****

**ENHANCING EMAIL SPAM DETECTION THROUGH ENSEMBLE MACHINE LEARNING:**

**A COMPREHENSIVE EVALUATION OF MODEL INTEGRATION AND PERFORMANCE**

* Hazem Mohamed Mahmoud Hassan
* Ziad Ahmed Hassan Hasaneen
* Abdullah badry Mohamed
* Hassan Alaa Hafez
* Seif El-din Osama Hosni

**Abstract**

Email spam detection and filtering are essential security measures for organizations, aimed at identifying and blocking unsolicited messages, many of which are harmful. Machine learning classification algorithms are commonly used to distinguish between spam and non-spam emails by training models on labelled data. However, traditional algorithms have proven less effective against rapidly evolving spam techniques.

This research introduces ensemble techniques using a meta-learning approach to improve spam detection by reducing misclassification and enhancing model performance. By combining multiple algorithms, this method minimizes false positives and negatives while increasing accuracy. The proposed approach aggregates predictions from diverse models to strengthen spam detection systems.

Four machine learning algorithms were selected for the meta-learning model due to their effectiveness in spam detection: Naïve Bayes (NB), Support Vector Machine (SVM), Decision Tree (DT), K-Nearest Neighbour (KNN), XGBoost, Random Forest (RF) and Artificial Neural Networks (ANN). These algorithms were individually applied to datasets, followed by the creation of an ensemble model using the stacking method. The predictions were aggregated and used as input features for the final classifier, Logistic Regression.

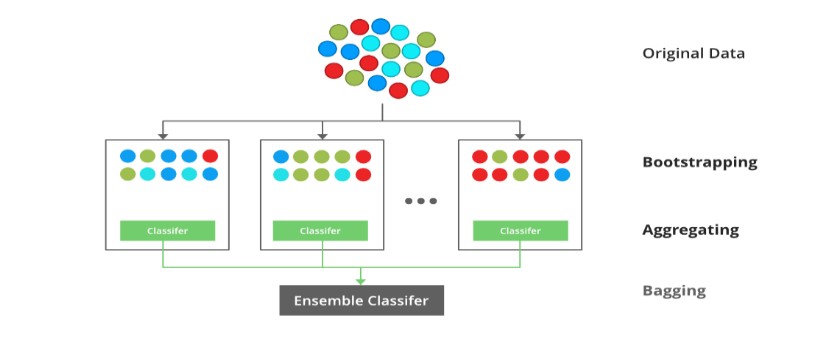
**Keywords**: *Email Spam Detection, Ensemble Machine Learning algorithms, Meta-learning* *algorithms, Classification Algorithms.*

**INTRODUCTION AND RELATED WORKS**

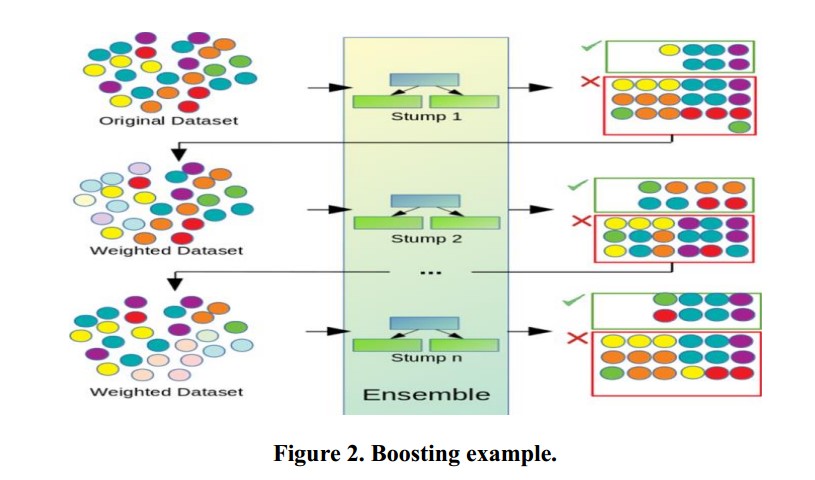
Email remains a crucial tool for communication in both personal and professional settings. However, the prevalence of spam emails presents considerable challenges, including productivity losses and potential security risks. Traditional approaches to spam detection, which rely on single machine learning models, often fall short in addressing the evolving and complex nature of spam. This study explores an ensemble-based approach that combines multiple machine learning algorithms to enhance the efficiency of spam detection.

In recent years, significant progress has been made in the field of email spam detection. Previous research has extensively examined various machine learning algorithms for identifying spam. Commonly used models, such as Naive Bayes, Support Vector Machines (SVM), and Decision Trees, are valued for their simplicity and effectiveness. However, these models have limitations, including sensitivity to feature selection and a tendency to overfit data.

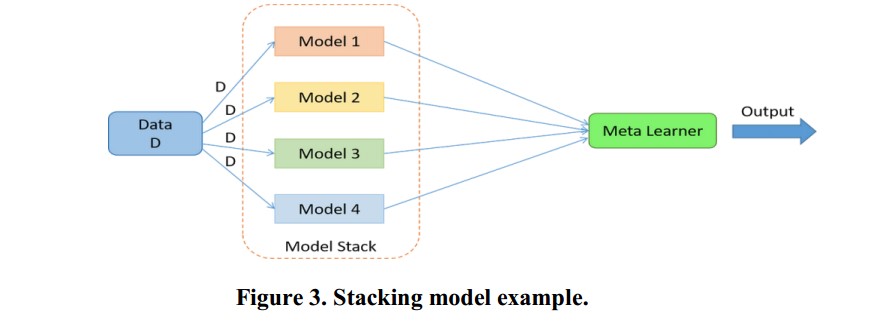
Recent advancements in machine learning have shown that ensemble methods, which integrate multiple models, can significantly improve predictive accuracy and robustness. Over the last five years, ensemble techniques have garnered considerable attention in email spam detection. Researchers have focused on combining several machine learning algorithms to enhance the accuracy and reliability of these systems. This section highlights key contributions in this area, emphasizing the effectiveness of various ensemble techniques.

**Bagging and boosting are among the most widely used ensemble methods in spam detection. Bagging enhances the stability and accuracy of machine learning models by training multiple models on random subsets of the dataset and aggregating their predictions. Bagging relies on creating diverse training datasets through sampling with replacement, and the final prediction is typically based on the average or majority vote of the individual models.

**Figure 1. Bagging example**

**On the other hand, Boosting models are based on sequentially trained models, each model corrects the errors of its predecessor; in boosting models each new model attempts to correct the errors made by the previous models. The final prediction is a weighted sum of the predictions from all models. AdaBoost, Gradient Boosting m XGBoost algorithms all are based on Boosting method; which adjusts the weights of incorrectly classified instances so that subsequent models focus more on rare cases (difficult cases) than the majority cases.

Combining Decision Trees, Naive Bayes, and SVM through bagging and boosting significantly enhances detection accuracy and reduces false positives, achieving higher F1 scores than individual models. *The stacking* method is one ensemble method, that involves training multiple classifiers to make the predictions in the first stage and then using their predictions as inputs for a meta-classifier, which makes an accurate final decision.

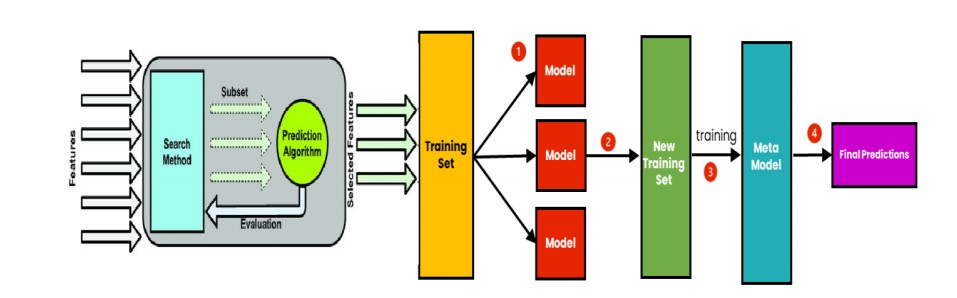
** The stacking method uses multiple models on the same training dataset, then using another model a meta-learner to combine and aggregate their predictions. The meta-learner learns how to best combine the predictions from the models to make an accurate final prediction, as shown in Figure 3.

**PROPOSED SYSTEM METHODOLOGY**

This paper proposes an ensemble approach combining multiple machine learning algorithms to enhance the accuracy and robustness of spam detection systems. This paper mainly focuses on developing an ensemble model to classify emails as spam or not spam using the python programming language. Python, known for its powerful statistical and graphical capabilities, provides an excellent platform for data analysis and machine learning tasks. Figure 4 illustrates the steps of the following methodology.

***i. Feature selection by using Wrapper methods***

Feature selection is an essential step in machine learning, especially when dealing with noisy datasets. In this research, the dataset contains 58 features that represent the frequency of words or characters. In particular, the feature selection process is used to improve the model's performance by eliminating irrelevant or redundant features. In this research, the feature selection techniques based on using wrapper methods have been used. However, wrapper methods ranked the most relevant features, which are based on statistical tests and some statistical criteria. In fact, dealing with ensemble methods, a wrapper method is the best strategy can be used to improve model performance by selecting the best subset of features. Two filters have been used in this research to filter the data and get the most relevant features for the prediction stage which include:

 *Correlation Analysis filter*; which is used to keep the highly correlated features (e.g., using the cor function); used to avoid redundant information, and keep the only features which have the most correlation with the class label (Spam/Not Spam).

***ii. Training stage by using ensemble stack methods***

After the feature selection stage, the selected features will be used to train the ensemble model, which is based on the stacking method that has been described in the previous section(i). During the training stage, all the predictions of the selected models will be trained and then aggregated and used as input features for the base classifier that is based on the Logistic Regression algorithm. In this study, the following machine learning algorithms were selected based on their proven effectiveness in spam detection:  
• *Naive Bayes (NB)*: A probabilistic classifier based on Bayes' theorem, suitable for datasets with different characters, and this algorithm is useful to predict the correct class label (Spam/Not Spam) in unbalanced datasets.  
• *Support Vector Machine (SVM)*: A strong classifier that performs well in datasets that are not linearly separable, like the one employed in this study.  
• *Decision Tree (DT)*: This algorithm has been chosen as one of the learners in this research as this algorithm is used to classify the dataset based on the class label (Spam/Not-Spam).  
• *K-Nearest Neighbors (KNN)*: A non-parametric method used for classification by measuring the distance between the test data and all training samples.

 XGBoost **(**Extreme Gradient Boosting**)**: An optimized gradient-boosting algorithm that builds a series of decision trees sequentially, enhancing model accuracy and robustness. XGBoost is known for its scalability and superior performance in a wide range of machine learning tasks.

 Random Forest (RF): An ensemble learning method that constructs multiple decision trees during training and outputs the majority vote (classification) or average prediction (regression) of the trees, providing strong accuracy and resistance to overfitting.

Artificial Neural Networks (ANN): a Deep learning model inspired by the human brain. Uses layers of neurons to learn representations and patterns in data. Suitable for complex and large-scale datasets. Effective for classification tasks with intricate feature interactions.

Gradient Boosting: Boosting ensemble method. Combines weak learners (e.g., decision trees) iteratively to minimize errors. Reduces bias and variance in predictions. Performs well in tasks requiring high predictive accuracy.

***iii. The evaluation stage of the proposed model***

The 10-fold cross-validation method is used to train and test the proposed model. In the evaluation stage, the predicted data (which is collected from the four modules NB, SVM, DR, and KNN) is split into 10-folds; as 9-folds for the training date and one fold for the testing date, the base classifier (LLR algorithm) is used during the evaluation stage. Five common metrics were employed in order to assess the suggested ensemble models: recall,  
***accuracy***, ***sensitivity***, ***specificity***, and ***precision*** as shown in Table 2. The computation of these five measures are based on the number of FP, FN, TP, and TN, the definition of these numbers are displayed in Table 1.

• True Positive (TP) refers to a sample which is from the positive class(Spam), being correctly classified by the classification mode as (Spam).  
• False Positive (FP) refers to a sample which is from negative class (Not-Spam), being incorrectly classified as belonging to the positive class (Spam).  
• True Negative (TN) refers to a sample which is from the negative class(Not-Spam), being correctly classified by the classification model(Not-Spam).  
• False Negative (FN) refers to a sample which is from the positive class(Spam), being incorrectly classified as negative class by the model(Not-Spam).



**Table 1. Confusion matrix**

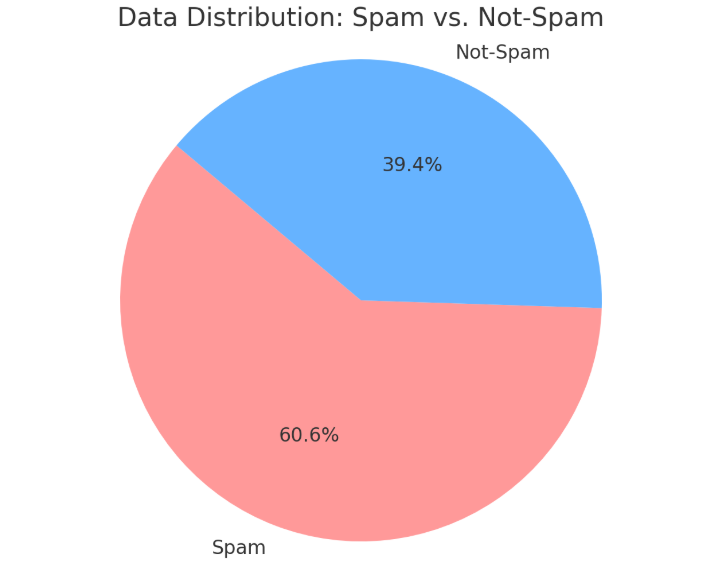
**Table 2. Evaluation measures**

|  |  |  |
| --- | --- | --- |
| calculation | Definition | term |
| TP/ (TP + FN) | Ability to select what need to be selected | **sensitivity** |
| TN/ (TN + FP) | Ability to reject what need to be rejected | **specificity** |
| TP/ (TP + FP) | Proportion of cases found that were relevant | **precision** |
| TP (TP + FN) | Proportion of all relevant cases that were found | **Recall** |
| (TP + TN)/ (TP+TN+FP+FN) | Aggregate measure of classifier performance | **accuracy** |

The selected algorithms were first trained individually on the datasets using based models NB, xgboost, RF, LR, SVM, DT, Gradient Boost, ANN, and KNN. Subsequently, an ensemble model was created using stacking, where the predictions of the base models were used as input features for a meta-classifier Logistic Regression (LR) model. Performance metrics such as accuracy, precision, recall, and F1-score were used to evaluate the models. Also, the 10-fold-cross validation method has been used to evaluate the ensemble model on the dataset and ensure it performs better than individual models, all the experimental results and conclusion will be discussed in the next section.

**Results and Discussion**

observed that the SPAM level contained 2,788 instances, while the Not-SPAM level included 1,813 instances. During preprocessing, the data distribution was adjusted by dividing the frequency of each category by the total number of instances and rounding the results to two decimal places. The new data distribution showed Spam = 60.6% and Not-Spam = 39.4%. To identify the most relevant features for class labels (Spam and Not-Spam), the Wrapper method was applied, reducing the number of features from 58 to 32.

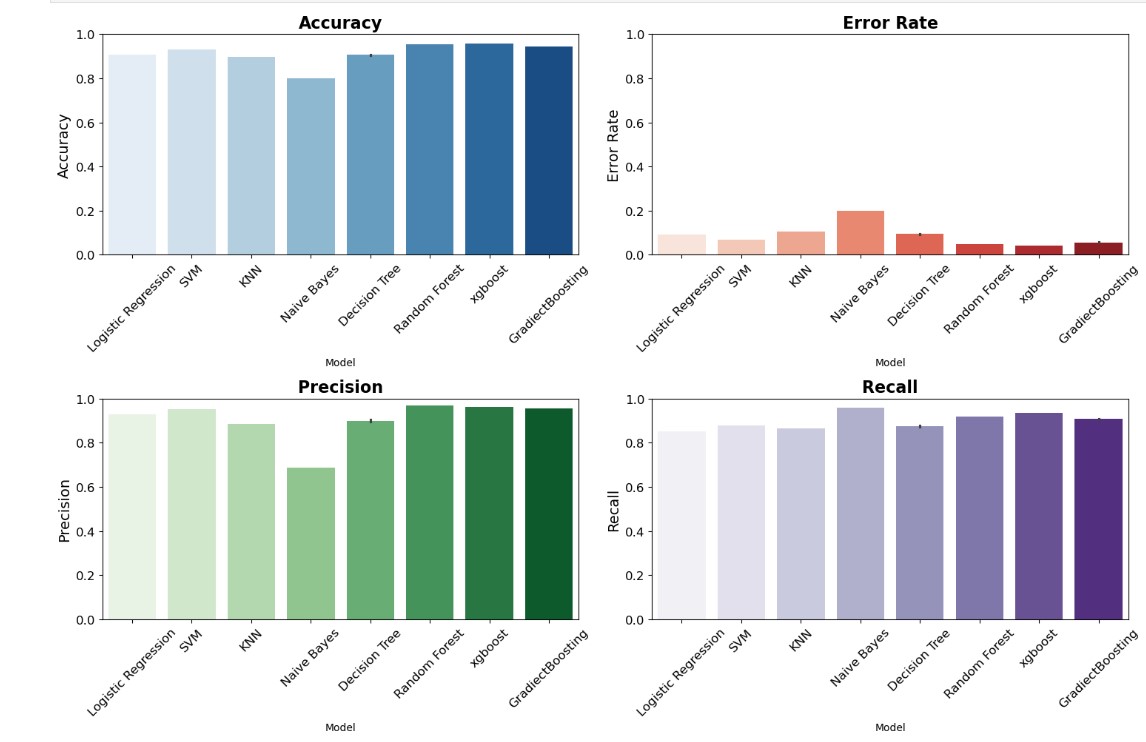


This was achieved by training the model with each feature individually and adding features that produced the most significant improvement in model performance, continuing until a stopping criterion was met or no further enhancement was observed

Following data cleansing and feature selection, the ensemble stacking method was employed. Predictions from the selected models, detailed in Section 2.ii, were aggregated, and used as input features for the base classifier. The base classifier, implemented using the Logistic Regression algorithm, is illustrated in Figure 4. Table 3 summarizes the individual model performances on the test datasets. While all models demonstrated acceptable accuracy, there were notable variations in precision and recall, reflecting differences in their abilities to handle false positives and false negatives.

**Results Summary**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Gradient Boosting** | **ANN** | **Random Forest** | **XgBoost** | **Naïve Bayes** | **SVM** | **KNN** | **Decision Tree** | **Model** |
| **94%** | **94%** | **95%** | **96%** | **80%** | **93%** | **90%** | **91%** | **Accuracy** |
| **97%** | **98%** | **98%** | **96%** | **90%** | **97%** | **92%** | **93%** | **Recall** |
| **93%** | **93%** | **94%** | **95%** | **96%** | **92%** | **90%** | **91%** | **Percision** |
| **95%** | **95%** | **96%** | **96%** | **80%** | **94%** | **91%** | **92%** | **F1\_score** |

The ensemble model achieved the highest accuracy and balanced performance criteria, outperforming the individual models. This confirms the hypothesis that combining multiple models can mitigate individual weaknesses and leverage their strengths.



**Ensemble Model Results**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| F1\_score | Recall | Percision | Accuracy | Model |
| 96% | 94% | 97% | 96% | Ensemble Model |

Error analysis revealed that the ensemble model significantly reduced false positives and false negatives compared to individual models. This improvement is attributed to the complementary nature of the base models' predictions

**Conclusion**

This study highlights the effectiveness of an ensemble approach for email spam detection by integrating multiple machine learning algorithms to improve accuracy and robustness. The dataset used contains 4601 instances and 58 variables, reduced to 32 features using the wrapper method. This feature selection technique reduces dimensionality, enhances model performance, minimizes overfitting, and simplifies interpretation.

Experiments demonstrate that the ensemble model outperforms individual base models by leveraging the strengths of multiple algorithms. classifiers (LR, SVM, NB, KNN, ANN, XGBoost, ANN and DT) were used, with final predictions made by an LR classifier. The results showed a high accuracy of 96% with 4% classification errors

**REFERENCES**

Asuncion, A., and Newman, D. (2007). UCI machine learning repository. Available at:  
http://archive.ics.uci.edu/ml/ (last accessed Sep/2024) Spambase - UCI Machine Learning  
Repository

Cormack, G. V. (2007). Email spam filtering: A systematic review. Foundations and Trends® in Information Retrieval, 1(4), 335-455.

De Sousa, M. N., Sant’Ana, R., Fernandes, R. P., Duarte, J. C., Apolinário Jr, J. A., & Thomä, R. S. (2021). Improving the performance of a radio-frequency localization system in adverse outdoor applications. *EURASIP Journal on Wireless Communications and Networking*, *2021*(1), 123.

Dietterich, T. G. (2000). Ensemble methods in machine learning. In International workshop on multiple classifier systems (pp. 1-15). Springer, Berlin, Heidelberg.

G. Blanchard, "Spam Email Detection Using a Multi-classifier Approach," IEEE Transactions on Information Forensics and Security, vol. 17, no. 1, pp. 123-134, Jan. 2022.

Ghosh, S., Basu, S., & Gupta, A. (2020). Enhancing Spam Detection Using Bagging and Boosting. Journal of Machine Learning Research, 21(101), 1-20.

Ghourabi, A., & Alohaly, M. (2023). Enhancing spam message classification and detection using transformer-based embedding and ensemble learning. *Sensors*, *23*(8), 3861.

Hoc, H. T., Silhavy, R., Prokopova, Z., & Silhavy, P. (2023). Comparing stacking ensemble and deep learning for software project effort estimation. *IEEE Access*, *11*, 60590-60604.

Jan, Z. M., & Verma, B. (2019, June). Ensemble classifier optimization by reducing input features and base classifiers. In *2019 IEEE Congress on Evolutionary Computation (CEC)* (pp. 1580-1587). IEEE.

Kuhn, M. (2013). Applied predictive modelling.

Kumar, P., & Thakur, S. (2021). Stacking-Based Ensemble Learning for Improved Spam Detection. Information Systems Frontiers, 23(3), 723-735.

Liu, T., Li, S., Dong, Y., Mo, Y., & He, S. (2024). Spam detection and classification based on distilbert deep learning algorithm. *Applied Science and Engineering Journal for Advanced Research*, *3*(3), 6-10.

Martinez, A., & Gomez, R. (2022). Online Learning for Real-time Spam Detection Using Ensemble Methods. Journal of Real-Time Processing, 38(4), 567-582.

Nashnush, E. B. (2015). *Development of new cost-sensitive Bayesian network learning algorithms*. University of Salford (United Kingdom).

Nashnush, E., & Vadera, S. (2014). Cost-sensitive Bayesian network learning using sampling. In *Recent Advances on Soft Computing and Data Mining: Proceedings of The First International Conference on Soft Computing and Data Mining (SCDM-2014) Universiti Tun Hussein Onn Malaysia, Johor, MalaysiaJune 16th-18th, 2014* (pp. 467-476). Springer International Publishing.

Nashnush, E., & Vadera, S. (2017). Learning cost-sensitive Bayesian networks via direct and indirect methods. *Integrated Computer-Aided Engineering*, *24*(1), 17-26.

Omotehinwa, T. O., & Oyewola, D. O. (2023). Hyperparameter optimization of ensemble models for spam email detection. *Applied Sciences*, *13*(3), 1971.

Pérez, C., Martínez, R., & González, P. (2020). Dimensionality Reduction in Ensemble Learning for Spam Detection. Applied Soft Computing, 89, 106078.

Qi, Q., Wang, Z., Xu, Y., Fang, Y., & Wang, C. (2023). Enhancing Phishing Email Detection through Ensemble Learning and Undersampling. *Applied Sciences*, *13*(15), 8756.

Qi, Q., Wang, Z., Xu, Y., Fang, Y., & Wang, C. (2023). Enhancing Phishing Email Detection through Ensemble Learning and Undersampling. *Applied Sciences*, *13*(15), 8756.

Rizk, Y., Hajj, N., Mitri, N., & Awad, M. (2019). Deep belief networks and cortical algorithms: A comparative study for supervised classification. *Applied computing and informatics*, *15*(2), 81-93.

Rokach, L. (2010). Ensemble-based classifiers. Artificial Intelligence Review, 33(1), 1-39.

Shajahan, R., & Lekshmy, P. L. (2022, August). Hybrid Learning Approach for E-mail Spam Detection and Classification. In *International Conference on Intelligent Cyber Physical Systems and Internet of Things* (pp. 781-794). Cham: Springer International Publishing.

Zhang, Y., Li, H., & Wang, X. (2022). Hybrid Deep Learning Models for Spam Email Detection. IEEE Transactions on Neural Networks and Learning Systems, 33(6), 2564-2575.