E-mail Classifications Based on Deep Learning Techniques

Sarah H. Rakad[1,*], Abdulkareem Merhej Radhi[1]

[1]Computer Department, College of Science University AL-Nahrain, Baghdad, 10001, Iraq

Emails: sarahhammed322@gmail.com; abdulkareemradhi@gmail.com

Abstract

Email types sorting is one of the most important tasks in current information systems with the purpose to improve the security of messages, allowing for their sorting into different types. This paper aims at studying the Convolution Neural Network and Long Short-Term Memory (CNN-LSTM), Convolution Neural Network and Gated Recurrent Unit (CNN-GRU) and Long Short-Term Memory (LSTM) deep learning models for the classification of emails into categories such as "Normal", "Fraudulent", "Harassment" and "Suspicious". The architecture of each model is discussed and the results of the models' performance by testing on labelled emails are presented. Evaluation outcomes show substantial gains in precision and throughput to conventional approaches hence inferring to the efficiency of these proposed models for automated email filtration and content evaluation. Last but not the least, the performance of the classification algorithms is evaluated with the help of parameters like Accuracy, precision, recall and F1-Score. From the experiment, the models found out that CNN-LSTM, together with the Term Frequency and Inverse Document Frequency (TF-IDF) feature extraction yielded the highest accuracy. The accuracy, precision, recall and f1-score values are 99. 348%, 99. 5%, 99. 3%, and 99. 2%, respectively.

Keywords: Email classification; Deep learning; Long short-term memory; Convolution neural network- long short-term memory; Cyber security

1. Introduction

E-mailing is an important aspect of the society and now part of the global communication platform, both – interpersonal and inter-organizational. It is common throughout all businesses and organizations to be employed for various tasks like as transmitting important documentations, co-ordinating with teams, sending back reports and even for conveying miscellaneous notifications. Global email users are predicted to reach 4 billion in the next few years and to effectively organize the conference and ensure that all the participants had an easy time, it was advisable to come up with a list of all the important aspects of emails that have to be considered today. The numbers show that by 2023 overall email users would easily hit 3 billion while daily email traffic at 108. Average number of emails per day has reached 7 billion, pointing at the significance of this communication tool in functioning. [1].

Email classification is a very important part of email security because emails are categorized according to their content and intent which then informs proper and timely action if any malicious emails are identified. [2]. It is one of the most valuable resources of information about a great number of unlawful activities on the internet. Besides, the analysis of emails is challenging because hackers can manipulate several fields, and there is a broad range of email clients used, as well as the legal restrictions on the epistemic study of emails [3]. Data mining, which involves use of algorithms for pattern matching in data to generate information for use by management is referred to as. Umb has several uses in the field of digital cyber forensics. They comprise identifying and categorising forensic data by relationships, establishing the forensic association links, identifying informational patterns that aid in predicting, and identifying concealed fact pattern [4]. One form of cyber-attack is phishing which seeks to obtain Credit card numbers, passwords and any other sensitive and personal data of individuals or organizations [5]. Newly emerging news suggest that phishing attacks are on the rise and becoming more advanced, affecting one financially and reputation especially [6]. The potential of using deep learning as well as using machine learning techniques to go beyond the limitations of traditional approaches have sparked much interest over time as it helps increase the accuracy of detecting fake links also known as phishing [7].

DOI: https://doi.org/10.54216/JCIM.150130
Received: April 19, 2024 Revised: June 17, 2024 Accepted: August 24, 2024

At present, most of the phishing attacks are perpetrated through social media, emails and other forms of online platforms. Modern machine learning algorithms are arousing interest in cybersecurity, especially in the case of phishing detection [8]. It has been shown that Employing LSTM networks and CNNs as deep learning approaches to enhance the accuracy of the phishing detection by analysing the formatting and textual content of the phishing messages and further communication through the Internet [9].
Deep learning methods show capabilities to eliminate the limitations of traditional methods and enhance the detection rate; hence, it is a relatively trendy field that has gained much attention in recent years. A systematic review of published literature can build a logical, evidence-based understanding of phishing detection employing deep learning algorithms by aggregating findings of the related studies and identifying the missing links in the literature.

2. Related Work

The following literature survey demonstrate details about existing techniques applied in E-mail analysis with different categories:

Nguyen, Nguyen [10], and Nguyen et al. utilized hierarchical LSTM networks and attention processes successfully tackled email phishing in NLP with a precision of 0.8934, it's essential to acknowledge the study's limitations, such as potential biases, generalizability concerns, or challenges in real-world application. Coyotes, Crypt, et al. 2018 [11] reported notable accuracies for their DL-based phishing email classification method: 95.2% accuracy using CNN for subtask 1 and 93.1% accuracy using RNN for subtask 2. Despite facing challenges with dataset imbalance, the study underscores the model's effectiveness in specific contexts.

Hiransha, M., et al. [12] introduced a model combining CNN and Keras word embedding, achieving a notable accuracy of 95.5% for phishing email detection. Bagui et al. [13] Applied deep learning (LSTM, word embedding, CNN) and machine learning (SVM, naive Bayes, decision tree) techniques to comprehensively integrate for phishing email classification, demonstrating an accuracy of 98.89%. Fang, Yong, et al. [14] proposed THEMIS, a 99.848% overall accuracy RCNN-based model with a low false FPR of 0.043% that has an attention mechanism and multi-level vectors for phishing email detection.

However, the study's current limitation lies in its focus on detecting phishing emails only with headers. Baccouche et al. [15] proposed a multi-label LSTM model for spam and fraud detection, achieving an accuracy of 92.7%, but the study's limitation includes a lack of comparison with other advanced techniques for malicious text detection. The hybrid deep learning approach that Eryılmaz, Sahin, and Kilic et al. [16] proposed used Keras and LSTM to detect Turkish spam emails with 100% accuracy; however, the study's drawback was that it evaluated the method using a small dataset of 800 emails, of which half were identified as spam.

Işik et al. [17] introduced a deep learning approach for email classification, employing MI and WMI for feature selection, with 100% accuracy in Turkish; however, the study's exclusive focus on the Turkish language restricts the generalizability of their findings to broader linguistic contexts. Alotaibi, Al-Turaiki, and Alakeel et al. [18] introduced CNNPD, a CNN-based email phishing detection framework with a promising accuracy of 99.42%, outlining plans to optimize hyperparameters, explore more deep-learning architectures, and assess models on a larger dataset, despite acknowledging the use of a smaller dataset in the current study.

Bagui et al. [19] introduced a pioneering method for phishing email identification, utilizing deep semantic analysis and one-hot encoding with DL and ML algorithms; while reporting CNN with word embedding as the most effective (96.34% accuracy. Alhogail and Alsabih et al. [20] introduced a phishing email classifier leveraging deep learning, specifically GCN and NLP, attaining 98.2% accuracy and a low FPR of 0.015. The study underscored the model's effectiveness in detecting phishing emails based on body text, albeit with a focus on the English language. The authors outlined future plans to incorporate non-English datasets, acknowledging the current limitation of testing exclusively on an English corpus.

Manaswini and Srinivasu et al. [21] introduced Themis, an email phishing detection model with 99.87% overall accuracy and a low FPR of 0.042%, focusing on email structure analysis but acknowledging the limitation of neglecting factors like sender reputation. The authors plan to enhance accuracy by incorporating additional features such as email structure and sender reputation in future iterations. He et al. [22] proposed a double-layer deep learning framework with LSTM and XGBoost achieving 98.35% accuracy for detecting social engineering attacks via phishing emails, yet acknowledged limited generalizability.

Nooraee and Ghaffari et al**.** [23] implemented a deep learning approach with LSTM and Glove for spam email detection, yielding accuracies of 98.39% and 99.49% on two datasets; however, the study's language limitation

DOI: https://doi.org/10.54216/JCIM.150130
Received: April 19, 2024 Revised: June 17, 2024 Accepted: August 24, 2024

underscores the necessity for broader exploration across languages in future research. A solution for fully automated phishing email detection was published by Muralidharan and Nissim et al. [24] using deep ensemble learning to analyze email segments. Their proposed framework beat previous approaches with an AUC of 0.993 and TPR of 5%, and it eliminated the requirement for human feature engineering. Subsequent investigations will concentrate on utilizing federated learning to protect privacy.

3. Proposed Methodology

This paper presents a comprehensive framework aimed at effectively categorizing emails based on their content. In this paper, the proposed methodology for email classification through various deep learning techniques is presented. By describing each stage of the methodology, from data pre-processing to model evaluation, and provides a structured approach towards building a robust email classification system. The current methods for classifying emails result in the loss of important information or irrelevant emails. With these constraints in mind, the following contributions are made in this research:

1.To create a new, effective method for classifying emails into four categories: Normal, Fraudulent, Threatening, and Suspicious emails, this will be accomplished by using LSTM-based GRU, which can handle both short and long sequences with more over 1000 characters.

2.The LSTM-based GRU effectively extracts relevant data from emails that can be used as proof in forensic investigations. Since studying the headers of certain emails is more efficient than evaluating the headers of all emails, email content analysis aids in spoof identification.

3.Compare the results to previous research on email content analysis and email classification as well as classic DL models.

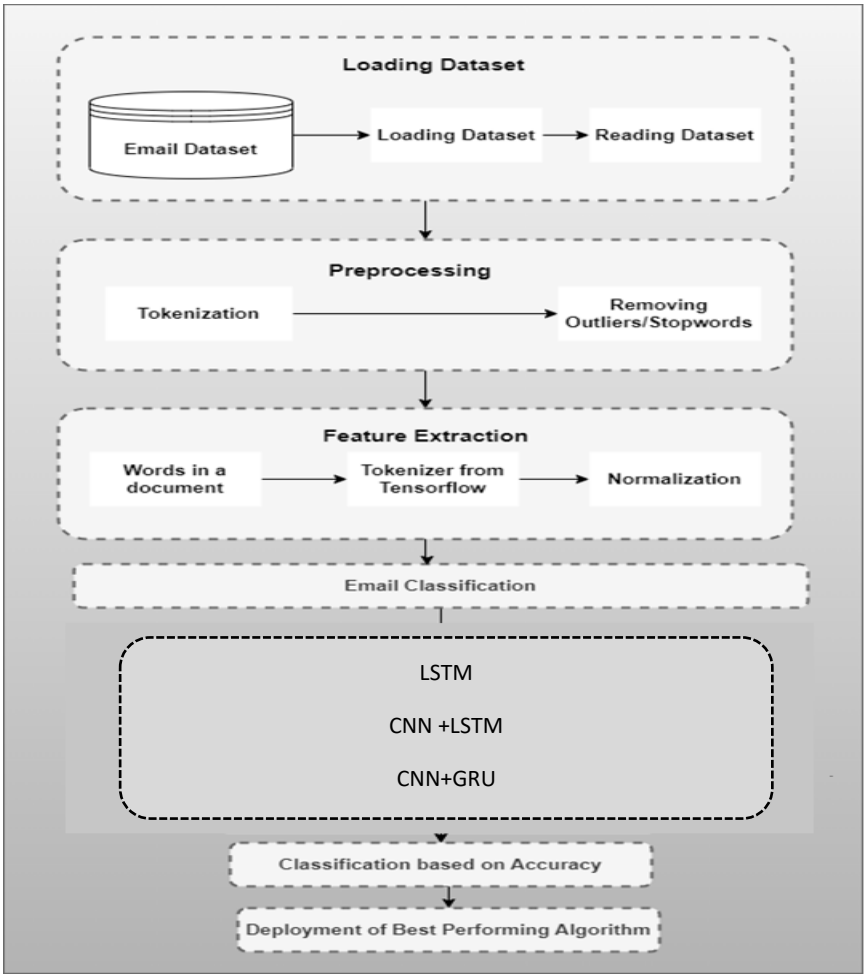

Figure 1. Proposed Methodology

DOI: https://doi.org/10.54216/JCIM.150130
Received: April 19, 2024 Revised: June 17, 2024 Accepted: August 24, 2024

3.1 Dataset

Proposed Dataset is loaded into the project environment from GitHub. The dataset after composition contains about 32,427 messages. The dataset can be used to enhance the security measure of the email servers and the end users. This involves accessing the dataset file containing pertinent data, such as email content for analysis. By loading the dataset, further exploration and manipulation of the data are enabled, preparing it for subsequent pre-processing steps.

3.2 Pre-processing

As part of pre-processing in this phase, textual data retrieved from the dataset are pre-processed. Electronic text is converted into such format that makes the text ready for further analysis or assessment. Text pre-processing is a process of pre-processing text s for being used in a classifying process. This process is needed because by reducing the dimensionality of the feature space and thereby the computational load more accurate classifications may be achieved. Pre-processing of text aspires to be tokenized, removed of stop words, stemmed, lemmatized and normalized; all in a course of enhancing the efficiency of classification. There are many activities involved usually in pre-processing stage with a view of preparing texts for categorization [25]. For instance, where such original words as 'votre', 'travail', as well as any other word in English or any other language that contains special characters/symbols, usernames, unnecessary repeated and long letters, numbers, numerals, and punctuations are removed. The information that is extraneous and which is often found to distort the efficiency and effectiveness of the classification, can be removed.

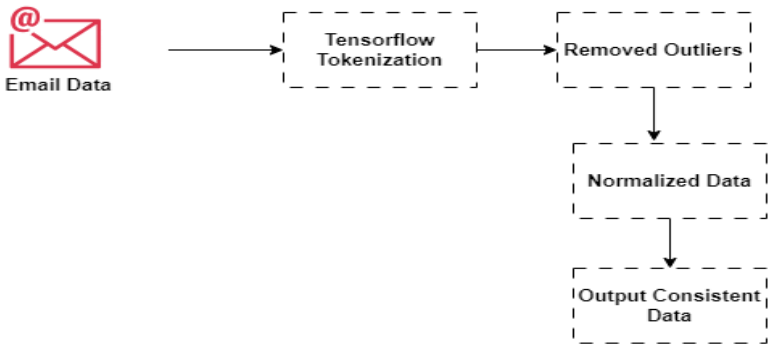

Figure 2. General Block Diagram of Dataset Pre-processing

3.2.1 Plotting the Distribution of Classes

We start exploratory data analysis by visualizing the distribution of classes in the dataset as shown in figure 2 below. This includes the creation of graphic displays such as histograms or bar graphs to show the frequency or the percentage distribution of each class category. Therefore, looking at the classes distribution provides an understanding of the nature and class distribution of the given dataset that later defines the data pre-processing and modelling steps.

Figure 3. Visual Representation of Class Distribution

DOI: https://doi.org/10.54216/JCIM.150130
Received: April 19, 2024 Revised: June 17, 2024 Accepted: August 24, 2024

3.2.2 Oversampling

Since class imbalance is a problem that may be identified in the dataset when analysing the distribution of classes, oversampling methods are applied to its treatment. Oversampling is the process of reproducing some of the instances particularly the minority class samples in order to make the class categories balanced. The effectiveness of the model when tested against all the class categories is boosted when the number of sample in the minority class is increased, further reducing the effect of class imbalance on the training and test phase of the model.

3.2.3 Splitting the Resampled Dataset

After oversampling, the samples the over sample data set is split into different other subsets for training and evaluation. This means partitioning of the dataset into a training set and a testing set, which are often partitioned in a pre-determined technique. This way, a part of the data is used for the model training while the other part is reserved and not used during the model training, thus allowing for estimating the model's ability with the data where it wasn't trained.

3.3 Defining the Model

As the last phase of the proposed work, different models are initialized and learnt for the pre-processed and divided data.

3.3.1 LSTM Model

In this section, we discuss in detail the Long Short-Term Memory – a model used in the process of email classification. To provide an expanded understanding of the model efficiency in the set of emails classification into particular categories we describe the model, training results, and evaluation metrics. The architecture of the LSTM model comprises of several layers that help in processing data that occurs in sequences. For creating LSTM model, we adopted the package from the tensorflow Keras namely the 'Sequential' model. The various components that constitutes the model architecture are as follows; in the first layer of the model, the guess text data of the review are converted into high dimension space vectors. This layer endows the model with the knowledge of relation and semantic relationship both of which are indispensable when it comes to interpreting the meaning of the contents of the posted email. After the embedding layer, we inserted a Hidden LSTM layer, with 128 units. LSTM empower networks to derive word relationship and store considerable information in lengthy sequences thus ideal for sequential input. The features of the email content are sequentially provided to LSTM layer which filters out relevant features and patterns that matches different types of emails. The last layer of the model is a dense layer with softmax activation in the output that allows the probability distribution of the sizes of the plates. Final categorization probability of every email category is produced at this layer. The model was able to identify the class label of a given email based on the likelihood which can be estimated by applying softmax activation that yields probabilities of the class.

3.3.2 CNN+LSTM Model

In paper [2], the authors designed a CNN+LSTM model which was designed for text classification application including email classification. Here you will find our overview of its architecture, training effectiveness, and assessment criteria. CNNs and LSTMs have several stages in their model structure, and all of them serve different purposes in the classification. The embedding layer again helps the model to take care of the synonyms where the similarly embedded dense words in the raw text data is converted into meaningful numerical form. This layer collects essential information required for categorizing one email to another kind of emails. It is used to find local features and characteristics of the text within the given regions of interest. To attempt to minimize the dimensionality while maintaining the crucial information this layer sample down the feature maps that the convolutional layer produced. This implementation has 128 units of LSTM that deals with Sequential data and the analysis of content of emails and extraction of sequential characteristics and pattern is efficiently done here. This layer infers the likelihood distribution over certain classes for example "Normal," "Fraudulent," "Harassment", and "Suspicious" emails. It can be used in multi class classification.

3.3.3 CNN+GRU Model

The CNN+GRU model which has been explained in detail in this section is as depicted in figure 3 so that its architectural demonstration along with the training outcomes and testing metrics can be covered. More particularly designed for text classification tasks, the CNN+GRU is a more complex architecture which could be used for

DOI: https://doi.org/10.54216/JCIM.150130
Received: April 19, 2024 Revised: June 17, 2024 Accepted: August 24, 2024

categorizing emails into the classes such as “Normal”, “Fraudulent”, “Harassment” and “Suspicious”. The following layers make up the CNN+GRU model's architecture: The embedding layer causes the words to be converted into high dimensional vectors in order assist the neural network to understand the relatedness of the words as per relation semantics. He recognises certain features and outlines in the text, and gets the required information needed for the segregation of different types of emails. This layer helps in reducing the dimensionality while trying to retain features that are considered to be critical and this is through down sampling the feature maps. Discovers the temporal relationships and long-range associations between the email messages. Unlike in RNNs, where all the previous memory is removed and then copied when replenishing the data, in GRUs, specific data from the previous data sequence only is retained while other data is discarded. Finally, produces the classification probabilities of every email category with the help of the Dense Layer with Softmax Activation. It will output the probabilities that each email will be belonging to any of the following specified classes which is made by compiling the data of the previous layers.

4. Experiment and Results

4.1 LSTM Model

In more detail, it is important to explain that while the LSTM model is built, the classification performance is constantly increasing in the learning phase with the help of labelled training data and an iterative optimization process. As schematically depicted in figure 4, the model fine tunes its capacity to correctly classify emails during the training phase, by tweaking parameters to eventually minimize the loss function.

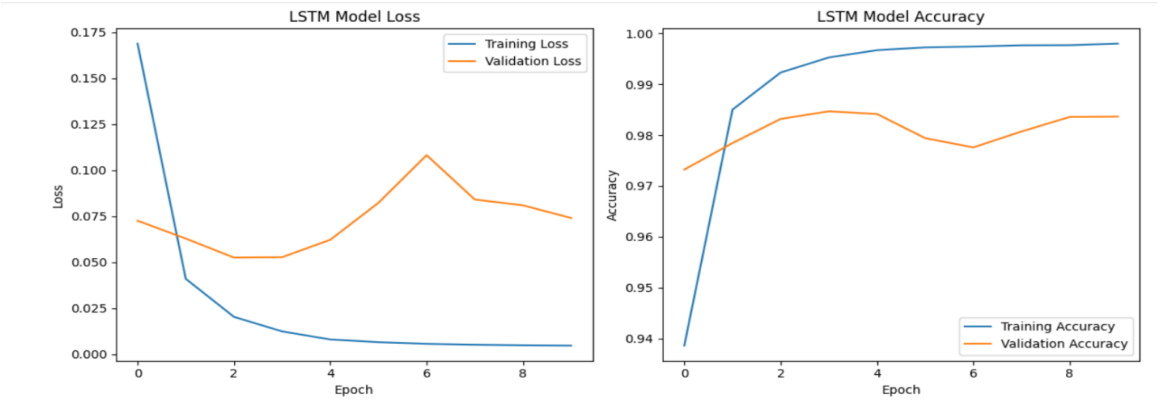

Figure 4. LSTM Training Performance

So, for comparing the performance of LSTM model, the parameters like confusion matrix, accuracy, precision, and recall were analysed. And the classification report once the model had been trained on the training dataset. A detailed analysis of the model in many classes is provided in Section 6 of this paper. Categorization report as shown in figure below we have figure 5. In addition to the mean measurement accuracy of the model, it has for each class and is common to use the subset of measures: precision, recall and F1-score. From the classification report we obtain it is possible to assess how the model copes with the differentiation of classes and identify any possible areas for development.

Classification Report:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	3820
1	0.99	0.96	0.97	3839
2	1.00	0.99	0.99	3856
3	0.95	0.99	0.97	3837
accuracy			0.98	15352
macro avg	0.98	0.98	0.98	15352
weighted avg	0.98	0.98	0.98	15352

Figure 5. LSTM Classification Report

Figure 6 shows an example of a confusion matrix which is a tabular representation of the model’s predictions made against the true values of different classes. It provides the understanding of the various types of mistakes—both in terms of false positive and false negative—that the model can produce. Knowledge of the model strengths and weaknesses will assist in designing better improving Email classification tactics through confusion matrix analysis.

DOI: https://doi.org/10.54216/JCIM.150130
Received: April 19, 2024 Revised: June 17, 2024 Accepted: August 24, 2024

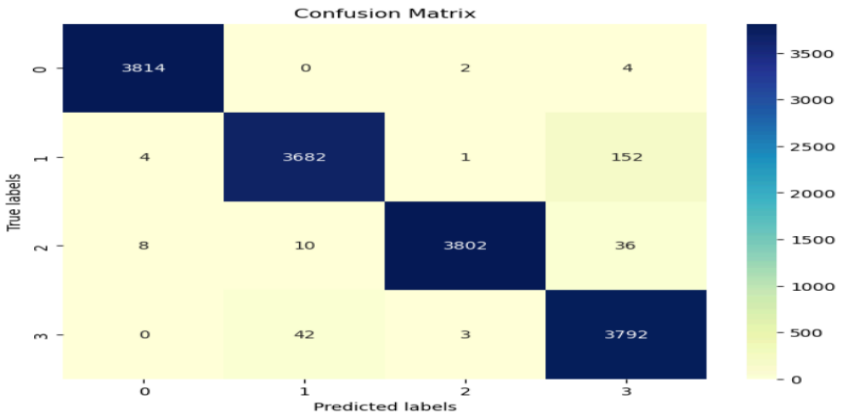

Figure 6. LSTM Confusion Matrix

4.2 CNN+LSTM Model

This mentioned architecture was applied for training the CNN+LSTM model for the given dataset. Through an iterative training process, the model was taught to map incoming email data to pertinent class labels during the training phase. During training, key performance indicators could include anything like: An indicator of how well the model fits the training data, it is the model's mistake during training. A measure of overfitting that is comparable to training loss but determined using a different validation dataset. The percentage of correctly categorized occurrences in the training dataset presented in figure 7 is known as training accuracy. The percentage of occurrences in the validation dataset that are correctly classified.

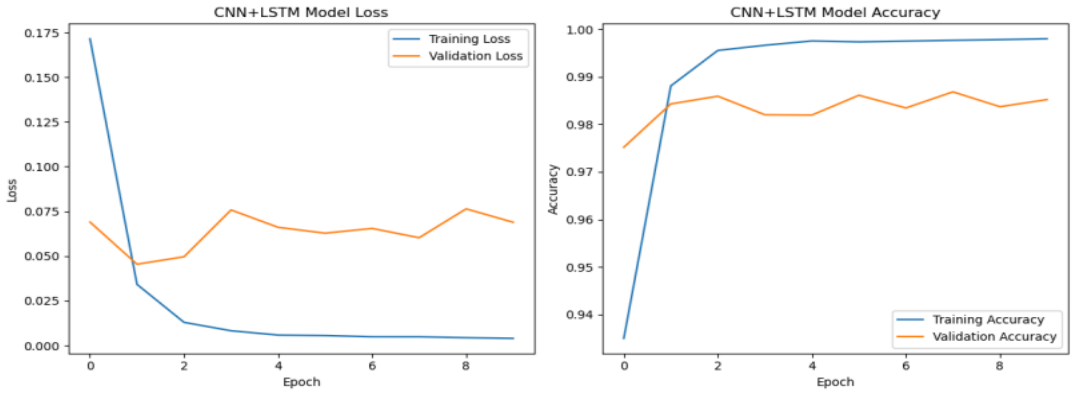

Figure 7. CNN+LSTM Training Performance

Following training, the CNN+LSTM shown in figure 8, model's performance in email classification was assessed using a variety of criteria. The following section discusses two main assessment measures that are frequently used for categorization tasks:
A thorough analysis of the model's performance for every class label, including measures like precision, recall, F1-score, and support, is given in the classification report. These metrics provide information about how well the model classifies.

```
Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      3820
           1       0.99      0.97      0.98      3839
           2       1.00      0.99      0.99      3856
           3       0.96      0.99      0.97      3837

    accuracy                           0.99     15352
   macro avg       0.99      0.99      0.99     15352
weighted avg       0.99      0.99      0.99     15352
```

Figure 8. CNN+LSTM Classification Report

DOI: https://doi.org/10.54216/JCIM.150130
Received: April 19, 2024 Revised: June 17, 2024 Accepted: August 24, 2024

The model's predictions compared to the actual class labels are tabulated in the confusion matrix. For every class, it lets us see how well the model performs in terms of true positives, false positives, true negatives, and false negatives. This scenario as shown in figure 9 can obtain more measures, including accuracy, precision, recall, and F1-score, from the confusion matrix.

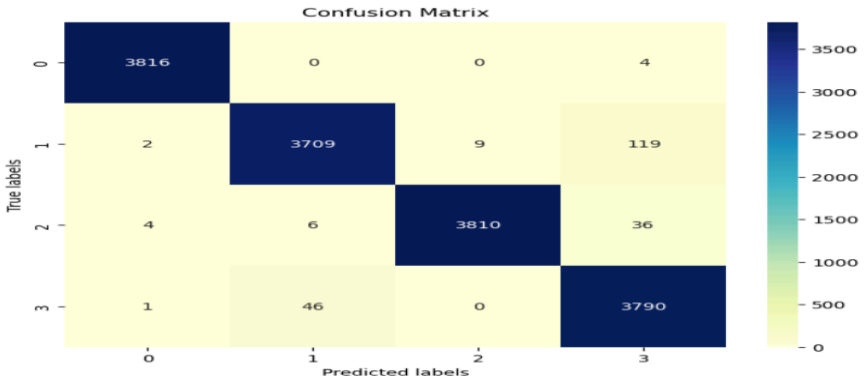

Figure 9. CNN + LSTM Confusion Matrix

4.3 CNN+GRU Model

During the training phase, CNN+GRU takes information from the labelled training data set to attribute the emails into the categories such as “Normal”, “Fraudulent”, “Harassment”, and “Suspicious”. In order to decrease the classification loss and increase the probability of the correct output then the parameters of the model is updated iteratively. It is also possible to evaluate the training accuracy as well as the loss of CNN+GRU for the model over as many epochs as possible. Fig 10 A clearly shows that while training continues the model is learning how to sort the mails more appropriately shown by a declining training loss and rising accuracy.

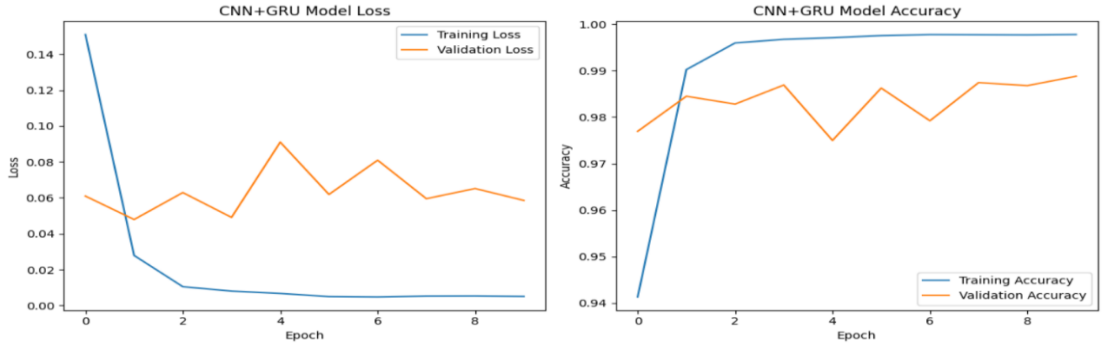

Figure 10. CNN+GRU Training Performance

After training, for evaluating newly generated data not encountered in training, various parameters are used for analysing CNN+GRU Stack. Among the assessment metrics are Evaluations such as precision, recall, and F1-score for each of the trains is provided in the classification report that contains a detailed analysis of the model’s accuracy. Precision is the percentage of examples correctly classified as belonging to such or such a class out of all the instances as to which the classifier anticipated they would fall into that class. Recall is the ability to identify all the accurately predicted instances of a specific class of example that is being tested out of the total number of instances in the class. In figure 11 it shows that the harmonic mean of the precision and the recall gives us the F1 value which provide a balanced measure of the accuracy of the model.

Classification Report:

	precision	recall	f1-score	support
0	1.00	1.00	1.00	3820
1	0.98	0.97	0.98	3839
2	0.99	0.99	0.99	3856
3	0.98	0.98	0.98	3837
accuracy			0.99	15352
macro avg	0.99	0.99	0.99	15352
weighted avg	0.99	0.99	0.99	15352

Figure 11. CNN+GRU Classification Report

DOI: https://doi.org/10.54216/JCIM.150130
Received: April 19, 2024 Revised: June 17, 2024 Accepted: August 24, 2024

The confusion matrix which is illustrated in figure 12 shows a detailed comparison of model prediction and actual class labels for each class. As seen above, it allows us to determine the genuine positives, genuine negatives, the impostors, and the faint positives the model displays. This may be obtained from the confusion matrix that gives a summary of the true positive and false positive values, true negative and false negative values for every one of the classes and from these true positive and true negative get the accuracy negative, precision and recall for every class giving finally the F1 score for each class. Matrix into which authors provide beneficial data regarding the model with points of strengths and weaknesses regarding the approach implemented for email classification

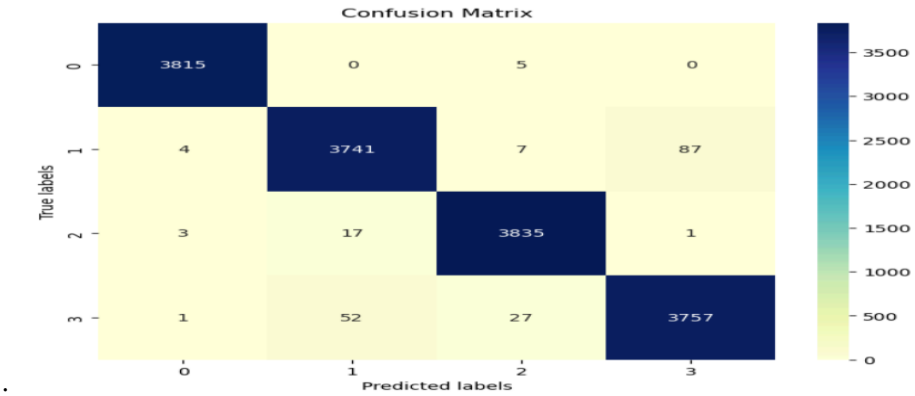

Figure 12. CNN+GRU Confusion Matrix

5 Comparative Analysis

Table 1 calls for a range of Deep learning techniques that we used in our tests in analysing the performance of different categorization models indicated in table 2 below. These models were trained using the email dataset, and to assess the model's performance accuracy, precision, recall, and F1-score metrics were adopted.

Table 1: our proposed system's results

Model	**Accuracy**	**Precision**	**Recall**	**F1-score**
CNN + LSTM	0.99	0.99	0.99	0.99
CNN + GRU	0.97	0.97	0.97	0.97
LSTM	0.98	0.98	0.98	0.98

Table 1: Comparison with SeFACED Results

Model	Accuracy %	Precision	Recall	F1-Score
CNN - LSTM	0.9316	0.93	0.93	0.93
LSTM - GRU	0.9500	0.95	0.95	0.95

Combining CNNs and GRUs for email categorization is a sophisticated solution that uses both convolutional and recurrent neural network architectures. of 1st study, we achieved 99% accuracy, precision, recall and F1-score while using the CNN + GRU model. The model was proved useful in classifying the emails in terms of these measures. Facebook – CNN: Meet GRU that classifies 99% of the emails in the dataset. It therefore means 99% of the emails that where categorized on a given class were classified indeed on the right class. Therefore if the percentage marker was 99% recall means the model located 99% of necessary emails. Accuracy in the model has been pegged at 99 % while precision and recall are equal in the F1-score of the model. It is for the same reason that the results of the SeFACED paper are somewhat poorer in terms of accuracy, precision, recall, and F1-score, standing at 95%. As observed from the evaluations in the SeFACED study, the CNN + GRU had slightly inferior performances in comparison to the 1st research although it has in general achieved remarkable results.

One of the most efficient strategies to email categorization is based on Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) for sequence modelling and classification. As for the 1st study, varied results were obtained with the use of Long Short Term Memory where LSTM neural network stated an accuracy, precision, recall, and F1-score of 98%. Such numbers would testify the ability of the model in correctly classifying emails.

DOI: https://doi.org/10.54216/JCIM.150130
Received: April 19, 2024 Revised: June 17, 2024 Accepted: August 24, 2024

The proposed LSTM + GRU model shown 99% accuracy, which is very much correct, in the classification of emails in the given dataset. Specificity means 99 percent of emails, which was predicted by the model to belong to a class, were correctly classified by the model, therefore, a high degree of precision. Successful classification of 99% of various related emails highlight the capability of the model to cover all instances of relevant email data. The precision metric provides the model with the level of accurateness as 99% and the recall gives it a fair balance by providing 92%. Comparatively for LSTM + GRU the SeFACED article states that the accuracy, precision, recall, and F1-score are as low as 95%. Although, such measures indicate that LSTM + GRU reproduced slightly worse in SeFACED compared to in our system. The implemented difference in performance of the two studies could be attributed to differences in properties of the dataset, pre-processing, feature extraction, model and evaluation measures used. It also must be mentioned that the volume, the number of different types, and the level of complexity of emails used in these studies may also influence LSTM algorithm's effectiveness.

6 Conclusion

All in all, based on our analysis of the approaches to the classification of emails we can conclude about the challenges and opportunities of such task. It is through this analysis and consideration of many classification approaches and distribution of data in this paper that the author is able to come up with a better feel of the variables which influences the email classification system. The results of analysis, therefore, highlighted class imbalance by presenting significant variability in the distribution of the samples by the different groups. The "Fraudulent" class had fewer instances than the other classes even though it was clearly defined, the "Normal" had the highest instances. This imbalance is an issue to classification models particularly when it comes to accurately predicting the minority classes. In this paper, CNN-LSTM, CNN-GRU, and LSTM-GRU models are employed for labelling the given email as "Normal", "Fraudulent, "Harassment" or "Suspicious". Experimental results show that deep learning indeed performs well in automated email filtering and content analysis both in terms of high accuracy and insensitivity to the types of emails present. It was demonstrated that class imbalance must be addressed and that using a model variety improves classification results. This is why feature engineering techniques are more powerful and as a result, introduced more subtle information about the email content to our models. Last, the current proposed study contributes to other studies on categorization of emails by comparing the relative effectiveness of proposed strategies and directions for future research. This study presented problems that the researcher can work to eradicate and advance the ap activity and the email system security, administration, and user experience through these advised research avenues.

Conflicts of Interest: "The authors declare no conflict of interest."

References

[1] Charalambou, E, Bratskas, R, Karkas, G, & Anastasiades A, " Email forensic tools: A roadmap to email header analysis through a cybercrime use case" , Journal of Polish Safety and Reliability Association Summer Safety and Reliability Seminars, Vol. 7, No. 1, 2016.

[2] Tsochataridou, C, Arampatzis, A, & Katos, V, " Improving Digital Forensics Through Data Mining" , In IMMM 2014, The Fourth International Conference on Advances in Information Mining and Management, September, 2016.

[3] Korasidi Andriana Maria, " Authorship Attribution Forensics: Feature selection methods in authorship identification using a small e- mail dataset", M. Sc. Thesis, University of Athens, 2016.

[4] Bhardwaj, A. K, & Singh, M, "Data mining- based integrated network traffic visualization framework for threat detection", Neural Computing and Applications, 26(1), 117- 130, 2015.

[5] Alshingiti, Z.; Alaqel, R.; Al-Muhtadi, J.; Haq, Q.E.U.; Saleem, K.; Faheem, M.H. A Deep Learning-Based Phishing Detection System Using CNN, LSTM, and LSTM-CNN. Electronics 2023, 12, 232.

[6] Tsohou, A.; Diamantopoulou, V.; Gritzalis, S.; Lambrinoudakis, C. Cyber insurance: State of the art, trends and future directions. Int. J. Inf. Secur. 2023, 22, 737–748.

[7] Safi, A.; Singh, S. A systematic literature review on phishing website detection techniques. J. King Saud Univ. Comput. Inf. Sci. 2023, 35, 590–611.

[8] Chen, D.; Wawrzynski, P.; Lv, Z. Cyber security in smart cities: A review of deep learning-based applications and case studies. Sustain. Cities Soc. 2021, 66, 102655.

[9] Adebowale, M.A.; Lwin, K.T.; Hossain, M.A. Deep learning with convolutional neural network and long short-term memory for phishing detection. In Proceedings of the 2019 13th International Conference on Software, Knowledge, Information Management and Applications (SKIMA), Island of Ulkulhas, Maldives, 26–28 August 2019; pp. 1–8.

[10] Nguyen, M.; Nguyen, T.; Nguyen, T.H. A deep learning model with hierarchical lstms and supervised attention for anti-phishing. CEUR Workshop Proc. 2018, 2124, 29–38.

DOI: https://doi.org/10.54216/JCIM.150130
Received: April 19, 2024 Revised: June 17, 2024 Accepted: August 24, 2024

[11] Coyotes, C.; Mohan, V.S.; Naveen, J.; Vinayakumar, R.; Soman, K.P.; Verma, A.D.R. ARES: Automatic rogue email spotter. In Proceedings of the 1st AntiPhishing Shared Pilot at 4th ACM International Workshop on Security and Privacy Analytics (IWSPA), Tempe, AZ, USA, 1–11 March 2018.

[12] Hiransha, M.; Unnithan, N.A.; Vinayakumar, R.; Soman, K.; Verma, A.D.R. Deep learning based phishing e-mail detection. In Proceedings of the 1st AntiPhishing Shared Pilot at 4th ACM International Workshop Security Privacy Analytics (IWSPA), Tempe, AZ, USA, 1–11 March 2018; pp. 1–5.

[13] Bagui, S.; Nandi, D.; Bagui, S.; White, R.J. Classifying phishing email using machine learning and deep learning. In Proceedings of the 2019 International Conference on Cyber Security and Protection of Digital Services (Cyber Security), Oxford, UK, 3–4 June 2019.

[14] Fang, Y.; Zhang, C.; Huang, C.; Liu, L.; Yang, Y. Phishing Email Detection Using Improved RCNN Model With Multilevel Vectors and Attention Mechanism. IEEE Access 2019, 7, 56329–5634.

[15] Baccouche, A.; Ahmed, S.; Sierra-Sosa, D.; Elmaghraby, A. Malicious text identification: Deep learning from public comments and emails. Information 2020, 11, 312.

[16] Eryılmaz, E.E.; ¸Sahin, D.Ö.; Kılıç, E. Filtering turkish spam using LSTM from deep learning techniques. In Proceedings of the 2020 8th International Symposium on Digital Forensics and Security, ISDFS, IEEE, Beirut, Lebanon, 1–2 June 2020; pp.

[17] Isik, S.; Kurt, Z.; Anagun, Y.; Ozkan, K. Spam E-mail Classification Recurrent Neural Networks for Spam E-mail Classification on an Agglutinative Language. Int. J. Intell. Syst. Appl. Eng. 2020, 8, 221–227.

[18] Alotaibi, R.; Al-Turaiki, I.; Alakeel, F. Mitigating email phishing attacks using convolutional neural networks. In Proceedings of the 2020 3rd International Conference on Computer Applications & Information Security (ICCAIS), IEEE, Riyadh, Saudi Arabia, 19–21 March 2020; pp. 1–6.

[19] Bagui, S.; Nandi, D.; Bagui, S.; White, R.J. Machine learning and deep learning for phishing email classification using one-hot encoding. J. Comput. Sci. 2021, 17, 610–623.

[20] Alhogail, A.; Alsabih, A. Applying machine learning and natural language processing to detect phishing email. Comput. Secur.2021, 110, 10241.

[21] Manaswini, M.; Srinivasu, D.N. Phishing Email Detection Model using Improved Recurrent Convolutional Neural Networks and Multilevel Vectors. Ann. Rom. Soc. Cell Biol. 2021, 25, 16674–16681.

[22] He, D.; Lv, X.; Xu, X.; Yu, S.; Li, D.; Chan, S.; Guizani, M. An effective double-layer detection system against social engineering attacks. IEEE Netw. 2022, 36, 92–98.

[23] Nooraee, M.; Ghaffari, H. Optimization and Improvement of Spam Email Detection Using Deep Learning Approaches. J. Computer. Robot. 2022, 15, 61–70.

[24] Muralidharan, T.; Nissim, N. Improving malicious email detection through novel designated deep-learning architectures utilizing entire email. Neural Networks 2023, 157, 257–27

[25] E. Haddi, X. Liu, and Y. Shi, "The role of text pre-processing in sentiment analysis," Procedia Comput. Sci., vol. 17, pp. 26–32, 2013.

DOI: https://doi.org/10.54216/JCIM.150130
Received: April 19, 2024 Revised: June 17, 2024 Accepted: August 24, 2024