Digital Forensics and Machine Learning to Fraudulent Email Prediction

2022 Fifth National Conference of Saudi Computers Colleges (NCCC) | 979-8-3503-3368-8/22/$31.00 ©2022 IEEE | DOI: 10.1109/NCCC57165.2022.10067685

1st Norah Al-Ghamdi
College of Computer and Information Systems
Umm Al-Qura University
Makkah, Saudi Arabia
s44180522@st.uqu.edu.sa

2nd Tahani Alsubait
College of Computer and Information Systems
Umm Al-Qura University
Makkah, Saudi Arabia
tmsubait@uqu.edu.sa

***Abstract*—E-mail is widespread in the modern commercial environment, providing an appropriate and efficient method for communication. However, today's e-mail security threats are multiplying at an unprecedented rate. Sending phishing, spoofing, spam, and scam e-mails attempting to gain access to victims' personal or financial information is a common way e-mail is used to commit a crime. The criminal activity needs to be combated through digital forensics. Unfortunately, cyber events are becoming significantly challenging, and human capabilities are limited. Using the SeFACED dataset, this research proposes a content base, E-mail multi-classification, into four different classes: Normal, Fraudulent, Threatening, and Suspicious, using four primary Machine Learning algorithms, namely Naïve Bayes (NB), Support Vector Machine (SVM), Logistic Regression (LR), and Random Forest (RF). In addition, the Term Frequency-Inverse Document Frequency (TF-IDF) and Word2vec are also used as feature extraction techniques to compare their results. The findings show that the best accuracy is achieved with the RF, LR, SVM model by the TF-IDF feature extraction with an accuracy of 95%. In addition, the result for classification with TF-IDF outperformed Word2vec classification.**

***Index Terms*—Multi-classification, Email Fraud Detection, Machine Learning, E-mail Forensics, Cybersecurity.**

I. Introduction

Artificial Intelligence (AI) is the science of making machines think or perform human tasks and make their own decisions without human intervention (e.g., Visual Recognition, Natural Language Processing, etc.). AI can be viewed as the things that can carry the human tasks and make these tasks easy [1]. According to the "AI Predictions 2021" report by PwC [2], United States companies are increasing their AI investments. 52% of respondents to this survey have accelerated their plans to adopt AI in the wake of the COVID-19. The results will be felt in the coming years. These "accelerating" companies cite new use cases for AI (40%) and increased AI investment (also 40%) as the most important changes. Of all respondents to this survey, 86% say AI will be a "mainstream technology" in their company by 2021.

As AI becomes more available to everyone, the potential for the technology to be used for malicious activity also increases significantly. To prevent this malicious activity and be ready for the digital forensic challenges to crimes performed by AI and Machine learning (ML), it is essential to analyze and evaluate the technology [3] and utilize it in the Digital Forensics (DF) field.

The Digital Forensic Research Workshop (DFRWS) has defined DF as the use of scientific methods for preserving, collecting, validating, identifying, analyzing, interpreting, documenting, and presenting the digital evidence. Digital evidences are derived from digital sources to facilitate or promote the reconstruction of events determined to be criminal or help prevent unauthorized acts that prove disruptive to planned operations [1].

The use of AI in Digital Forensics is referred to as machine learning forensics; it realizes the patterns of a crime and predicts the malicious activity or finds a sign from a similar kind of crime's history [4]. Furthermore, behavioral analysis performed by machine learning models can dramatically improve the performance of modeling, profiling, and prediction processes in law enforcement systems [5].

The main contribution of this research is to propose an ML model that learns from historical activities to predict fraudulent e-mails in advance to avoid cybercrime victimization and support the investigation of DF. Moreover, our model promises to provide a 95% accurate assessment of e-mail content data obtained from e-mails forfeited from a crime scene.

This paper has been structured in such a way so that the background for it is addressed first, which provides background about the research field of DF, the basic concept of ML Algorithm, and fraud Emails detection techniques. Section 3 briefly describes the previous works related to this research. Section 4 explains the methodology used in this research by describing the dataset used to train the model, presents the preprocessing setup of the research, and the experiment of the four algorithms SVM, LR, RF, and NB. Section 5 presents the results of the four algorithms and then the comparison between them. Section 7 shows the conclusion with what has been done in this research and proposes future work to enhance the model.

2022 NCCC, December 21-22, 2022
979-8-3503-3368-8/22/$31.00 © 2022 IEEE

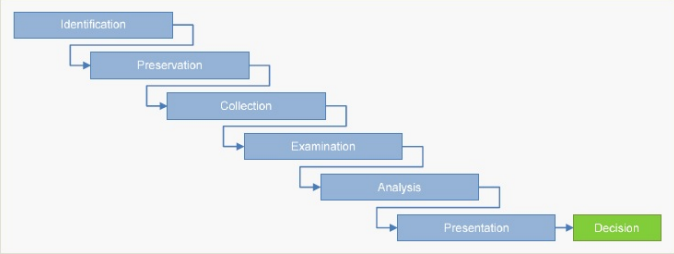

Fig. 1. The Digital Forensic Investigation Process [6]

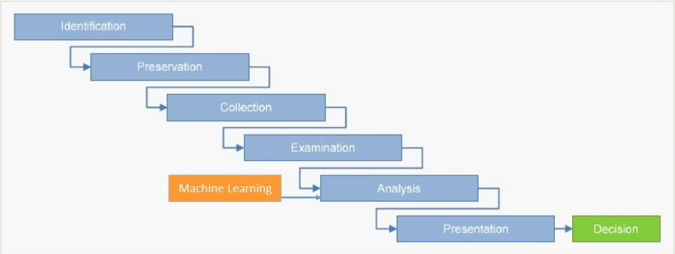

Fig. 2. Role of Machine Learning in Digital Forensics Investigation.

II. Background

A. Digital Forensics

Digital forensics is described as a division of forensic science devoted to retrieve and investigate digital data. It is an evolving domain that is constantly developing to bear with devices alteration and how they are employed to identify, preserve, analyze, and recover data from computer systems and other digital storage [7]. The aims of digital forensics are the identification of the evidence, documenting the crime, collecting and preservation of the evidence, packaging the evidence, and transporting the evidence in a not manipulative manner [8].

B. The Digital Forensics Investigation Process

In the Digital Forensics Investigation Process, the investigator examines digital devices to generate evidence linked to a crime. As illustrated in Digital Forensic Research Workshop (DFRWS), the Process consists of six steps, as stated in Fig1. All the components, devices, and data related to the crime are determined in the identification step. The second step is the preservation phase, which avoids activities that can damage the handled digital information. In the collection step, the digital information related to the incident under investigation is assembled. Next is the examination phase, used for an in-depth systematic examination of evidence linked to the investigated incident. Then, the analysis phase is where the investigator obtains a determination for the evidence collected in the previous phase. In the final step, named the presentation phase, findings are reviewed and shown to the court of law [6].

C. Challenges in Digital Forensics and Role of ML

One of the fundamental challenges in digital forensics is the enormous volumes of unstructured data, frequently with natural ambiguity and errors. Accordingly, every digital forensics phase is hugely time and resource consuming, usually passing the available investigation time and resources.

The Digital Forensic Investigation Process includes six-phase, as previously described above. To overcome the challenges that exist in the Digital Forensics field nowadays, it has been a real focus on taking advantage of big data, automation, computational intelligence, and machine learning in the forensic process [9].

The first phase, the ***identification phase***, is possibly assisted by intelligent detection and identification methods. Furthermore, automated remote evidence acquisition tools with built-in evidence integrity assurance may aid the ***collection phase***. Moreover, the ***examination phase*** could be assisted by automated data restoration and data minimization. In addition, in the ***analysis phase***, which we are interested in in this research, ML techniques could be implemented Fig2. The investigative team handles this step based on the results of the examination of the evidence. Identifying relationships between fragments of data, analyzing hidden data, determining the significance of the information obtained from the examination phase, reconstructing the event data based on the extracted data, and arriving at proper conclusions. Those are some of the activities to be performed at this stage [10]. Finally, the ***presentation phase*** can benefit from a broad range of visualization tools and built-in reports generation.

III. Literature Review

A. Machine Learning-based Fraud Detection Approaches

Recently, many researchers have tried to apply machine learning methods for email spam detection. For example, in [11] four machine learning techniques are used to detect illegitimate fraud e-mails from legitimate E-mail. The authors used Decision Tree (DT), Random Forest (RF), Naïve Bayes (NB), and Support Vector Machine (SVM) classifiers for the experiments. Two different datasets are used for their experimental results. Results showed that Support Vector Machine has the best performance with more than 98% achieved accuracy, precision 0.985, recall 0.97 and F1-score 0.97. Meanwhile, the Decision Tree got the lowest results among the other classifiers with an accuracy of 96.19%.

In [12], the authors seek for a classifier that is suitable for text classification. Additionally, they evaluated machine learning algorithms for spam emails detection. The selected dataset contains two categories of documents: ham and spam with 960 emails. They used the WEKA tool for their experimental work. The Email dataset was preprocessed for precise and good features acquisition. Email body data was extracted and converted into numerical form by tokenizing the text and numeric representation, stop words are removed, and Tf-IDF of the document is calculated. Bag of word was used for representation of text and the N-gram technique was applied,

which works as a sliding window for semantic accuracy, having a variable length. The training is based on a NB model on each of the training sets. Outcomes show that the NB algorithm gives adequate accuracy and precision.

Euna et al. [13] presented a content-based spam email detection approach achieving 97.6% accuracy using the SVM classifier. They used SVM, MNB, LR, and DT classifiers for learning the various features from the contents of emails. In addition, several preprocessing steps were performed before extracting features from the text, namely: Special character removal, Stop words removal, Tokenization and Lemmatization. Moreover, they have used different features such as word2vec, word n-grams, character n-grams, and a combination of variable length n-grams for comparative analysis in their proposed approach. The dataset contains 5731 emails. In conclusion, SVM has proven to be the best classifier and effectively recognizes spam emails. Furthermore, with the combination of uni-gram and bi-gram, SVM achieves the highest accuracy of 97.6%.

There are many criteria to filter emails, domain names, content of the email and more. Authors in [14] proposed project detection is designed especially for filtering mails according to the body (content) of the email. The dataset is obtained from the "Kaggle" website for training. The name of the dataset used is "spam.csv". Furthermore, to test the trained model, a different CSV file is developed with unseen data. Data that is not used for the training of the model; named "emails.csv" Several steps for data preprocessing were followed: Data cleaning, Data Integration, Data reduction, Stop words, Tokenization, Bag of words. As for classifiers, seven of them were used: SVM, K-Nearest Neighbour (K-NNr), NB, DT, RF, AdaBoost Classifier, and Bagging Classifier to check and compare the results for greater accuracy. The result shows that the MNB gives the best outcome but has limitations due to class-conditional independence, making the machine misclassify some tuples.

The advent of social media and the increase of online discussions has dramatically increased cyber harassment. As a result, various social discussion portals and social networking websites have devised policies to discipline offenders. However, identifying the nature and extent of negativity/vulnerability of a comment is a significant challenge. Study [15] presents a method for classifying online comments in terms of their toxicity level, which can assist filter out those who are harassing others. It will eventually improve policy implementation and assign punishment to those who do not respect it, reducing toxicity levels in online discussions. Two approaches they proposed in their study to address this dilemma. The first approach is to train classifiers for each facet of toxicity in comments separately. The second approach handles the dilemma as a multi-label classification dilemma. In addition, various machine learning approaches such as LR, NB, and DT classification are used for this study. They used a total of 1,59,571 comment observations as a dataset, and for each observation, six binary class labels are possible(toxic, severe toxic, obscene, threatening, insulting, identity hate). A significant improvement in accuracy for the simple classification models using their novel preprocessing strategy. As for the experimental results, logistic regression produces better results for both binary classification and multi-classification.

Reference [16] suggests an approach with machine learning for multi-class categorization of emails in a simple text classification method. The data sets contain 1,000 emails with predefined classes: social, promotional, educational, spam and general. The classification task proposed in their paper uses the fastText library by Facebook. In terms of feature extraction, the fastText library itself extracts features by using the skipgram method; however, their proposed model was enhanced by other linguistic feature extraction techniques. The accuracy acquired from the technique described in the paper was 92.5%, which is very close to those found from other techniques for binary classification and higher than most methods for multiple classes. The overall result derived in that method demonstrates the need of preserving likely words' importance. FastText has an advantage over many other classification techniques for its simplicity that most of the work required a neural network working on multinomial logistic regression.

The comparative study [17] between traditional machine learning algorithms used for classification, and Deep learning architectures reveals that, generally, DL achieves higher accuracy than the ML. Their work presented how one-hot encoding can be used for phishing vs. non-phishing email classification with and without word-phrasing. Furthermore, they compared the accuracy of the different ML and DL models with and without word phrasing. The email dataset they used have 18366 labeled emails, of which 3416 are phishing emails, and the rest are regular emails. Their work focused on analyzing the text content of the emails and classifying them as phishing or not. A comparison of the results of several machine learning classifiers, NB, SVM, DT, DL classifiers, LSTM, CNN, and Word Embedding have been presented. On average, the DL models performed slightly better than the ML models, except for SVM, accuracy with word phrasing was lightly better than without the word phrasing, which means that the meaning of the email language is essential in deciding whether an email is a phishing or not.

B. Deep Learning-based Fraud Detection Approaches

There is some related work that apply Deep Learning methods in email spam detection, the authors in [18] design a novel efficient approach named SeFACED whish uses Long Short-Term Memory with Gated Recurrent Neural Network for E-mail classification into four different classes: Normal, Fraudulent, Threatening, and Suspicious E-mails by using Long Short-Term Memory (LSTM) based Gated Recurrent Neural Network (GRU), that not only deals with short sequences as well long dependencies of 1000+ characters. The LSTM based GRU efficiently captures meaningful information from E-mails that could be used for forensic analysis as evidence. Furthermore, E-mail content analysis helps spoof identification since it is more efficient to analyze the headers of specific E-mails than all E-mails. They extracted features from

emails applying TF-IDF, Word2vector, and Word Embedding to classify them. The authors evaluated the performance of SeFACED compared to traditional ML as well as DL models. Results demonstrate that the SeFACED effectively classify E-mail content with an accuracy of 95.0%, the precision of 95.0%, recall of 95.1%, and f-score of 95.1%.

Kaddoura et al. [19] propose a spam detection mechanism based on neural networks using the Enron dataset. There proposed methodology aims to filter the e-mails into two classes: Ham and Spam. Furthermore, two different deep learning approaches for feature extraction were applied: Feed Forward Neural Network (FFNN) and Bidirectional Encode Representations from Transformers (BERT). FFNN model has been studied to test their performances to segregate e-mails as spam or ham experiments on the Enron dataset. Then they compare this model with the BERT dataset. They considered different variations of its architecture in terms of the number of layers, number of neurons per layer, and number of neurons per input layer and measured the corresponding F1-score for each architecture.

Another study aims to capitalize on the open issue of email classification using deep learning approaches [20]. The goal is to explore the options for applying deep learning techniques for email classification and compare the result between various techniques and machine learning algorithms under the deep learning frame. The research project has undertaken three datasets (ENRON Intent, University Help Desk, and Ask-Teacher), and the MLP algorithm was analyzed against each dataset. Once the data is Pre-Processed, they fed the data to the model and analyzed the algorithm's results by optimizing the network. Which include two phases; firstly, they varied the values of the various parameter in the primary network to reach the most optimized results, then they tried using the advanced features of the network and see their variation on the results. In the scope of the research, their basic parameters are the Number of Epochs, Drop-out Ratio, and Batch size, and the advanced features include: weight initialization, batch normalization, and see their impact on the results. The use of each feature varies for each dataset, but all the features tested in this research enhanced the network's performance. However, drop-out ratio and batch normalization are found to be alternate of each other; therefore, using both of them together does not show a variation in the accuracy of an individual. With the optimization of various parameters, the MLP network performed better. However, due to the poor size of the data set, both the university help desk and Ask Teacher Dataset had lesser accuracy values ranging from 72-74% compared to Enron Intent Dataset, which achieved the maximum accuracy of 91.18%.

Marza et al., in their work [21], suggested that the Min-hash technique is combined with the Deep Neural Network (DNN) algorithm to classify emails into Spam and Ham. The dataset in their study has 5725 instances. Their proposed methodology consists of several steps: After the data cleaning step, calculating Hashing shingles, they used K- shingles with (k=3) length. The characteristics matrix and signature are implemented to generate a dense matrix from the sizeable sparse matrix. The characteristic matrix was dense using the (h=4) Min-hash function. Then the values of the min-hash are used to build the signature matrix. These values feed to the deep neural network as input vectors. After the k- shingle stage, the Hash functions (crc32) are used. As for the k-hashed tokens, the Min-hash algorithm is used. Following they set the number of hidden layers, nodes on each layer, training batches, and the number of training data for each batch in a DNN with several hidden layers. Then the DNN Spam classifier is used to classify the checked emails. As compared to alternative works, it has been observed that the proposed method is relatively more effective, as the accuracy rate obtained was remarkably high (98%). Their findings show that the signature matrix is adequate for this mission, emphasizing tempo, secrecy, and honesty.

IV. Methodology

A. Model framework

The general workflow of the research work for classifying emails into four different classes is shown below in Fig3.

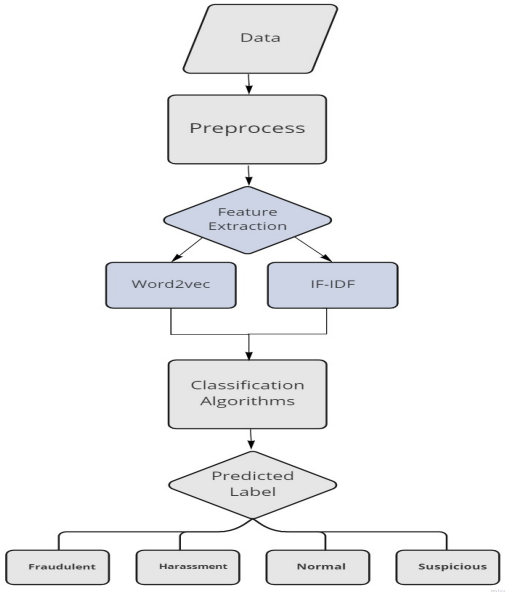

Fig. 3. Proposed model for E-mail classification.

B. Data Description

TABLE I
Composition of SeFACED Dataset.

E-mail Type	Normal	Fraudulent	Harassment	Suspicious
Number	12498	5142	19190	5323
Percentage	29.6%	12.19%	45.5%	12.6%

In this research project, the SEFACED Email Forensic Dataset [18] was used for model development in machine

learning. The dataset is a combination of four different datasets. First, it contains **normal** emails and **fraudulent** emails containing misleading information to get important information, and second, **harassing** messages , used for cyber-bullying and threatening people. Finally, the **suspicious** dataset contains some terrorism-related messages that include some texts about unlawful activities. These different datasets are merged to provide a multi-class email classification. Table I shows the instance number for each class in the dataset. The total instances number is 42,153. only the email content is used; the header information is removed.

C. Environment Setup

A) Python 3.8 is a new version of the Python programming language. It is a highly recommended programming language for AI-based projects, and the new generation's programmers love it because it is straightforward to learn, understand and use.

B) Google CoLab is an open-source distributor of the Python programming language that is entirely free to use. Users can work online through the browser and the Jupiter notebook.

D. Data Pre-Processing

Preprocessing of data is essential before applying a ML algorithm because not all information in an email is useful for fraud classification. Removing particular noisy and less informative terms can improve the performance of a classifier and reduce the dimensionality of the feature space in most cases. The following are the steps used in preparing text data, consider the following email:(Hallo! We plan to design a translated version of the application).

- Removing the Special Characters: Numbers, punctuation, and special characters must be deleted in all text, such as (!, =, ¿). Furthermore, punctuation involves a full stop, comma, brackets for separating sentences and clearing up meaning. For punctuation removal, we use the "NLTK" library. Thus, the text would become (hallo we plan to design a translated version of the application).
- Stop Words Removal: Words like (is, it, and, the) occur very frequently and are not relevant to the context of the email; it creates noise in the text data and does not add much meaning to a sentence. Therefore, it is called stop words and should be removed from the text to be processed, using the "NLTK". After removing it, the provided email becomes: (plan design translated version application).
- Tokenization: Decomposition of the original text into its constituent parts is the step of tokenization in natural language processing. There are predefined rules for tokenizing documents into words. After tokenization, the provided email is converted to a list of words such as ("plan", "design", "translated", "version", "application").
- Lemmatization: Generally, lemmatization proposes eliminating the inflectional endings of the phrase and returning the lemma, which is the base or dictionary form of an expression. Lemmatization output is a proper term. The text of the email after lemmatizing, are like ("plan", "design", "translate", "version", "application").

E. Feature Extraction

Feature extraction is a phase in which raw data is converted into valuable knowledge by reformatting, merging, and transforming basic features into new ones. For this research, two feature extraction techniques are used :

- Term frequency-inverse document frequency: Term Frequency is the number of occurrences of a word in a document, each frequency logarithmically scaled by the ratio of a total number of documents and the number of documents containing the term (IDF) [22]. It is used for information retrieval. It gives information about how important a word is in a document or corpus. To weight the terminology word that occurs less frequently in a document than the stop word by calculating the inverse document frequency, which is given by the following mathematical equation 1.

$$idf\,(t, D) = log \frac{N}{|\{d \in D : t \in d\}\,|} \tag{1}$$

  Where N is the total number of documents in the corpus. Now TF-IDF can be calculated as 2 equation.

$$tfidf\,(t, d, D) = tf\,(t, d) * \ idf\,(t, D) \tag{2}$$

- Word2vec: Word2vec (W2V) was first introduced by the Google research team, with the idea of aggregating related models to produce word embedding [23]. W2V algorithm introduced new ideas that had a considerable impact on natural language processing [24]. In natural language processing (NLP), word embeddings are the joint names for language modeling and learning techniques in which words or phrases are expressed as absolute numbers vectors. The Word Embeddings models are various, W2V model is one of them. W2V expresses words in vectors based on multiple features such as the window size, and dimensionality of vector [13]. Rationally, Word Embeddings includes mathematical formulas. The equation 3 below shows how W2V manages word context using probability measures [18]. *D* expresses the pairwise mapping of a set of words, (w,c) is the pair of word-context drawn from the large set *D*.

$$P\,(D = 1|\ w, c_{1:k}) = \frac{1}{1 + e^{-(w.c_1 + w.c_2 + ... + w.c_k)}} \tag{3}$$

F. Modelling

Classifying is a manner of data analysis that can extract models that describe essential classes of data [25].Moreover, the classifier or model is built for predicting the class label. Four supervised classification algorithms were selected to train and test the accuracy of email classification with the grouped features.

- Naïve Bayes: The NB algorithm is a straightforward conventional probabilistic classifier that makes computation for a collection of probabilities. It counts the frequency and combination of values in a given data set [26]. This classifier can compute the best possible output according to the input. Further, it can assume that the existence or nonexistence of a particular feature of a class is not related to the existence or nonexistence of another feature, given the class variable [27].NB algorithm is as follows:

$$P(c/x) = \frac{P(c) * P(x/c)}{P(x)} \tag{4}$$

  - P(class(c)/features(x)) : Posterior Probability
  - P(class(c)) : Class Prior Probability
  - P(features(x)/class(c)) : Likelihood
  - P(features) : Predictor Prior Probability

  Naïve Bayes algorithm provides the means to compute the posterior probability P(c—x) from P(c), P(x), and P(x—c). Thus, in this classifier, the impact of the predictor (x) value on a specific class (c) is separated from other predictor's values.
- Support Vector Machine: SVM generally utilized for prediction and classification problems. Moreover, it has been widely applied in many applications for solving real-world problems [28]. For a simple binary high-dimensional linear classification problem, this algorithm creates a hyperplane that reduces the distance between the hyperplane and the nearest data points on each side [29]. As a result, SVM provides significantly greater search accuracy, as experiments have shown. SVM uses the following decision rule 5 :

$$h\,(\,\vec{x}\,) = sign\left(\sum_{j}\; a_j \;\; y_j \;\; (\;\vec{x}\;\; .\,\vec{x}_j\;) - b\right) \tag{5}$$

- Logistic Regression: This is a classification algorithm used to divide the data into different classes. These can be normal, ordinary, and multinomial. For example, in binary logistic regression, the result or classification can be divided into 0's and 1's, while in multinominal regression, the result or classification can be done in multiple ways [29]. The Sigmoid function is the activation function applied for this purpose. The mathematical representation of the sigmoid activation function is as follows 6:

$$\sigma(x) = \frac{1}{1 + exp(-w^T x)} \tag{6}$$

- Random Forest: The Random Forest (RF) is a hierarchical collection of tree-structured base classifiers. Text data usually has a large number of dimensions. The dataset contains a large number of irrelevant attributes. Breiman formulated the RF algorithm using sample data sets and to construct multiple decision trees by mapping a random sample of feature subspaces [30]. The RF algorithm can be described as follows in conjunction with a set of training documents D and N_f features:
  - Initial: D1, D2, ... D_k are selected with predetermined probability with replacement.
  - For each document D_k construct a decision tree model. The training documents are randomly selected from the available features using the subspace of dimension m-try. Calculate all possible probabilities based on the m-try features. The leaf node generates the best data partition. The process continues until it reaches the saturation criterion.

  Combine the K unpruned trees $h_1(X_1)$, $h_2(X_2)$ ..... into a random forest ensemble and use the high probability value for the classification decision.

G. Evaluation Measures

In this research, various measurements are used to evaluate the performance of classifiers, such as accuracy, precision, recognition, and f-score. These measurements are calculated using a confusion matrix which consists of four terms.

- True positive (TP): is when the predicted class is positive values and correctly classified as positive.
- True negative (TN): is when the predicted class is negative values that are correctly classified as negative.
- False Positive (FP): is when the predicted class is negative values and incorrectly classified as positive.
- False Negative (FN): is when the predicted class is positive values and incorrectly classified as negative.

For the performance evaluation of our proposed model, we use the following metrics.

A) Accuracy

Accuracy metric reveals how many instances were correctly classified made by the algorethm out of the entire classified dataest. The accuracy of a recognition mechanism can be calculated using the equation 7

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

B) Precision

Precision reveals the number of correct positives (TP) predictions divided by the correct and incorrect predictions of positives, so if the model predicts positives, the precision assures that instance is labeled correctly as positive. Thus, a high precision value indicates that the algorithm has provided a relevant result and can be calculated using the equation 8.

$$Precision = \frac{TP}{TP + FP} \tag{8}$$

C) Recall

Precision gives the number of correct positive predictions (TP) divided by the sum of the correct prediction of positives (TP) and the incorrect prediction of negatives (FN), so if the model predicts positives, the precision assures that the instance is labeled correctly as positive. Thus, a high precision value indicates that the algorithm has provided a relevant result. The recall is valid when

the cost of false negatives is high. It can be calculated as follows. Equation 9

$$Recall = \frac{TP}{TP + FN} \tag{9}$$

D) F-Measure
The combination of Precision and Recall calculates F-Measure to evaluate the overall accuracy of the algorithm. Thus, a low score for false positives and false negatives indicates a good model that accurately predicted the outcome. The following formula 10 can calculate the score of F-measure.

$$F1 = 2 * \frac{TP}{TP + FP + FN} \tag{10}$$

V. Results

Using the library of Scikit-learn, we utilize the Term Frequency-Inverse Document Frequency (TF-IDF). It was applied to Random Forest, Multinomial Naive Bayes Support Vector Machine, and Logistic Regression. Moreover, Using the Gensim Models, Word2Vec was applied to Random Forest, Gaussian Naïve Bayes Support Vector Machine, and Logistic Regression.

A. Support Vector Machine Result

SVMs are integrated with kernel functions to adapt to nonlinearities in data [31]. We used the Radial Basis Function (RBF) for the SVM model from different types of kernels available. The SVM with TFIDF achieved 95% accuracy. The SVM with W2v achieved 89% accuracy

B. Logistic Regression Result

From the Scikit-learn, we used the logistic regression for multi-class classification using built-in one-vs-rest, which enables a multi-class approach to be used with any binary classifier such as logistic regression [32]. Logistic Regression with TF-IDF achieved 95% accuracy. Logistic Regression with W2V achieved 87% accuracy.

C. Random Forest Result

For Random Forest, the grid search was used for hyperparameters optimization, which aids in finding the most suitable combination. In RF, hyperparameters are n_estimators and max_features. Table II shows the achieved results. The accuracy score for the Random Forest classifier with TF-IDF was about 95%. And the accuracy score for the Random Forest classifier with W2V was about 91%.

D. Naïve Bayes

Naive Bayes is a group of supervised learning algorithms based on the principle of the Bayes theorem. This theorem works on conditional probability by calculating the probability of the events. Binary and multiple classifications are done using different algorithms like GaussianNB, MultinomialNB, BernoulliNB . Multinomial Naive Bayes is a specific instance of naïve Bayes classifier which uses multinomial distribution for each of its features [33]. It was used based on TF-IDF ,and its overall accuracy was 93%. On the other hand, Gaussian Naïve Bayes was used based on W2V. GNB achieved 70% accuracy. Detailed of the result is shown in Table II.

TABLE II
SUMMARY OF EXPERIMENTAL RESULTS

Classifier	Feature Extraction	Acc%	Pre%	Rec %	F1 %
SVM	TF-IDF	**0.95**	0.95	0.95	0.95
LR		**0.95**	0.95	0.95	0.95
RF		**0.95**	0.95	0.95	0.95
MNB		0.93	0.93	0.93	0.92
SVM	Word2Vec	0.89	0.90	0.89	0.89
LR		0.87	0.87	0.87	0.87
RF		0.91	0.91	0.91	0.90
GNB		0.70	0.71	0.70	0.69

VI. Conclusion And Future Work

The number of e-mail users all over the world is growing every year [34]. As a result, many organizations share their necessary info utilizing e-mail like delivering a document, sharing messages, collaborations, essential updates, and notifications. Unfortunately, e-mail has generated one of the primary forms of cybercrime-fraud, or unsolicited advertisements for products and services, which experts estimate to account for about 50 % of e-mails spreading on the Internet. To investigate crimes committed by e-mails, we need to utilize DF. The DF investigation involves the examination of digital evidence of the crime committed, which can be used as evidence in court. In this process, ML can be viewed as an optimal approach to solve the problems that exist in the DF field.

We have proposed a content base, E-mail multi-classification based on ML approaches in this research. Using the Sefaced dataset, e-mail is classified into four classes, Normal and three different categories of illegal e-mails to be detected. We have explored four supervised ML methods, namely Naive Bayes, Logistic regression, Random Forest, and Support Vector Machine. In addition, two feature extraction techniques (Word2vec, TF-IDF) were used to find out which works best for the e-mail classification process. Our model shows significant accuracy in almost all the supervised ML algorithms with the TF-IDF. Moreover, Random Forest, Logistic Regression, and Support Vector Machine show the highest accuracy with 95%. Additionally, we compared our result with [18], and we managed to achieve the same result which they achieved with their novel approach named SeFACED. Additionally, the result of the experiment based on TF-IDF is much better than the Word2vec result. This indicates that TF-IDF is a much better choice for a text e-mail multi-classification.

The research experiment can be extended by working on developing an algorithm and technique that can handle imbalanced data much better. Moreover, more ML techniques can be applied to discover a better result. On the other hand, in the future, we can apply Deep Learning (DL) approaches such as Deep Neural Networks on the text email multi-classification.

REFERENCES

[1] S. Iqbal and S. A. Alharbi, "Advancing automation in digital forensic investigations using machine learning forensics," *Digital Forensic Science*, p. 3, 2020.

[2] P. g. C. o. E. f. m. r. PwC Research and insight, "AI Predictions 2021," 2021. [Online]. Available: https://www.pwc.com/us/en/tech-effect/ai-analytics/ai-predictions.html

[3] P. Bhatt and P. H. Rughani, "Machine learning forensics: A new branch of digital forensics." *International Journal of Advanced Research in Computer Science*, vol. 8, no. 8, 2017.

[4] K. Rajendiran, K. Kannan, and Y. Yu, "Applications of machine learning in cyber forensics," in *Confluence of AI, Machine, and Deep Learning in Cyber Forensics*. IGI Global, 2021, pp. 29–46.

[5] K. U. Maheswari and S. N. Bushra, "Machine learning forensics to gauge the likelihood of fraud in emails," in *2021 6th International Conference on Communication and Electronics Systems (ICCES)*. IEEE, 2021, pp. 1567–1572.

[6] B. Almaslukh, "Forensic analysis using text clustering in the age of large volume data: A review," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 6, 2019.

[7] H. Khan, S. Hanif, and B. Muhammad, "A survey of machine learning applications in digital forensics," *Trends in Computer Science and Information Technology*, vol. 6, no. 1, pp. 020–024, 2021.

[8] I. Goni, J. M. Gumpy, T. U. Maigari, M. Muhammad, and A. Saidu, "Cybersecurity and cyber forensics: Machine learning approach," *Machine Learning Research*, vol. 5, no. 4, pp. 46–50, 2020.

[9] A. Årnes, *Digital forensics*. John Wiley & Sons, 2017.

[10] P. V. Kayarkar, P. Ricchariaya, and A. Motwani, "Mining frequent sequences for emails in cyber forensics investigation," *International Journal of Computer Applications*, vol. 85, no. 17, 2014.

[11] R. Al-Haddad, F. Sahwan, A. Aboalmakarem, G. Latif, and Y. M. Alufaisan, "Email text analysis for fraud detection through machine learning techniques," in *3rd Smart Cities Symposium (SCS 2020)*, vol. 2020, 2020, pp. 613–616.

[12] A. Bibi, R. Latif, S. Khalid, W. Ahmed, R. A. Shabir, and T. Shahryar, "Spam mail scanning using machine learning algorithm." *J. Comput.*, vol. 15, no. 2, pp. 73–84, 2020.

[13] N. Jahan Euna and M. S. I. Hossain, S.M.M.and Anwar, "Content-based spam email detection using n-gram machine learning approach," *Preprints*, 2021.

[14] N. Kumar, S. Sonowal *et al.*, "Email spam detection using machine learning algorithms," in *2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA)*. IEEE, 2020, pp. 108–113.

[15] M. Husnain, A. Khalid, and N. Shafi, "A novel preprocessing technique for toxic comment classification," in *2021 International Conference on Artificial Intelligence (ICAI)*. IEEE, 2021, pp. 22–27.

[16] R. Tahsin, M. H. Mozumder, S. A. Shahriyar, and M. A. S. Mollah, "A novel approach for e-mail classification using fasttext," in *2020 IEEE region 10 symposium (TENSYMP)*. IEEE, 2020, pp. 1392–1395.

[17] S. Bagui, D. Nandi, S. Bagui, and R. J. White, "Classifying phishing email using machine learning and deep learning," in *2019 International Conference on Cyber Security and Protection of Digital Services (Cyber Security)*. IEEE, 2019, pp. 1–2.

[18] M. Hina, M. Ali, A. R. Javed, F. Ghabban, L. A. Khan, and Z. Jalil, "Sefaced: Semantic-based forensic analysis and classification of e-mail data using deep learning," *IEEE Access*, vol. 9, pp. 98 398–98 411, 2021.

[19] S. Kaddoura, O. Alfandi, and N. Dahmani, "A spam email detection mechanism for english language text emails using deep learning approach," in *2020 IEEE 29th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)*. IEEE, 2020, pp. 193–198.

[20] A. Altaf, A. Mehmood, and M. Khan, "Email organization through deep learning algorithms," in *2020 14th International Conference on Open Source Systems and Technologies (ICOSST)*. IEEE, 2020, pp. 1–5.

[21] N. H. Marza, M. E. Manaa, and H. A. Lafta, "Classification of spam emails using deep learning," in *2021 1st Babylon International Conference on Information Technology and Science (BICITS)*. IEEE, 2021, pp. 63–68.

[22] N. Pattaniyil and R. Zanibbi, "Combining tf-idf text retrieval with an inverted index over symbol pairs in math expressions: The tangent math search engine at ntcir 2014." in *NTCIR*, 2014.

[23] H. H. Mohammed, E. Dogdu, A. K. Görür, and R. Choupani, "Multi-label classification of text documents using deep learning," in *2020 IEEE International Conference on Big Data (Big Data)*. IEEE, 2020, pp. 4681–4689.

[24] M. Grohe, "word2vec, node2vec, graph2vec, x2vec: Towards a theory of vector embeddings of structured data," in *Proceedings of the 39th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, 2020, pp. 1–16.

[25] R. Bhuvaneswari and K. Kalaiselvi, "Naive bayesian classification approach in healthcare applications," *International Journal of Computer Science and Telecommunications*, vol. 3, no. 1, pp. 106–112, 2012.

[26] M. Jaiswal, S. Das *et al.*, "Detecting spam e-mails using stop word tf-idf and stemming algorithm with naïve bayes classifier on the multicore gpu." *International Journal of Electrical & Computer Engineering (2088-8708)*, vol. 11, no. 4, 2021.

[27] S. S. Nikam, "A comparative study of classification techniques in data mining algorithms," *Oriental journal of computer science & technology*, vol. 8, no. 1, pp. 13–19, 2015.

[28] A. Ben-Hur, C. S. Ong, S. Sonnenburg, B. Schölkopf, and G. Rätsch, "Support vector machines and kernels for computational biology," *PLoS computational biology*, vol. 4, no. 10, p. e1000173, 2008.

[29] N. A. Unnithan, N. Harikrishnan, S. Akarsh, R. Vinayakumar, and K. Soman, "Machine learning based phishing e-mail detection," *Security-CEN@ Amrita*, pp. 65–69, 2018.

[30] V. Jackins, S. Vimal, M. Kaliappan, and M. Y. Lee, "Ai-based smart prediction of clinical disease using random forest classifier and naive bayes," *The Journal of Supercomputing*, vol. 77, no. 5, pp. 5198–5219, 2021.

[31] B. K. Dedeturk and B. Akay, "Spam filtering using a logistic regression model trained by an artificial bee colony algorithm," *Applied Soft Computing*, vol. 91, p. 106229, 2020.

[32] D. Cauchi and A. Muscat, "One vs previous and similar classes learning–a comparative study," *arXiv preprint arXiv:2101.01294*, 2021.

[33] Y. Singhal, Y. Varshney, A. Goyal, and N. Aggrawal, "Multiclass classification approachsms categorization," *International Journal of Advanced Studies of Scientific Research*, vol. 3, no. 10, 2018.

[34] statista. Number of e-mail users worldwide from 2017 to 2025. [Online]. Available: https://www.statista.com/statistics/255080/number-of-e-mail-users-worldwide/