IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 12, Issue 5, May 2024

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.379

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205246 |

Spam Email Classification using BPNN

Hamza Abdulnabi, Dr. Olusolade Aribake Fadare , Prof. Dr. Fadi Al-Turjman

Master Student, Department of Electrical and Electronic Engineering, Near East University, Faculty of Electrical and Electronics Engineering, Nicosia, Mersin, Turkey

Department of Artificial Intelligence, Software, and Information Systems Engineering, AI and Robotics Institute, Near East University, Faculty of Engineering, Nicosia, Mersin, Turkey

Department of Artificial Intelligence, Software, and Information Systems Engineering, AI and Robotics Institute, Near East University, Faculty of Engineering, Nicosia, Mersin, Turkey

ABSTRACT: The continued expansion of digital communication came with great benefits related to effective communications and comfortable lifestyle, however, this comes with an increased spam and phishing emails that invaded our inboxes in many shapes and thoughts in a try to scam the receivers. This paper explores how it is possible to deploy the backpropagation neural networks (BPNN) for spam email detection. The paper proposes a spam-email classification technique using back propagation neural networks and tree forest. The classification technique application was coded with MATLAB and used relevant functions to train the BPNN on spam detection using a data set of non-spam and spam emails. The achieved accuracy of the BPNN after training reached 97% accuracy. Another code was built to recall the saved BPNN and used it to classify if an input text is spam or not

KEYWORDS: Artificial intelligence, Artificial Neural Network, BPNN, Spam classification

I. INTRODUCTION

The widespread use of email communication in today's digital world has fundamentally transformed the manner in which individuals and organizations engage with one another. Nevertheless, despite the convenience and effectiveness provided by email, there remains a significant and ongoing threat posed by spam emails. Spam, which refers to unwanted and frequently harmful messages, floods email inboxes at a concerning speed, presenting substantial obstacles to users' efficiency, confidentiality, and online security. Reports indicate that spam emails make up a significant majority of all emails sent worldwide, emphasizing the immediate requirement for efficient detection and mitigation measures.

By 2023, over 45.6 percent of global emails were classified as spam, a decrease from the previous year's figure of nearly 49 percent. Since 2011, the proportion of spam emails in the overall email traffic has substantially declined, however, they still constitute a considerable portion, the peak of spam e-mails occurred in May 2023, accounting for about 50 percent of global e-mail traffic (Petrosyan, 2024).

Spammers usually collect the targeted emails datasets using various ways but the most common ones are the websites and computer viruses. all of us has received a spam email or message at some point, the majority of spam emails are mainly advertisement and invitations to websites visits, however, it can be really annoying and time consuming to internet users. Additionally, some spam emails might have viruses or malware included which can impose a harmful impact on the devices including their storage capacity, CPU, and other damages. According to (Yu et al., 2008) spam can have an adverse impact on network performance and speed as it consumes a portion of the network bandwidth in addition to its being time-consuming for users to read, identify a spam email, and delete it.

Back Propagation Neural Network (BPNN) can be effectively used to address this issue by using it to classify and detecting spam emails given that BPNN is a specific kind of the artificial neural network that employs the backpropagation technique to train the network by modifying the weights of connections between neurons. Following BPNN approach to identify and classify emails includes certain key steps including data processing, which converts the text into a numerical form that is suitable for BPNN input, the BPNN is trained based on the supervised data provided to generate a trained BPNN used for email classification purpose. BPNN or back propagation neural network is a widely employed neural network structure given its ability to effectively simulate complex exponential issues, making

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205246 |

it very adaptable to a diverse variety of applications, BPNN is well-suited for tasks such as classification, function approximation, prediction, and more (Wu & Tsai, 2009).

II.REVIEWS

Multiple methodologies exist for designing machine learning (ML) algorithms where the objective of ML algorithms is to utilize observations as input, which can encompass data, patterns, and past experiences, machine learning techniques are utilized to enhance the performance of instances, this is achieved by employing any classifier to categorize input patterns into a set of categories or cluster unknown instances (Kaur & Kumar, 2015). Every machine learning algorithm possesses its unique set of challenges, neural networks, for instance, are plagued by the issues of sluggish training speed and susceptibility to getting stuck in local minima. The classic method for representing features, known as the vector space model (VSM), has three limitations: sparse representation, large dimensionality, and semantic concerns (Li & Huang, 2012).

Email classification can be considered a specific instance of text classification, however, the sparsity and noise in the feature space make it more challenging due to the presence of informal language and a higher frequency of spelling errors in email content (Yu & Zhu, 2009).

In their paper Tuteja and Bogiri (2016) BPNN was employed for the purpose of spam email filtering where 200 emails were used as the basic dataset and were processed using the K-Means Clustering Algorithm achieving an accuracy of 95.42% however the proposed approach had limitations in terms of the lengthy duration required for the training and testing the networks performance. In their study, (Kumar et al., 2012) investigated various classifiers for the purpose of distinguishing between ham and spam data. The researchers focused on the spam-based UCI dataset and used various feature selection filtering algorithms, a maximum accuracy of 99% was achieved when deploying a random tree algorithm with Fisher filtering and run time filtering

Gomez and Moens (2012) suggested using two classifiers to classify (ham) and (spam) emails, the classifiers are derived from principle component analysis document reconstruction (PCADR), that are based on the power factorization method and SVM. The testing process was performed using the public e-mail corpora, that contains of 1,250 real messages (ham) and 1,250 unsolicited messages (spam), the results showed that PCADR outperformed SVM with an accuracy of 93.67% when following a 10-fold cross validation.

III. INTRODUCTION TO ARTIFICIAL NEURAL NETWORK (ANN)

Artificial neural networks are artificial intelligence (machine learning) techniques that replicate the cooperative behavior of biological neurons to identify patterns, evaluate probabilities, and draw conclusions. This enables the NN to make assessments in a manner that mimics the human brain. Neurons are the fundamental cellular units that compose the brain. The human brain structure consists of around 1011 interlinked neurons and 10,000 connections between them. ANN is a real duplication of this biological structure that is performed by connecting artificial neurons in a way similar to the brain neural network (Kukreja et al., 2016).

A neural network consists of an input layer, hidden layers, and a final layer of output, which are composed of nodes or artificial neurons. Each node is assigned a weight and threshold and is interconnected with other nodes. If the output of a node surpasses a specified threshold value, it becomes activated and conveys data to the following network layer. If not, no data is transmitted to the succeeding tier of the network. The below figure shows the similarity between a biological neuron and an artificial neuron:

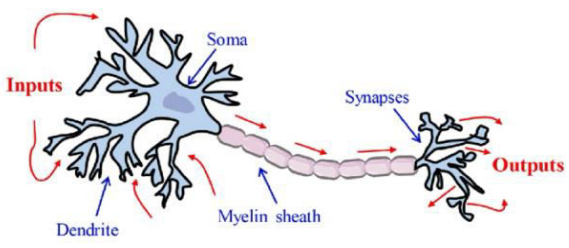

Figure 1 Biological neuron. Source (Kriesel, 2007).

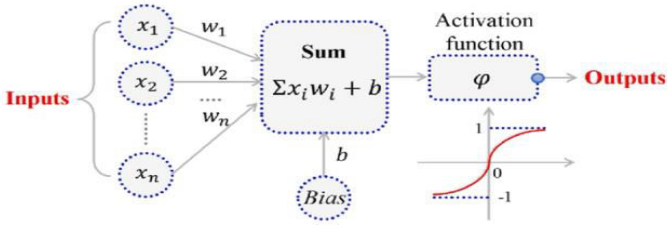

Figure 2 Artificial neuron. Source (Haykin, 1998).

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205246 |

These weights are used to ascertain the value of each variable, with higher weights having a greater impact on the output in comparison to other inputs. Subsequently, each input is multiplied by the weight associated with it and then aggregated. Subsequently, the resulting output is fed into an activation function, which then decides the final output. If the value of that output surpasses a specified threshold, it triggers (or initiates) the node, transmitting data to the subsequent layer in the network. This leads to the transfer of the output from one node to the input of the next node. The transmission of data from one layer to the subsequent layer characterizes this neural network as a feedforward network. The formula can be mathematically presented as follows:

$\sum wixi + bias = w1x1 + w2x2 + w3x3 + bias$
output = f(x) = 1 if $\sum w1x1 + b >= 0$; 0 if $\sum w1x1 + b < 0$

Backpropagation Neural Network (BPNN):
Backpropagation is the basic method used to train neural networks. Weight adjustment in a neural network is achieved by the process of refining the netwrok parameters including its weights based on the error rate calculation achieved in the previous epoch where optimizing the weights enables you to minimize error rates and enhance the model's reliability by improving its ability to generalize, to put it differently, backpropagation is a supervised learning technique specifically designed for training artificial neural networks (ANNs) where backpropagation refers to the process of propagating error gradients in reverse through a network. This step is conducted ti ensure an update and fine-tuning to the network's weights which will eventually reduce the loss function (Johnson, 2024).

When it comes to a back propagation neural network (BPNN), it can be considered as a specific structure of a wider range of artificial neural networks (ANNs), the BPNN uses a specific mechanism to improve the learning and training processes, and this approach is conducted through continuously updating and adjusting the NN parameters including the weights and biases of the BPNN by predicting the variation error rates during the estimation and training process. This method can significantly improve the efficiency of the neural network learning process.

How the backpropagation learning algorithm works:
Back-propagation approach is a highly effective method used for training ANN especially when using a supervised learning scenario. The network has the capability to learn by efficiently adjusting its internal parameters (weights and biases) to minimize the difference between its predictions and the desired outputs. The process of backpropagation can be outlined by the following phases:

• **Forward propagation**: consists of the introducing the input data into the NN. Afterward, calculations are then performed subsequently through every single layer until an outcome is generated. Each individual neuron carries out a calculation by calculating a weighted total of its inputs, including a bias factor, and subsequently applying the activation function to obtain the output. The output of every single layer serves as the starting point for the following layer in the network. The weight value and offsetting value of the network remain unchanged, and the state of each layer of neurons only impacts the subsequent layer of neurons. If the anticipated output cannot be achieved by the output layer, the system can be switched to take advantage of the backpropagation of error signals (Li, 2012).

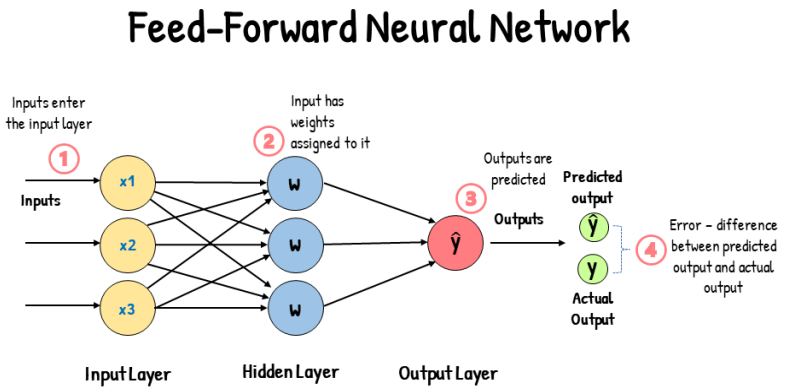

Figure 3 Feed-Forward Neural Network. Source (Kalirane, 2023).

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205246 |

•**Error calculation**: After performing the forward pass, the discrepancy or deviation between the predicted output and the true result is computed using a loss function. The commonly employed loss functions in learning algorithms include the mean squared error (MSE) for analysis of regression and cross-entropy loss for classification tasks, the choice of the loss function depends on the specific characteristics of the current circumstance (Bishop, 2006)

• **Backward propagation**: During the backward pass, the chain rule of calculus is used to calculate the gradient of the loss function concerning each parameter (weights and biases) in the network. This process is known as backward propagation. The process is initiated by computing the loss function gradient compared to the activations of the output layer. Subsequently, it propagates this gradient in the reverse direction across the network, computing the gradients of the loss function with respect to the activations of each hidden layer and the parameters (weights and biases) of every layer. The chain rule in Calculus is employed to calculate the impact of each weight and bias in the network on the total error (Rumelhart et al., 1986).

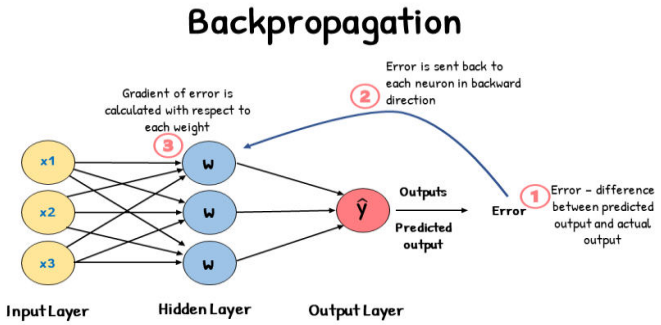

Figure 4 Backpropagation neural network. Source (Kalirane, 2023).

• **Parameters (weights and biases) update**: the neural network's characteristics (weights and biases) are modified by heading in the reverse directions of the computed gradients to reduce the loss function. The learning rate hyperparameter controls the size of the parameter modification throughout updating. Before the reduction from the parameters, the gradients are scaled by the learning rate. This process is carried out continuously for a pre-determined number of epochs or until the loss exceeds an acceptable threshold (Bottou, 2010).

How BPNN can be used to classify emails

Backpropagation Neural Network (BPNN) to classify emails and indicate whether if they are spam or non-spam. The process starts with having a clear datasets that can be used as the fundamental base of the BPNN and to guide it throughout the training process. The dataset should should indicate if the text is spam or non-spam to allow the BPNN to link the features of the emails with its status and learn from it, it is necessary to split the data sets into training and testing data to ensure that the BPNN can identify error rate between the expected and actual output based on both the training and test data. Another curtail process is to extract email features which means converting them to a numerical representation that can be further processed by the network using methods like bag-of-words or TF-IDF

Now after the network identified the features of each email and its corresponding labels (spam or Ham), the BPNN structure starts to set up its input neurons, and hidden layers to receive the inputs and compute outputs relying on the initial weights and biases. The BPNN now starts to update its parameters based on the error rate between the expected and actual outputs through the backpropagation path. the process continues to adjust and fine-tune the BPNN parameters (weights) to minimize the error between actual and expected output. This process might continue for multiple epochs until a desired output is achived. To accurately assess the classification performance of the model, we construct many models using various divisions of the instances into testing and training datasets. The classification error is subsequently computed as the average across all models. The technique is referred to as n-fold cross-validation, where "n" is the number of repeats the dataset is divided into partitions (Ayodele & Khusainov, 2010).

Creating a MATLAB code for spam classification:

An email spam classification code was created using MATLAB to illustrate and explain the training process of BPNN. A prepared dataset from UCI Machine Learning Repository was used for this purpose, the dataset consists of two columns where the first column represents the text and the second column labels each text whether it is spam (1) or

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205246 |

non-spam (0). The total number of texts available in the dataset was 5572 texts. The following steps explain how the code was built:

• Loading and understanding the dataset:
this is the first where the dataset is loaded and its two columns have been named 'Text' and 'Label' for further processing, missing values and duplicates should be check and addressed to minimize the error and run time.

• Data processing:
In this step, the text column needs to be processed by lowercasing the letters and removing special characters, punctuations, and digits to ensure smooth processing. Additionally, a tokenization process is performed to break the text down into individual words to ease further analysis, also removing stop words to reduce any noise and ensuring that only meaningful words are being considered.

• Features extraction:
This process involves converting raw text data into a numerical feature that can be used for further analysis, the bag-of-word approach was used in this code which represents text data to a matrix of word frequencies in the dataset. The Term Frequency-Inverse Document Frequency (TF-IDF) is calculated manually. This is a technique to quantify the importance of a word in a document relative to a collection of documents.

• Data splitting: The data is divided into testing and training sets using the holdout method with an 80-20 split.

• Model building, Back Propagation Neural Network: in this step, a BPNN is being created where a neural network model facilitates the categorization of textual material. Backpropagation is a widely used training technique for neural networks, and adjusting its parameters, such as the number of epochs and learning rate, has an impact on the learning process of the model. K-fold cross-validation enhances the resilience of the model by training and testing it on distinct subsets of data. In this code, the hidden layers' size was set to 100 similar to the number of epochs, where each epoch represents one complete pass of the entire training dataset through the neural network. Additionally, the training rate was set to 0.01.

The code ensured that there is a cross-validation with 5 folds which means that the BPNN will be trained and evaluated five times which will help providing more reliable outcome and accuracy as the system will be trained for multiple times using multiple diverse data combinations. After the final (fifth) fold of cross-validation, the model will be saved to be recalled and used in another code for spam classification. Through each fold, we can see that the accuracy increases from around 87% until it reaches nearly 97% in the last fold. Another step of model building is to build a random forest approach which can be effectively used for classification purposes.
• Model and training visualization: the last step is to plot the fold wise of accuracies, training performance, and learning curve.

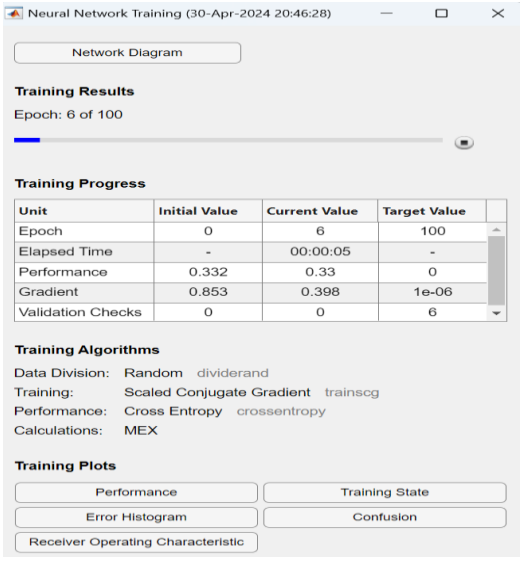

Figure 5 shows how the BPNN is being trained

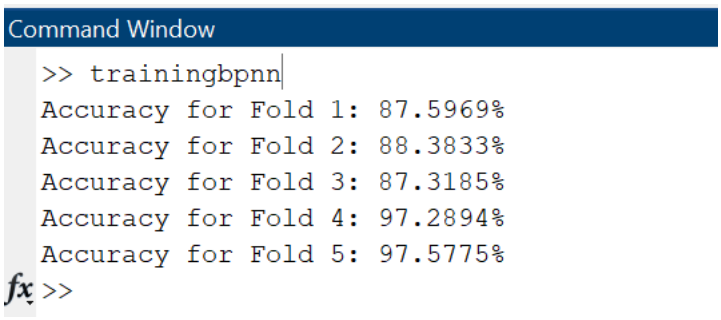

Figure 6 shows the output results of accuracies per each fold

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205246 |

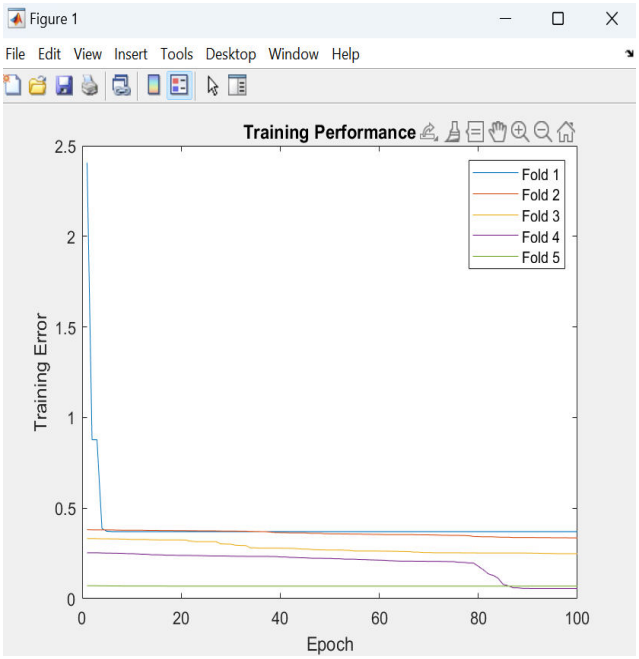

Figure 7 shows the training performance through training error compared to epochs per each fold

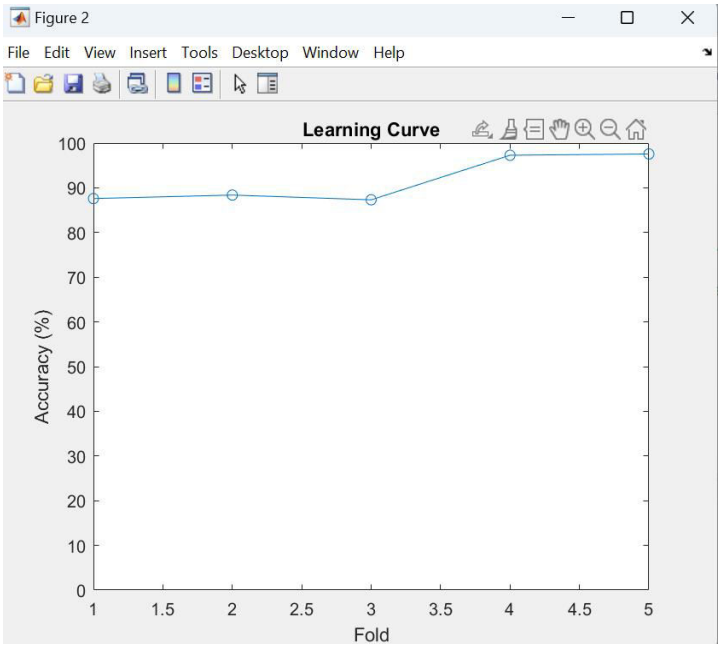

Figure 8 shows the learning curve through the accuracy per each fold

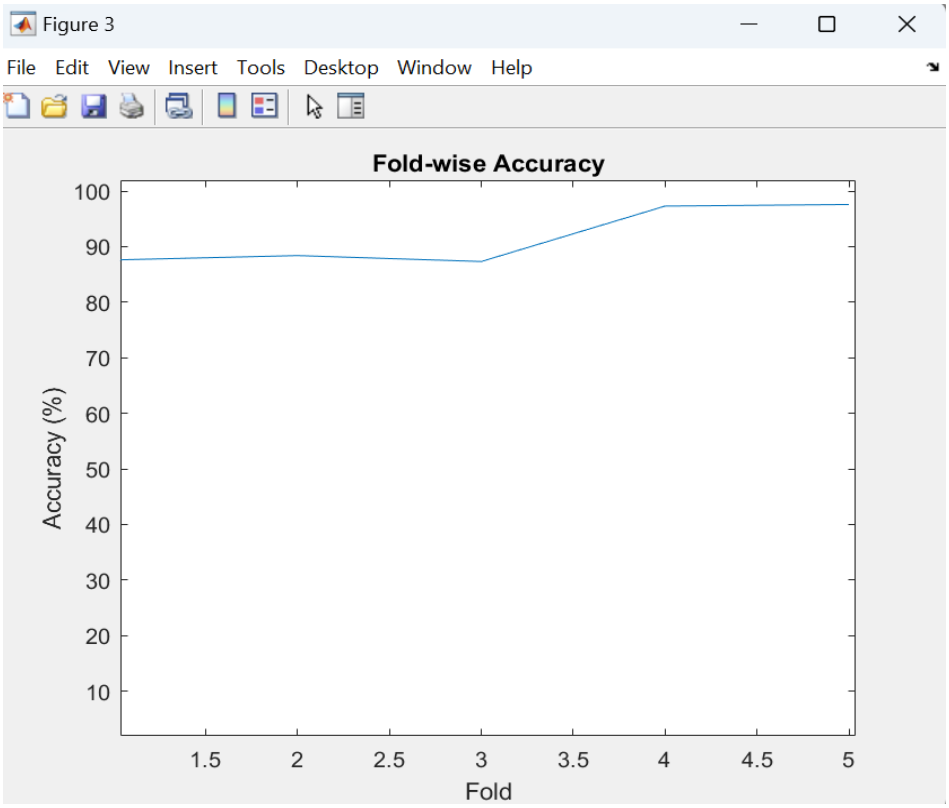

Figure 9 shows the fold accuracy throughout the training process

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205246 |

Email classification and spam detection code:
Another code was built to recall and use the trained and saved BPNN along with the saved test and train datasets. The input text will be processed following the same approach used by the training code i.e. lowercasing the text, removing stop words, and tokenizing the text to identify the vocabulary used during the training through the back of words technique. Finally, a prediction function is used to compare the input text matrix with the saved BPNN that was recalled in the earlier stage of the code. The following are the results of inputting spam and non-spa texts:

IV. CONCLUSION

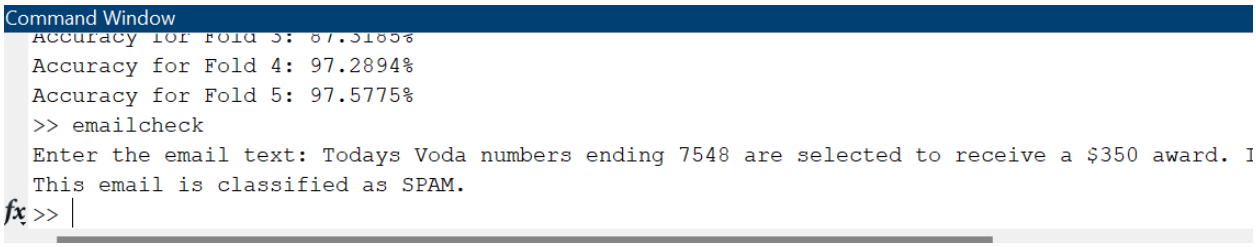

Figure 10 the output when entering a spam text

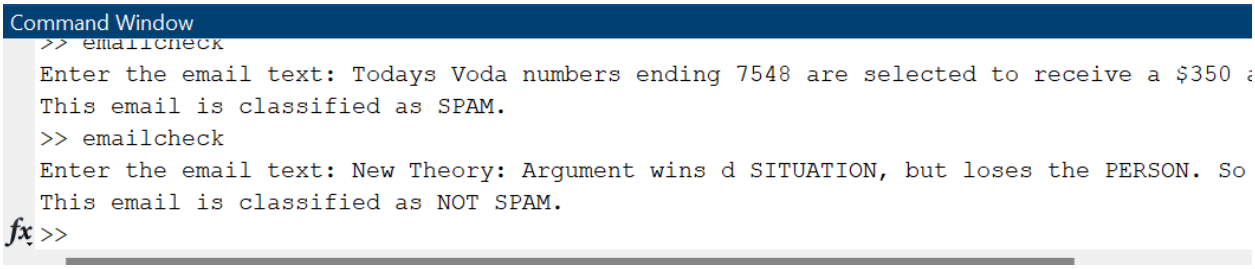

Figure 11 the output when entering a non-spam text

The paper provided a comprehensive explanation of the (ANN) artificial neural networks and (BPNN) back propagation neural networks and also explored how BPNN can be used and trained to detect and filter spam emails using available datasets, the code used achieved a 97% accuracy.

The research subsequently explained the practical application of BPNN in classifying emails as either spam or non-spam. It discussed important procedures such as data preparation, feature extraction, model construction, and evaluation methods such as n-fold cross-validation. The related MATLAB code example demonstrated the steps of implementation, providing a practical illustration of the theoretical topics covered.

The email classification code effectively showcased the practical use of the trained BPNN model in accurately categorizing new email messages. This example highlighted the model's value in effectively tackling the persistent issue of spam email detection.

Lastly, the paper enhances the existing knowledge on spam email detection by offering a thorough examination of the BPNN technique, encompassing its theoretical foundations and actual execution. Utilizing neural networks, BPNN presents a potential resolution to the ongoing issue of spam, allowing users to effectively handle their email exchanges and protect their online security and privacy.

REFERENCES

1. Bishop, C. M. (2006). Pattern recognition and machine learning. Springer google schola, 2, 5-43.
2. Bottou, L. (2010). Large-scale machine learning with stochastic gradient descent. In Proceedings of COMPSTAT'2010: 19th International Conference on Computational StatisticsParis France, August 22-27, 2010 Keynote, Invited and Contributed Papers (pp. 177-186). Physica-Verlag HD.
3. Gomez, J. C., & Moens, M. F. (2012). PCA document reconstruction for email classification. Computational Statistics & Data Analysis, 56(3), 741-751.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205246 |

4. Haykin, S. (1998). Neural networks: a comprehensive foundation. Prentice Hall PTR.
5. Johnson, D. (2024). Back Propagation in Neural Network: Machine Learning Algorithm. Guru99. Retrieved on 31 March 2024 from: https://www.guru99.com/backpropogation-neural-network.html
6. Johnson, D. (2024). Back Propagation in Neural Network: Machine Learning Algorithm. Guru99. Retrieved on 28 April 2024 from: https://www.guru99.com/backpropogation-neural-network.html
7. Kalirane, M. (2023). Gradient Descent vs. Backpropagation: What's the Difference?. Analytics Vidhya. Retrieved on 29 April 2024 from: https://www.analyticsvidhya.com/blog/2023/01/gradient-descent-vs-backpropagation-whats-the-difference/
8. Kaur, K., & Kumar, M. (2015). Spam Detection using KNN, Back Propagation and Recurrent Neural Network.
9. Kriesel, D. (2007). A Brief Introduction to Neural Networks. URL: availableathttp://www.dkriesel.com
10. Kukreja, H., Bharath, N., Siddesh, C. S., & Kuldeep, S. (2016). An introduction to artificial neural network. Int J Adv Res Innov Ideas Educ, 1(5), 27-30. Dongare, A. D., Kharde, R. R., &
11. Kumar, R. K., Poonkuzhali, G., & Sudhakar, P. (2012, March). Comparative study on email spam classifier using data mining techniques. In Proceedings of the international multiconference of engineers and computer scientists (Vol. 1, pp. 14-16). Newswood Limited, Hong Kong.
12. Li, C. H., & Huang, J. X. (2012). Spam filtering using semantic similarity approach and adaptive BPNN. Neurocomputing, 92, 88-97.
13. Li, J., Cheng, J. H., Shi, J. Y., & Huang, F. (2012). Brief introduction of back propagation (BP) neural network algorithm and its improvement. In Advances in Computer Science and Information Engineering: Volume 2 (pp. 553-558). Springer Berlin Heidelberg.
14. Petrosyan, A. (2024). Spam: share of global email traffic 2011-2023. Retrieved on April 21 2024 from https://www.statista.com/statistics/420400/spam-email-traffic-share-annual/
15. Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. nature, 323(6088), 533-536.
16. Tuteja, S. K., & Bogiri, N. (2016, September). Email Spam filtering using BPNN classification algorithm. In 2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT) (pp. 915-919). IEEE.
17. Wu, C. H., & Tsai, C. H. (2009). Robust classification for spam filtering by back-propagation neural networks using behavior-based features. Applied Intelligence, 31(2), 107-121.
18. Yu, B., & Xu, Z. B. (2008). A comparative study for content-based dynamic spam classification using four machine learning algorithms. Knowledge-Based Systems, 21(4), 355-362.
19. Yu, B., & Zhu, D. H. (2009). Combining neural networks and semantic feature space for email classification. Knowledge-Based Systems, 22(5), 376-381.
20. Ayodele, T., Zhou, S., & Khusainov, R. (2010). Email classification using back propagation technique. International Journal of Intelligent Computing Research (IJICR), 1(1/2), 1.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi crossref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details