a very large volume of Unstructured text files have a special place. for example, one of its applications can be mentioned in spam detection. Nowadays, the presence of spam content in social media is increasing drastically, and therefore spam detection has become critical. Users receive many text messages through social networks. These messages contain malicious links, programs, etc., and it is necessary to identify and control spam texts and emails to improve social media security. There are various techniques for this, among which neural networks have shown more effective results. In this article, an approach based on deep learning using an LSTM neural network and Glove word embedding method is introduced to display text word vectors to detect spam emails. The results of the proposed model have been evaluated using accuracy criteria. This model has shown successful and acceptable performance by achieving 98.39% and 99.49% accuracy on two different data sets.

Keywords: spam emails, LSTM, GloVe, deep learning.

1.Introduction

Artificial intelligence is one of the emerging technologies that has changed the way we look at business issues. More and more companies are turning to advanced analytics and machine learning to solve their problems. With this evolution in the era of artificial intelligence, natural language processing (NLP) provides many opportunities for people and businesses to use the advantages and high power of these methods. Artificial intelligence and natural language processing can help combat massive unstructured data in various fields including healthcare, education, fake news, business sectors, security, and trust, as well as analyzing public opinion and sentiment in various sectors. Natural language processing enhances human-machine interaction more effectively.

Nowadays, the volume of electronic data such as e-books, digital libraries, e-mails, e-newspapers, and e-publications is increasing rapidly, and hence the management of such electronic data is more challenging. Text mining is a method capable of extracting useful information from texts, as well as summarizing, classifying, and retrieving data.

Recently, the widespread use of the Internet and the low cost of sending email, and its high speed and free access have attracted companies and business owners to market in this way. In addition to increasing and attracting attention to this issue, the possibility of abusing its capabilities in the form of spam is also formed. Spam is one or more

*Corresponding Author. Email: hghaffari@ferdowsiau.ac.ir

unsolicited messages in the form of advertisements such as debt reduction, get-rich-quick ways, online dating, health-related products, etc., and advertising companies attract users through fake advertisements and use deceptive pretenses. forces them to click on their links and visit the sites. In addition, these spam messages waste network bandwidth and time, which can cause dissatisfaction to recipients and cause damage to their systems by sending viruses and malware. Therefore, it is very important to recognize these thousands of letters.

Spam detection methods are generally divided into three categories. Simple, smart, and combined methods. A simple method involves listing unknown and suspicious senders, aliases, etc. The intelligent methods include the use of artificial neural networks (ANN), support vector machine (SVM), simple Bayes, etc., And the hybrid methods involve a combination of 2 or more methods to facilitate performance improvement. While machine learning algorithms such as Bag-of-Words, Decision Trees, Naïve Bayes, and N-gram have limitations, deep learning sequence models such as RNN, GRU, and LSTM are emerging with more promising results [1] [2].

In this article, we tried to design a model that can detect spam with high accuracy by using intelligent methods and using artificial neural networks. For this purpose, an LSTM neural network model based on deep learning is proposed. At the beginning and before training the model, the concept of stop words and textual data preprocessing techniques have been used, and a tokenization method has been used to extract features. To embed the words, the concept of Glove is used, which represents each word as a vector. Next, the text passes through the LSTM neural network and various sub-layers in the architecture. Finally, the model is tuned to distinguish between genuine and spam emails.

The next parts of the article are organized as follows. Section 2 summarizes related work. Section 3 describes the data used in this study and the proposed model, the information related to the data set, and the architecture layers required. The results of the proposed model and its performance evaluation are done in section 4. Finally, Section 5 describes the conclusions of the findings and the proposed model.

2. Related Works

The problem of identifying spam emails has attracted the attention of many researchers. In this section, previous related works focusing on spam classification using ML and deep learning techniques are discussed.

In [1], text classification techniques with continuous word methods (CBOW), Naíve Bayes, N-gram, and tree-based methods have been investigated, which also had relative success.

In [2], the idea of transferring the word embedding from the pre-trained corpus for similarity and context extraction is adopted, and in [3], sequence neural models are used to track relevant information, and in this paper, it is shown that sequence models automatically extract text-sensitive features from raw text. In [4], [5], and [6], text classification techniques are used to detect email spam and sentiment analysis. In [7], Gradient Descent optimization is investigated to find global minima to update new weights and biases in each iteration. In [8], a supervised machine learning-based solution using SVM is proposed for effective spam detection.

In [9], some of the limitations of spam detection methods such as blacklisting are mentioned and a new technique based on deep learning technique is proposed to deal with the above challenges. The syntax of each tweet is taught through WordVector and deep learning. A binary classifier is then used to distinguish between spam and normal tweets.

In [10], an ensemble approach for spam detection at the tweet level is proposed and various deep learning models based on Convolutional Neural Networks (CNN) are developed. Five CNNs and one feature-based model are used in this set. Each CNN uses different word embeddings (Glove, Word2vec) to train the model. The feature-based model uses content-based, user-based, and N-gram features. This paper combines both deep learning models and

traditional feature-based methods using a multi-layer neural network that acts as a meta-classifier.

In [11], a new deep learning architecture based on a convolutional neural network (CNN) and short-term neural network (LSTM) is proposed. This model is supported by introducing semantic information in word representation with the help of knowledge bases such as WordNet and ConceptNet. Using these knowledge bases improves performance by providing a better semantic vector representation of test words that previously had a random value due to not having been seen in training.

In [12], a spam detector is introduced using a pre-trained BERT model that classifies emails and messages by understanding their context, and the spam detector model is implemented using multiple sets such as SMS set, Enron set, SpamAssassin, the body of Ling-Spam has been trained. In [13], an efficient spam detection model using a pre-trained two-way encoder Representation of Transformer (BERT) and machine learning algorithms is proposed to classify concurrent or spam emails. The email texts were fed to BERT and the features obtained from the BERT output were used to display the texts.

3. Description of Model

In this paper, we aim to train our model to detect spam emails. First, we intend to build a model that can help people in the community. Secondly, our focus is to minimize the spread of spam through early detection in order to prevent its further spread and its subsequent consequences, such as the spread of viruses and malware, or involving bandwidth and wasting network traffic, etc.

3.1. Description of the Dataset

To distinguish between real and spam emails, data is collected from two different sources(Table1). Data set of real and spam emails taken from kaggle.com website [14]. The data set has two columns of text and label, which contains about 5570 emails, and among them, both real emails (0) and spam (1) can be seen. The second dataset is the open-source spam dataset from Kaggle [15], which contains 5726 emails containing 1368 spam emails.

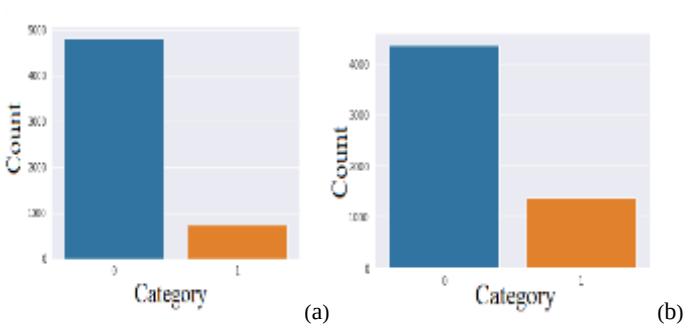
Table 1
Specifications of the data set.

Data Set	Real email	spam	Total samples
Shalini Gupta	4825	747	5572
Karthickveerakumar	4358	1368	5726

3.2. Visualization and Pre-Processing of Data

The dataset is divided into two classes, one as genuine emails and the other as spam. Data visualization helps us better understand the meaning of relative data by displaying the data in a visual environment, such as maps or charts. This makes it easier for the human mind to spot trends, patterns, and outliers in large data sets by making the data more natural for analysis. The dataset is classified into two categories: spam and real emails. The first category is the category of spam emails, which is indicated by the class "1" and the second category is the category of real emails, which is indicated by the class "0".

Data preprocessing is an important step that involves manipulating data before execution to increase efficiency. This pre-processing includes data cleaning and data transformation, which we will discuss further. To start with the data preprocessing, Figure 1 shows the number of articles according to the corresponding class labels, respectively, spam and real.



(a) (b)

Fig.1. Email dataset according to spam and genuine categories. a) Shalini Gupta. b) Karthickveerakumar

3.2.1. Words Stemming

The word formation process consists of reducing different forms of words to general forms. Rooting refers to the extraction of word roots or root forms that may fully reflect semantic thinking.

3.2.2. Removing stop words

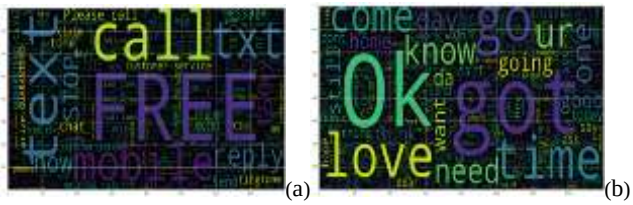
Stopwords are English words that do not add much to the meaning of the sentence. They can be easily omitted without changing the meaning of the sentence. Examples of stop words include the, he, and have. Such words have already been collected and stored in a package called corpus, which can be seen in the NLTK directory. It can be installed in the Python environment itself.

3.2.3. Removing Symbols and Special Characters

In the continuation of the cleaning and pre-processing process on the data, symbols and special characters such as (* /, ×? > " : and ...) can be removed from the desired text.

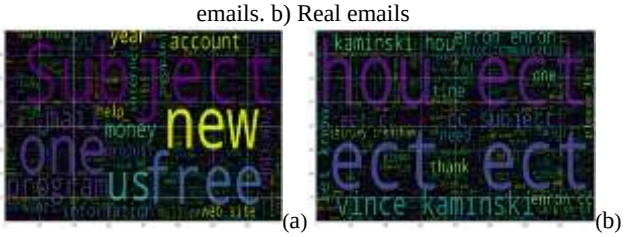
3.3. Data Processing

Figures 2 and 3 show the keywords in the set of real emails and spam in two different datasets, respectively. A maximum of 2000 words is considered to create a word cloud for each category. Such words are known as word clouds, which depict groups of words displayed and highlighted in multiple sizes and lengths. In Figures 2 and 3, longer and larger words are observed such as free, call, etc., which means that these words contribute more to the spam email dataset. The bigger and bolder this phrase is, the more it appears in a document and the more important it is. The words from the input are considered in two separate categories, which include spam emails and the other category contains real emails. And according to each category, the word cloud is generated.



(a) (b)

Figure 2 .Word cloud for Shalini Gupta dataset. a) Spam emails. b) Real emails



(a) (b)

Fig.3.Word cloud for karthickveerakumar dataset. a) Spam emails. b) Real emails.

Also, in Figures 4 and 5, what is the average length of words in real and spam emails on two different data sets.

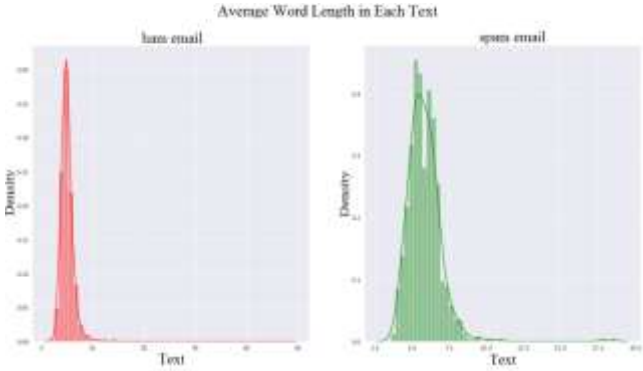


Fig. 4.Average word length on Shalini Gupta dataset.

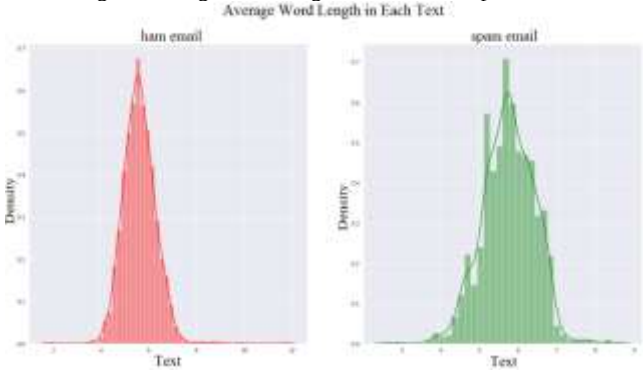


Fig.5. Average word length on Karthickveerakumar dataset.

3.4. Text tokenization and feature extraction

The process of representing each word in the form of a number or vector is called tokenization. To use textual data for predictive modeling, the text is parsed to remove specific words known as the

tokenization process. Then, feature (or vector) extraction is performed where the words are converted to integers or floating point numbers to be used as input in machine learning methods.
Machine learning techniques are widely used in the field of text analysis. However, since most algorithms require fixed-size numeric feature vectors rather than variable-length raw text documents, the raw data, a sequence of symbols, cannot be provided directly. The most common methods for extracting numerical features from text content are:

- By using white spaces and punctuation marks as token separators, text can be tokenized and each potential token assigned an integer identifier,
- Occurrence of a token in each document can be counted,
- Tokens that appear in most samples or documents are normalized and weighted by decreasing token importance.

3.5. Words Embedding: (GLOVE)

The purpose of Word Embedding is to convert the text into a form that can be understood by the computer, and since computer systems work with numbers, it is necessary to convert the input data into numbers by using word embedding methods. There are two main methods for teaching Word Embedding models:

A. Distributed Semantic Models: These models are based on the co-occurrence/proximity of words in a large text collection. A co-occurrence matrix (co-occurrence matrix) is formed for a large text collection (a matrix with values that each represent the probability of words occurring next to each other), and this matrix is decomposed to form a word vector matrix. Word Embedding modeling techniques in this way are known as Count Based methods.

B. Neural Network Models: Methods based on neural networks are generally prediction-based methods in which models are used to predict content words using middle words, or vice versa (prediction of the middle word using a set of of content words) are made.

Prediction-based methods generally outperform count-based methods from the point of view of efficiency and perform better than them. Some of the most common Word Embedding algorithms, such as word2vec, Glove, fast text, etc., are all predictive methods. The problem with neural network-based models is the high complexity of these types of networks. This complexity in neural network models (both feedforward and recurrent models) originates from nonlinear hidden layers. In this article, we have used the Glove word embedding algorithm.

The GloVe is an unsupervised learning technique that generates word vector representations. This method has the advantage that, unlike word2vec, it does not only rely on local statistics to generate word vectors but also includes global statistics. If you want to get more information about how Glove works, refer to [16].

3.6. Neural Network: Deep Learning LSTM Model

In this research, we have used neural networks to build the model. The model needs to know what input shape to predict. As a result, in a sequential model, the first layer must receive the input shape information. Classification is a predictive modeling problem where the goal is to predict a category given a sequence of inputs spanning space or time. The fact that sequences may have different lengths, contain large vocabularies of input symbols and may require the model to learn long-term context or associations between symbols in the input sequence makes this challenge difficult. Therefore, a short-term memory model has been used to detect spam emails.

A recurrent Neural Network (RNN) is a type of artificial neural network that is used in speech recognition, natural language processing (NLP), and also in the sequential data processing. Many deep networks like CNN are feed-forward networks, that is, the signal in these networks moves in only one

direction from the input layer to the hidden layers and then to the output layer, and the previous data is not stored in the memory. become But Recurrent Neural Networks (RNN) have a feedback layer where the output of the network along with the next input is fed back to the network. Due to its internal memory, RNN can remember its previous input and use this memory to process a sequence of inputs. In simple words, recurrent neural networks include a feedback loop that makes the information we have obtained from previous moments not be lost and remain in the network. The problem with this type of neural network is that it faces problems when the text is long. In these cases, a special type of artificial network called LSTM should be used.

LSTM network is a type of recurrent neural network that uses LSTM cell blocks instead of traditional neural network layers. The inpu gate, forget gate, and output gate are three components of these cells. Figure 6 shows the LSTM network architecture.

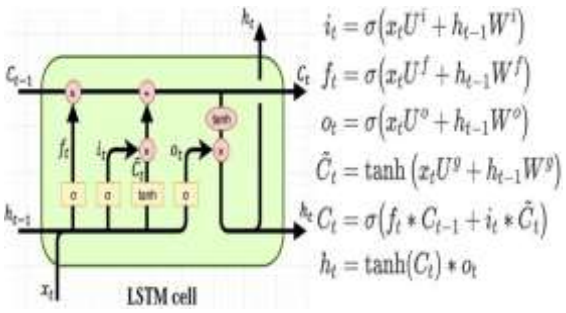


Fig.6. LSTM internal architecture

In this research, we have used the cumulative LSTM model to improve the system. The return sequence is set to True while using the stacked version of LSTM. When the recursive sequence is set to True, the hidden state output of each neuron is used as input to the following LSTM layer. A complex LSTM model with multiple layers of LSTM and Dense is required to classify a particular email into genuine email or spam.

- The first layer is the Embedding layer, which represents each word with 32 length vectors,
- The next two layers are the LSTM layer, which has 256 and 128 memory units, respectively,
- Batchnormalization layers that normalize the output of the previous layer and give it as input to the next layer,
- There are two layers of dense output. The first dense layer contains 64 memory units and the ReLu activation function,
- The next dense layer is the output layer, which consists of a single neuron and a sigmoid activation function.

In a neural network, a dense layer is an ordered layer of neurons. Each neuron in the previous layer receives information from all the neurons in the layer above it and they connect it tightly. A weight matrix W, a bias vector b, and the activation of the previous layer constitute this layer. In this presented model, to avoid overfitting, we have used two normalization layers of batch and bypass in our proposed network.

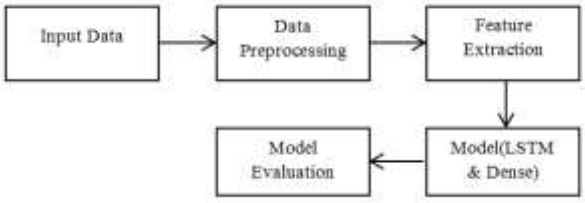


Fig.7.model architecture.

This model is trained for 30 periods. Also, the binary cross-entropy loss function is implemented and the Adam optimizer is used to upgrade the weights. A learning rate of 0.01 is chosen. The batch size is set to 256. while the embedding size is 200. The training of the proposed model has been implemented using colab.google website and on GPU.

4. Results

The training data was randomly divided into two training and validation sets by the train_test_split function. This model is now trained to recognize real emails and spam. The parameters used in this model are in accordance with table number 2. The proposed model is evaluated based on accuracy, precision, recall, F1 score and support criteria. In addition, a comparative analysis has been performed according

to Table 3. In this table, a comparative analysis is shown along with the accuracy of the model.

Table 2
Meta-parameters for the proposed model for the dataset Shalini Gupta:

Dataset	Shalini Gupta	Karthickveerakumar
Hyperparameters	Value	Value
Embedding Layer	1	1
LSTM Layer	2*(256-128)	2*(256-128)
Dense Layer	2*(64-1)	2*(64-1)
Loss Function	Binary Cross Entropy	Binary Cross Entropy
Activation Function	Relu	Relu
Optimizer	Adam	Adam
Learning Rate	0.01	0.01
Number of Epochs	30	30
Embedding Size	200	200
Batch Size	256	256

Table 3
Comparative analysis of results.

Dataset	Accuracy on Training Data	Accuracy on Testing Data		Precision	Recall	F1-score
Shalini Gupta	99.8563	99.4256	ham	0.99	1.00	1.00
			spam	1.00	0.96	0.98
Karthickveerakumar	99.9068	98.3938	ham	0.99	0.99	0.99
			spam	0.96	0.98	0.97

Testing and training is a very necessary part of building any model. The complete basis of the model depends on how well we train the model. Figure 7 shows the details of test and training accuracies obtained from the number of epochs. It can be seen in Figure 8(a) that since the model is trained with 30 cycles, the accuracy of training and testing is growing and improving until the 15th iteration and does not grow further, and in other words, the accuracy The model does not improve and continues with a constant trend. It can also be seen in Figure 8 (b) that after about 10 training and testing periods, the error does not decrease, and this means that after 10 periods, the minimum error is met.

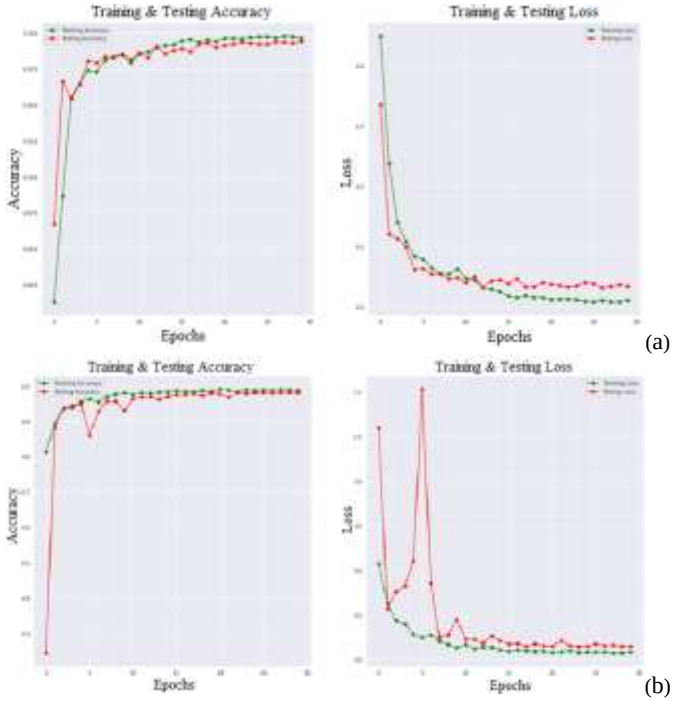


Fig.8. Showing the accuracy and error of testing and training against the number of courses.
a) Shalini Gupta's data set. b) Karthickveerakumar's data set

Figure 9 shows the confusion matrix obtained for the proposed model on two data sets.

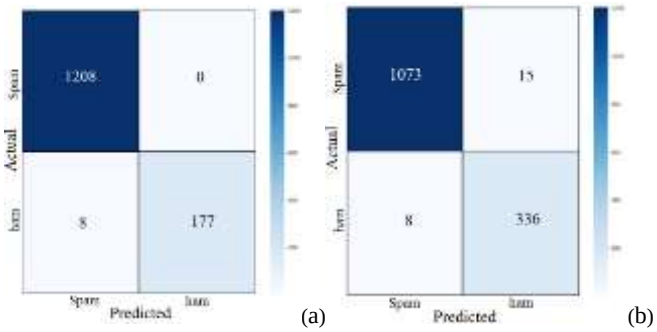


Fig. 9. Confusion matrix. a) Shalini Gupta dataset.
b) Karthickveerakumar dataset.

Figure 10 shows the comparison of the results of the work done on the Karthickveerakumar dataset by different methods.

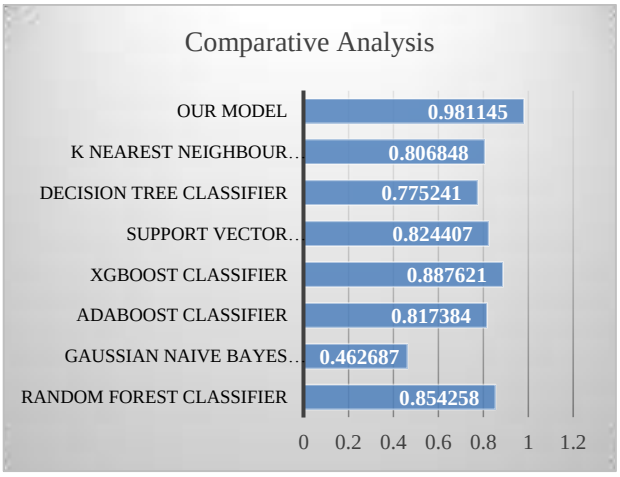


Fig.10.Comparison of accuracy analysis of different models.

Figure11 shows the effect of using word embedding techniques in the proposed model.

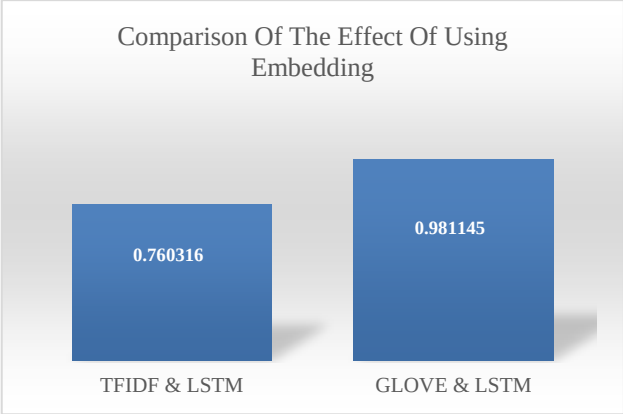


Fig.11.Comparison of the effect of using word embedding

5. Conclusion and Future Work

In this article, by using the GLOVE word embedding technique and LSTM and Dense layers, a deep neural network model was proposed for the task of detecting real and spam emails, and the results of the work were tested and compared with several well-known and widely used machine learning models. The results show that our proposed model has the best results with 99.90% accuracy on training data and 98.39% on test data on Karthickveerakumar dataset and 99.85% accuracy on training data and 99.42% accuracy on test data for Shalini Gupta dataset. In this work, we tested the proposed model on two different datasets as described in Table 1 to prove the strength of the model and the high accuracy of the obtained results. we have achieved

As a suggestion for further work, the presented model can be trained using a larger data set and the accuracy of the model can be improved. Also, the task of detecting spam can be applied to other languages such as Chinese, Farsi, etc.

References

[1] U. Ugurlu, I. Oksuz and O. Tas, "Electricity price forecasting using recurrent neural networks", Energies, vol. 11, no. 5, pp. 1-23, 2018.

[2] A. Malte and P. Ratadiya, Evolution of transfer learning in natural language processing, Oct. 2019.

[3] I. Sutskever, O. Vinyals and Q. V. Le, Sequence to Sequence Learning with Neural Networks, pp. 1-9, 2014.

[4] S. Rayana and L. Akoglu, "Collective Opinion Spam Detection", Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '15, pp. 985-994, 2015.

[5] Y. Ren and D. Ji, "Neural networks for deceptive opinion spam detection: An empirical study", Inf. Sci. (Ny)., vol. 385–386, pp. 213-224, Apr. 2017.

[6] K. Kowsari, K. J. Meimandi, M. Heidarysafa, S. Mendu, L. Barnes and D. Brown, "Text classification algorithms: A survey", Inf., vol. 10, no. 4, pp. 1-68, 2019.

[7] S. S. Du, J. D. Lee, H. Li, L. Wang and X. Zhai, "Gradient Descent Finds Global Minima of Deep Neural Networks", undefined, 2018.

[8] Zheng X et al (2015) Detecting spammers on social networks. Neurocomputing 159:27–34.

[9] Wu T et al (2017) Detecting spamming activities in twitter based on deep-learning technique. Concurr Comput Pract Exp 29(19):e4209.

[10] Madisetty S, Desarkar MS (2018) A neural network-based ensemble approach for spam detection in Twitter. IEEE Trans Comput Social Syst 5(4):973–984.

[11] Jain G, Sharma M, Agarwal B (2019) Spam detection in social media using convolutional and long short term memory neural network. Ann Math Artif Intell 85(1):21–44.

[12] Thaer Sahmoud, Dr. Mohammad Mikki (2022) Spam Detection Using BERT. https://doi.org/10.48550/arXiv.2206.02443.

[13] Guo, Y., Mustafaoglu, Z. ., & Koundal, D. . (2022). Spam Detection Using Bidirectional Transformers and Machine Learning Classifier Algorithms. Journal of Computational and Cognitive Engineering. https://doi.org/10.47852/bonviewJCCE2202192.

[14] Shalini Gupta. [Online]. Available : https://www.kaggle.com/datasets/shalini2810/input-file.

[15] Karthick veerakumar,Spam filter, 2017. [Online]. Available: https://www.kaggle.com/karthickveerakumar/spam-filter.

[16] Pennington, J. , Socher, R. , & Manning, C. D. (2014). Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP) (pp. 1532–1543) .