FEN ve MÜHENDİSLİK BİLİMLERİ DERGİSİ

Vol: 7 No: 2 Year: 2025 | Araştırma Makalesi/Research Article | e-ISSN: 2667-7989

https://doi.org/10.47112/neufmbd.2025.85

Türkçe ve İngilizce Dillerinde Spam Posta Tespiti: Bireysel, Toplu ve Hibrit Yaklaşımları İçeren Yapay Zeka Tabanlı Tekniklerin Bütünsel Bir Çalışması

Esma Nisa CANDAN[1] Rehnüma KÜÇÜKİLHAN[2] Alperen EROĞLU[1*]

[1] Necmettin Erbakan University, Faculty of Engineering, Department of Computer Engineering, Konya, Türkiye
[2] Afyon Kocatepe University, Faculty of Engineering, Department of Environmental Engineering, Afyon, Türkiye

Makale Bilgisi

Geliş Tarihi: 25.09.2024
Kabul Tarihi: 14.11.2024
Yayın Tarihi: 30.08.2025

Anahtar Kelimeler:
Hibrit Öğrenme,
İngilizce Veri Setleri,
Toplu Öğrenme,
Türkçe Veri Setleri,
Spam Mail.

ÖZET

Artan e-posta ve sosyal medya kullanımı nedeniyle spam sayısı artmış ve bu durumun sistemlere zarar vermeden etkili bir şekilde tespit edilmesi ve sınıflandırılması konusunda kritik bir zorluk oluşturmuştur. Bu makale, Türkçe ve İngilizce veri kümelerini kullanarak e-postaları spam veya ham olarak tespit etmek ve sınıflandırmak için en etkili yaklaşımları analiz etmek ve ortaya çıkarmak için bütünsel bir strateji sunmaktadır. Birleşik olarak oluşturulan yeni veri kümelerine ek olarak, farklı dillerde oluşturulan iki farklı veri kümesi kullanılmaktadır. Gelişmiş makine öğrenmesi ve derin öğrenme yaklaşımlarını temel alarak en iyi spam posta algılama yöntemlerini sunmak için karşılaştırmalı bir çalışma yapılmaktadır. Ayrıca yeni bir yaklaşım olarak spam posta tespiti için toplu ve hibrit öğrenme yöntemleri bir araya getirilmiştir. Optimize edilmiş özellik seçimi yaklaşımları ve ön işleme ile doğal dil işlemeyi ve geliştirilmiş öğrenme algoritmaları kullanılmaktadır. Literatürde yaygın olarak kullanılan Multinomial Naive Bayes, Destek Vektör Makinesi, Lojistik Regresyon, K-En Yakın Komşular, Karar Ağacı, Rastgele Orman, Oylama sınıflandırıcısı ve makine öğrenme algoritmaları olarak Yığınlama Sınıflandırıcısı ile Uzun Kısa Süreli Bellek, Çift Yönlü yöntemlerini karşılaştırmaktayız. Uzun Kısa Süreli Bellek, Transformatörlerden Çift Yönlü Kodlayıcı Gösterimleri ise derin öğrenme algoritmaları olarak kullanılmaktadır. 5 kat çapraz doğrulamaya ek olarak, veri kümeleri her model için 80:20 oranlarıyla eğitim verileri ve test verileri olarak bölünmüştür. Izgara Arama tekniği kullanılarak modellerin hiper parametreleri de optimize edilmektedir. Makine öğrenmesi yaklaşımlarına dayalı toplu öğrenme yöntemi, İngilizce Enron veri seti için %99,9 ile en iyi performansı sağlarken, basit ortalamaya dayalı hibrit toplu öğrenme yaklaşımı, UCI ve Kaggle'dan Türkçe veri seti için %98,43 ile en iyi doğruluk değerini vermektedir.

Spam Mail Detection in Turkish and English Languages: A Holistic Study of AI-based Techniques including Individual, Ensemble and Hybrid Approaches

Article Info

Received: 25.09.2024
Accepted: 14.11.2024
Published: 30.04.2025

Keywords:
English Datasets,
Ensemble Learning,
Hybrid Learning,
Turkish Datasets,
Spam Mail.

ABSTRACT

Spam has surged due to increased email and social media use, posing a critical challenge in effectively detecting and classifying this growing volume without causing harm to systems. This paper presents a holistic strategy to analyze and reveal the most efficient approaches for detecting and classifying e-mails as spam or ham by using Turkish and English datasets. We use two different datasets generated in different languages in addition to conjunctively generated new datasets. We make a comparative study to find out the best spam mail detection approaches based on our enhanced machine learning and deep learning methods. We also bring ensemble and hybrid learning methods together as a new approach for spam mail detection. We utilize natural language processing, and improved learning algorithms with optimized feature selection approaches and preprocessing. We compare various methods commonly used in the literature which are Multinomial Naive Bayes, Support Vector Machine, Logistic Regression, K-Nearest Neighbors, Decision Tree, Random Forest, Voting classifier, and Stacking classifier as machine learning algorithms, and Long Short Term Memory, Bidirectional Long Short Term Memory, Bidirectional Encoder Representations from Transformers as deep learning algorithms. We split the datasets as train data and test data with the 80:20 ratios in addition to 5-fold cross-validation for each model. We also optimize the hyperparameters of our models by using Grid Search. The ensemble method based on machine learning approaches provides the best performances which are the percentage of 99.9% for the English Enron dataset, and the hybrid ensemble approach based on simple average yields the best accuracy value of 98.43% for the Turkish dataset from UCI and Kaggle.

To cite this article:
Candan, E. N.; Küçükilhan, R. & Eroğlu, A. (2025). Spam Mail Detection in Turkish and English Languages: A Holistic Study of AI-based Techniques including Individual, Ensemble and Hybrid Approaches. *Necmettin Erbakan University Journal of Science and Engineering, 7*(2), 189-205. https://doi.org/10.47112/neufmbd.2025.85

***Corresponding Author:** Alperen Eroğlu, *aeroglu@erbakan.edu.tr*

INTRODUCTION

Electronic mail (e-mail) is increasingly favored for communication by institutions, organizations, and individuals due to its efficiency and accessibility. The number of e-mail users is rising significantly, projected to reach 7.73 billion by 2026. While e-mails serve various purposes such as business processes, document sharing, and marketing, they pose risks like information theft and system vulnerabilities [1–3]. Spam e-mails, including harmful content and unwanted advertisements, contribute to network congestion and inefficiency. Despite advancements in spam detection, they still constitute 53% of worldwide e-mail traffic. Researchers propose combining machine and deep learning algorithms into network systems to address this issue for more effective spam detection and removal.

There are two common approaches to detect whether an e-mail is legitimate or not. The first one is based on non-Artificial Intelligent (AI) techniques, software frameworks such as server authorization methods, architectural modification-based statements, collaborative models, rule-based models, and content-dependent models [4]. The latter leverages AI-based approaches including machine and deep learning techniques. In recent years, machine and deep learning approaches have been used for classifying and detecting spam or ham e-mails [5], [6]. Big companies like Google and Yahoo utilize a machine learning-based spam mail filtering approach. The content and behavior of the unsolicited mail can be changed in time. Thus, instead of static solutions, our systems can learn and decide effectively and dynamically with a higher accuracy performance. It is possible to implement spam mail filters on different sides of the network such as a firewall filter, a mail server filter, a mail transfer server filter, a mail delivery agent filter, and a client-side filter. A spam mail filter can generally be deployed on a gateway, or a router, on the applications hosted by a cloud platform, and on the user's system [6, 7].

Machine and deep learning algorithms are beneficial solutions to optimize, categorize, and classify in different areas [8–10]. In this study, we propose an AI-based spam filtering solution by utilizing machine learning (ML) and deep learning (DL) approaches. We exploit natural language processing (NLP) to understand the content of e-mails and improve our spam mail filter accuracy. We use three different datasets. They are generated in Turkish and English language. We also take advantage of feature selection and optimization methods to enhance the accuracy performance of learning models. At this point, our contributions can be presented as follows:

We present a comprehensive and comparative study that analyzes ML and DL algorithms while considering hybrid and ensemble methods. We propose optimized solutions for spam mail detection including a novel hybrid ensemble method approach. We use Turkish and English imbalanced datasets. We also generate two new datasets by combining all Enron datasets into one and bringing together two Turkish data sets with a new one. The generated English dataset is more balanced in comparison to separate individual datasets. Hence, we consider both balanced and imbalanced datasets in our experiments. Different optimized feature selection approaches are leveraged to overcome overfitting problems and longer processing times so that we improve commonly used ML and DL algorithms in the literature. Our results also present the processing time for each technique. Different NLP solutions for modifying our data to enhance the performance of learning algorithms are utilized. To the best of our knowledge, there is a gap in the literature that aggregates and compares all these methods with a categorical point of view in addition to seeking out the best of them for different datasets. Therefore, we propose a holistic approach to spam mail detection solutions. We bridge this gap in this study by formulating recommendations for individual, ensemble, and hybrid methods by consolidating the most effective techniques and considering the running time details of each algorithm.

The rest of the paper is organized as follows. The successive session discusses a comprehensive literature review, and we present the common datasets with the newly generated datasets. The implementation of commonly used ML and DL algorithms and their comparisons are illustrated.

Moreover, we explain the steps beginning with preprocessing and splitting the datasets, feature extraction and selection, the structure for ML and DL models, and the model validation. In the Experimental Results section, we present the outcomes of our experiments using various techniques across different datasets. In the Discussion section, we reveal the best results of our solutions for datasets in different languages. In the Conclusion section, we summarize the study and discuss future work.

MATERIALS AND METHODS

Related Work

This section presents the state of the art regarding spam classification and detection of algorithms in different ways. We can roughly classify the related studies in the literature as individual, hybrid, and ensemble approaches as demonstrated in Table 1. The first category includes only ML and DL algorithms. The hybrid method including different methods comes together to propose more accurate models, on the contrary, the ensemble learning methods include the same category but different algorithms' voting or other weights. Based on the literature, most of the studies commonly develop their models using the following steps: data collection or dataset selection, data preprocessing, feature extraction and selection, training of models, and evaluation of the models. Most of the studies use accuracy, precision, recall, and F1 metrics to analyze and compare the performance of models. There are two kinds of validation methods such as cross-validation and splitting of the data as train and test.

Although many studies and algorithms have been put forward in spam email filtering, on the other hand, since spam contents have a very dynamic structure there is a need for efficient, reliable, and agile algorithmic approaches to detect new spam variants in different contexts [11]. Rustam et al. [11] make use of multiple features to increase the accuracy of the supervised ML algorithms. They utilize a composition of two feature selection methods which are the bag of words (BoW) and the term frequency-inverse document frequency (TF-IDF) feature selection methods. The authors consider imbalanced data and use a resampling approach against this issue. Random Forest (RF), Support Vector Machine (SVM), Gaussian Naive Bayes (GNB), Gradient Boosting Machine (GBM), and Logistic Regression (LR) are analyzed as machine learning models with several optimized hyperparameters. Long short-term memory (LSTM) and gated recurrent unit (GRU) are also applied for the spam mail classification. Two Kaggle datasets which are " the Spam or Ham - EMP Week 2 ML HW Dataset" and " the Spam filter Dataset" are utilized in addition to the combination of these two datasets. They use new data to validate and test the models' performances. Based on the results of that study, the LR and RF algorithms have the highest accuracy performance results which are 99% and 99.1%, respectively. In [6] evaluates the effectiveness of K-Nearest Neighbors (K-NN), Naive Bayes (NB), and SVM classifiers, the Multinomial Naive Bayes (MNB) algorithm achieving the highest accuracy performance at 96.2%.

To make an accurate detection of spam content, ML models require optimization techniques. In Gibson et al. [5], the bio-inspired metaheuristic approaches such as Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) are leveraged to optimize the following machine learning algorithms: Multi-Layer Perceptron (MLP), NB, RF, SVM, and Decision Tree (DT). They analyzed the Ling-Spam dataset, 6 different Enron datasets, the PUA dataset, and the SpamAssassin dataset. To validate their results, they split their data into 70%-80% train and 25%-20% test data. According to the results, optimized MNB based on GA has the highest performance which is 100% when the SpamAssassin dataset is used. There are various datasets including text analysis approaches and content-based features for training spam and phishing mail filtering models. The result of that study classifies the data into three classes which are spam, ham, and phishing. Magdy et al. [7] use the SpamBase dataset, the CSDMC2010 dataset, and a combination of the SpamAssassin and Nazario datasets as benchmark datasets including content-based features since text analysis approaches require more time to process. They suggest a deep learning approach based on a neural network. They select their features by implementing Low variance,

Principal Component Analysis (PCA), and Chi-squared (CHI) techniques. To validate and analyze their method performance, they exploit a 10-fold cross-validation approach. Their model gives the highest result when Chi-squared is used as a feature selection approach and the Phishing Corpus dataset is utilized as the benchmark, which consists of three classes ham, spam, and phishing in comparison to other datasets. In addition to CHI, another study utilizes different feature selection methods such as information gain (IG), accuracy (ACC), document frequency thresholding (DF), and odds rate (OR) [12]. Synthetic-minority over-sampling technique (SMOTE) is also used for preprocessing [13].

Table 1

A categorization and comparison table including state of the art individual, hybrid and ensemble approaches.

Category of solution manner	Utilized Datasets	Used Algorithms/Classifiers	Best algorithms and Performances	Feature Extraction / Preprocessing methods	Validation	Ref
Individual (DL)	Enron dataset: 32638 emails for 150 users	Feed Forward Neural Network (FFNN), BERT.	FFNN, 99.22% (F1-score)	BoW TF-IDF IDF	80% training 20% testing. and 5-folds cross validation.	[14]
(Individual) ML	E-mail msg spam detection dataset from Kaggle	MNB, Bernoulli Naive Bayes (BNB), GNB, RF, SVM	MNB, 98.8%, (accuracy)	BoW	any random email (new data)	[15]
(Individual) ML and DL	Sms-spam collection dataset form Kaggle	NB, RF, Artificial Neural Networks (ANN), SVM, LSTM, Bi-LSTM	Bi-LSTM, 98.57%, (accuracy)	TF-IDF Word Embedding Text vectorization	NA	[16]
ML and DL (Individual)	The Enron email dataset	LSTM, Bi-LSTM, BERTSVM,K-NN, MNB, DT, LR, RF	BERT, 99.14%, (accuracy)	Count Vectorizer, TF-IDF	70% for training, 30% for testing	[17]
ML (Individual)	2006 Enron corpus dataset, 2007 Trec dataset	K-NN	K-NN, 93.18%, (accuracy)	Tokenization, Removing stop words, noise, and stemming	80% for training, 20% for testing	[18]
ML (Individual)	TurkishEmail, TrHamSpamEmai lv1.0	SVM, RF, NB, K-NN, C4.5, MLP, LR, Sequential minimal optimization (SMO)	MLP %98, (accuracy)	BoW, TF-IDF, CHI, IG, ACC, OR, DF	NA	[12]
ML, DL (Individual)	Various datasets are referenced	NB, DT, RF, SVM, Neural Networks (NN), Automatic Identification System (AIS)	NB 99.46%, (accuracy)	Vectorizer	Various methods	[19]
ML, DL (Individual)	Kaggle Dataset in English, and its translated version into URDU language	NB, SVM, LSTM, Convolutional Neural Networks (CNN)	LSTM, 98.4%, (accuracy)	Tokenization	80% for training, 20% for testing	[20]
ML (Hybrid)	SPAMBASE dataset from UCI	K-NN, RF, NB, Fuzzy K-NN (FKNN)	BGWOA-FKNN 97.61%, (accuracy)	Black widow Optimization Algorithm (BWO), Binary Grey Wolf Optimization (BGWO)	10-fold cross-validation	[21]
ML and DL (Hybrid)	Enron email dataset, SMS Spam Collection dataset	CNN, RF, SVM, LSTM with RNN, DT, GNB, XGB	CNN with the Glove model, 96.52%, (accuracy)	Count Vectorizer, TF-IDF, Word Embedding	a 10-folds cross-validation mode.	[22]
DL, ML (Hybrid)	TREC Public Spam, GenSpam, SA , Enron (EN) , and LS .	SVM, LR, K-NN, RF, BERT + SVM, BERT + K-NN, BERT + LR, Federating Learning (FT) + Hierarchical Attention Network (HAN)	FT+HAN, 99.2%, (accuracy)	Word Embedding	10-fold cross validation, splitting train and test	[23]
DL, ML (Hybrid)	Enron, Dredze, TREC 2007	SVM, CNN, K-NN, multi-modal architecture based on model fusion (MMA-MF), Proximal Support Vector Machine (P-SVM)	99.16%, (accuracy)	Paragraph Vector, Doc2Vec DBoW	5-fold cross validation	[24]
ML (Hybrid)	Enron Spam dataset form UCI	GA, DT, K-NN, SVM, J-48, NB, GADT	GADT 95.5%, (accuracy)	BoW,TF-IDF, PCA	NA	[25]
ML, DL (Ensemble)	SMS spam collection from UCI	SVM, NB, K-NN, RF, CNN, LSTM, Deep Neural Networks (DNN) + Discounted cash flow (DCF)	DCF, 98.38%, (accuracy)	Word Embdedding, SMOTE	10-fold cross validation 80% for training, 20% for testing	[13]
Ensemble (ML)	Email Spam Classification Dataset, Spam filter dataset from Kaggle	DT, SVM, NB, MLP, VC	99% (accuracy)	NA	Three cases of train:test ratios - 70:30, 80:20, and 90:10	[26]
ML, (Ensemble)	SMS-Spam-Collection-Dataset from Kaggle,Twitter spam dataset from nclab	K-NN, NB, ETC, RF, SVC, LR, XGB, DT, the proposed VC	VC, 97.96%, (accuracy) ETC, 97.77% (accuracy)	Vectorizer	80% for training, 20% for testing	[27]
ML, DL, (Ensemble), (Hybrid) (Ensemble + Hybrid)	ENRON Dataset Turkish spam dataset from Kaggle Turkish spam dataset from UCI	K-NN, MNB, SVM, RF, SVM, LR, DT, BI˙-LSTM, LSTM, BERT, BERT-TURK	K-NN+MNB with stacking, 99.9%, (accuracy) BERT-Turk + SVM with simple average, 98.43% (accuracy)	TF-IDF, BoW Tokenizer, Word2Vec, Keras, BERT-Turk	80% for training, ⅕ for testing 5-fold cross-validation	**our study**

The e-mail data classification such as normal, harassment, fraudulent, and suspicious is another important issue [28]. Some disadvantages of skipping meaningful information of some filtration and keyword-based search algorithms cause short sequence emails and extraneous problems. Thus, the study

proposes a multi-class email classification solution relying on LSTM based on a Gated Recurrent Neural Network (GRU) to overcome these limitations. In that study, some of the feature extraction techniques such as Word2Vec, TF-IDF, Vector Normalization, BoW, and embedding vector for the DL algorithm. This study presents a hyper-tuned DL scheme. In this study, the loss function is selected as Categorical cross-entropy and the value of the weights optimizer is chosen as the ADAM optimizer. The accuracy of the model is the percentage of 95%. A self-extended dataset is used including three original e-mail datasets, social media sources from Twitter, and criminal activities. For the validation part, they split their data into three sub-datasets which are training, validation, and test. The ratio of the three sub-datasets is 65, 10, and 25, respectively.

All in all, there are many state-of-the-art kinds of research presented in Table 1 by categorizing them. We provide a holistic approach to show the best algorithms by relying on individual methods including ML-based feature selection methods, hybrid and ensemble methods. ML-based feature selection methods called hybrid ensemble algorithms have the best performance solutions even if we have different data sets in different languages.

Ml-Based and Dl-Based Methods by Using Different Datasets In Turkish and English

In this study, we use several datasets, including the open-source six pieces of the Enron dataset for detecting spam emails in English, and open-source datasets from UCI and Kaggle for Turkish spam email detection. Subsequently, the dataset is preprocessed by employing the most suitable techniques. The preprocessed datasets are then transformed into numerical vectors using various methods such as BoW, TF-IDF, Word2Vec, and Glove. This transformation is carried out to make the data suitable for input into ML, DL models, and other different learning model implementations. We select various ML and DL algorithms demonstrating effective performance in practical applications, including document classification and spam filtering. ML approaches including the MNB algorithm [29-31], SVM [32], K-NN [33,34], DT [35], RF [35], and LR [36] algorithms as well as ensemble learning methods like VC and Stacking classifier (SC), are trained. DL algorithms such as LSTM [37, 38], Bi-LSTM [39], BERT [40, 41], and BERTurk [42] are employed, and their accuracy values are compared. All these steps are discussed in-depth in this paper. Figure 1 illustrates our methodology. We use the Google Colab platform to conduct our experiments. The system properties are like the following: Linux 5.15.120+ x86_64, 12 GB RAM, 110 GB Harddisk, Tesla T4 GPU, and Intel(R) Xeon(R) CPU @ 2.20GHz, Thread(s) per core: 2, Core(s) per socket: 1.

Datasets

English Datasets

The detection of spam emails in English is carried out using the open-source Enron dataset [43]. Within the Enron dataset, six separate datasets are chosen for analysis, categorized into spam and raw emails in varying quantities. Here's a breakdown of the Enron datasets: Enron 1 dataset comprises a total of 5,172 email texts, with 3,672 being raw emails and 1,500 being spam emails; Enron 2 dataset contains a total of 5,857 email texts, consisting of 4,361 raw emails and 1,496 spam emails; Enron 3 dataset includes a total of 5,512 email texts, with 4,012 being raw emails and 1,500 being spam emails; Enron 4 dataset encompasses a total of 6,000 email texts, with 4,500 being spam emails and 1,500 being raw emails; Enron 5 dataset consists of a total of 5,175 email texts, with 3,675 being spam emails and 1,500 being raw emails; Enron 6 dataset contains a total of 6,000 email texts, with 4,500 being spam emails and 1,500 being raw emails. Each of these datasets is used to analyze and detect spam emails.

The process for handling the English datasets begins with the extraction of data from compressed files. Then we read the email contents from text files and save them in comma-separated values (CSV) file format. During this process, we determine that there are no missing data points within the datasets.

However, duplicate text contents are identified as follows: 178 in the Enron 1 dataset, 33 in the Enron 2 dataset, 238 in the Enron 3 dataset, 149 in the Enron 4 dataset, 63 in the Enron 5 dataset, and 11 in the Enron 6 dataset. To ensure that these duplicate entries are systematically removed from the datasets, making them ready for preprocessing steps.

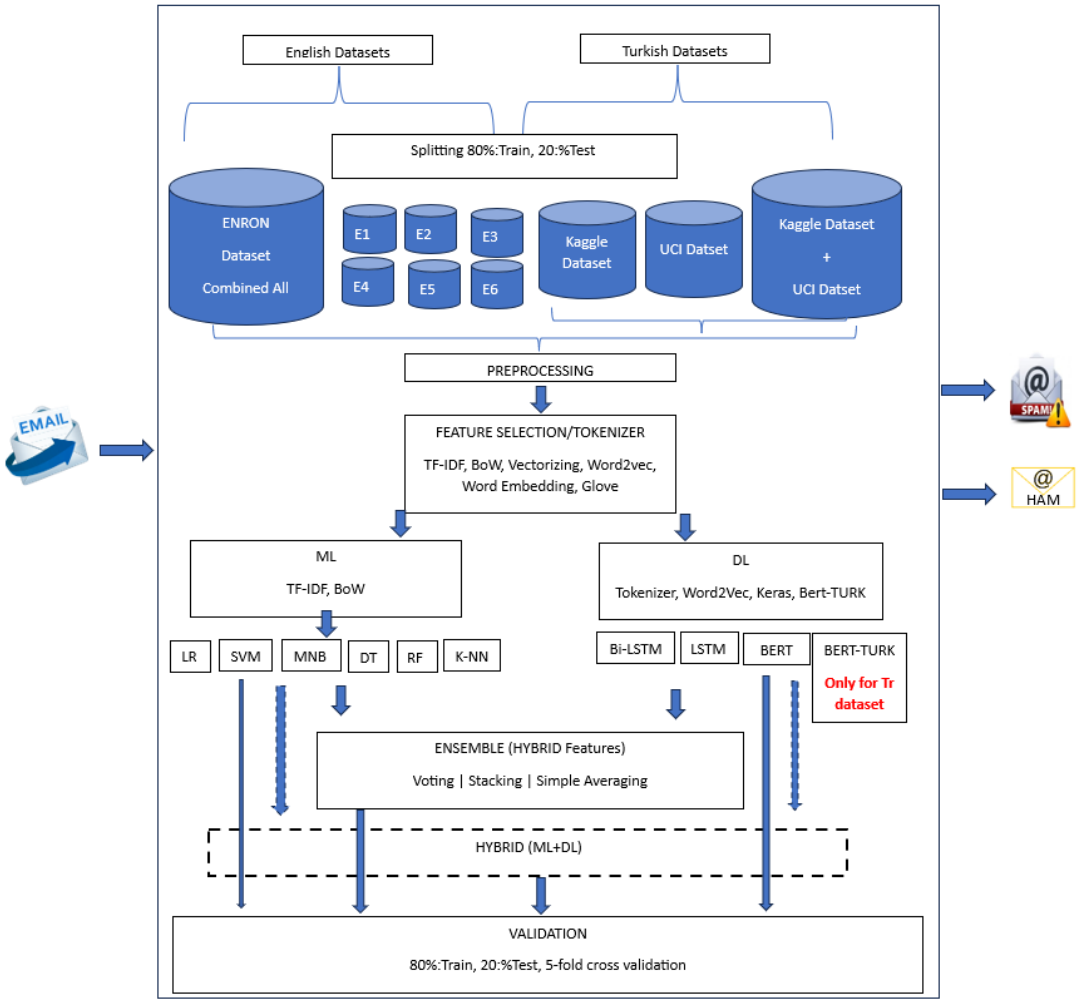

Figure 1
Proposed spam/ham mail detection methodology.

Turkish Datasets

To detect spam emails in Turkish, open-source datasets from Kaggle [44] and UCI [45] websites are utilized. The Kaggle dataset comprises a total of 1,015 email texts, with 514 categorized as spam and 501 as raw. In contrast, the UCI dataset contains 790 email contents, consisting of 324 spam emails and 466 raw emails. To facilitate the analysis, the content from the Kaggle dataset is extracted from text files and converted into CSV file format. However, the UCI dataset initially presented challenges as it could not be correctly read in CSV format. As a result, is processed to be converted back to CSV format to make it suitable for analysis. During the analysis, we determine that the UCI dataset has one row of missing data. The Kaggle dataset contains four repeated rows, and the UCI dataset has 71 repeated rows. The identified duplicate rows are removed from the dataset to eliminate such records that may potentially cause issues in future analyses and to prepare for preprocessing.

Data Preprocessing

The initial step in any ML or data analysis solution involves cleaning and processing the data. The execution of these procedures plays a key role in the success of the intended model. In this study, a

comprehensive exploration of commonly employed preprocessing steps including NLP tasks for textual data is conducted. These preprocessing procedures encompass the following steps which are the removal of stop words, and elimination of the symbols, signs, punctuation marks, and numerical values from the text to ensure uniform treatment of all text data. The others are like the following: creating regular expressions to handle special cases such as emails, URL domains, HTML tags, and codes, and then removing them from the dataset, discarding words that do not contribute to the meaningful content of the text, and converting words to lowercase. Each of the described steps is applied to every dataset individually. Subsequently, each of the English and Turkish datasets are separately merged and prepared for analysis. Once the datasets have undergone preprocessing, they are prepared for conversion into digital vectors, a crucial step in many NLP and ML tasks.

Feature Selection Methods

Text analysis is a significant domain for ML algorithms. However, these algorithms typically require fixed-size numerical feature vectors as input, whereas raw text documents come in variable lengths. To bridge this gap, several methods have been developed to extract numerical features from email content. Within the scope of this project, some of these methods, namely BoW, TF-IDF, Word2Vec, and Glove are employed. The BoW technique serves as a method for extracting features from text, which can then be utilized in ML algorithms. BoW is primarily used to determine whether known words are present in the text [46]. The TF-IDF method is a statistical approach employed to assess the significance of a word within an entire corpus of text [47]. Its importance increases proportionally to the frequency of the word's occurrence within a document [48]. Word2Vec is a transfer learning algorithm enabling the conversion of text data into input suitable for ML algorithms and facilitating model training [49]. Glove, Global Vectors for Word Representation, is a type of word embedding method. In Glove, each word is represented as a vector, and these vectors are computed based on the word's usage in language and its associations with other words within a specific context [50, 51].

In the conducted studies, models are trained using BoW and TF-IDF vectors with ML algorithms. Additionally, in the context of DL algorithms, models are trained using Word2Vec and Glove methods. Furthermore, uncased tokenizers, readily available in BERT models, and Keras tokenizer vectors from the Keras library are trained and evaluated. The results obtained from these different approaches are compared for the analysis.

Splitting of Datasets for Training and Test Steps

After performing all the preprocessing steps and being transformed into numerical vectors, the datasets are prepared for model training. They are split into two sets: an 80% training set and a 20% test set. This division allows for the training and evaluation of models using the data.

Evaluation Metrics

In the study, various classification performance evaluation metrics are utilized, including accuracy, recall, precision, and F1-score. These metrics rely on different aspects of model performance such as True Positives (TP), True Negatives (TN), False Positives (FP), False Negatives (FN). TP are values that are true and correctly predicted. The TN Values are not correct and are correctly predicted as such. The FP Values mean that are not correct but are incorrectly predicted as correct. The values of FN are correct but are incorrectly predicted as not correct. The representation of these components in a confusion matrix is illustrated in Table 2. This matrix provides a comprehensive view of the model's performance across various categories, facilitating the assessment of classification results.

Table 2
Confusion matrix.

	Predicted (HAM) 0	Predicted (SPAM) 1
Actual (HAM) 0	TN	FP
Actual (SPAM) 1	FN	TP

This study uses accuracy, recall, precision, and F1-score as evaluation parameters to assess the performance of classification models, and these metrics can be calculated based on the values presented in the confusion matrix, as follows:

The value of accuracy is calculated by using (1):

$$\text{Accuracy} = \frac{(\text{TP} + \text{TN})}{(\text{TP} + \text{FP} + \text{TN} + \text{FN})} \tag{1}$$

We can find the value of recall by utilizing (2):

$$\text{Recall} = \frac{\text{TP}}{(\text{TP} + \text{FN})} \tag{2}$$

The value of precision is measured by (3):

$$\text{Precision} = \frac{\text{TP}}{(\text{TP} + \text{FP})} \tag{3}$$

F1-Score equals to (4):

$$\text{F1-Score} = \frac{2 \times (\text{precision} \times \text{recall})}{(\text{precision} + \text{recall})} \tag{4}$$

EXPERIMENTAL RESULTS

Machine Learning Results

In the Enron 1 dataset, LR and SVM models, trained with TF-IDF vectors, achieve an accuracy of 99.1%. The Enron 2 dataset has an accuracy of 99.74% result from a MNB model trained with TF-IDF vectors achieving. For the Enron 3 dataset, an MNB model trained with TF-IDF vectors reaches an accuracy of 99.62%. In the Enron 4 dataset, an LR model achieves an accuracy of 98.88%. For the Enron 5 dataset, a MNB model, trained with BoW vectors, attains an accuracy of 99.51%. On the Enron 6 dataset, the K-NN model, trained with TF-IDF vectors, reaches an accuracy of 98.08%. Finally, for the combined dataset, an SVM model, trained with TF-IDF vectors, achieves the highest accuracy of 99.51%.

Similarly, for the detection of spam emails in Turkish, the training process involved exploring all hyperparameter combinations, and subsequently, the models are retrained and tested using the hyperparameters that yield the best results. After conducting these tests, the model's performance is evaluated using various evaluation metrics. The best results of the ML models developed for Turkish and English spam email detection, along with their respective results, can be found in Table 3.

Deep Learning Results

In the Enron 1 dataset, the Bi-LSTM model, trained with Word2Vec vectors, achieves an accuracy of 97.49%. In the Enron 2 dataset, a Bi-LSTM model trained with Keras Tokenizer word vectors reaches an accuracy of 98.12%. The same accuracy is also attained by the Bi-LSTM model trained with Keras Tokenizer word vectors in the Enron 3 dataset. For the Enron 4 dataset, the Bi-LSTM model trained with Word2Vec vectors achieves an accuracy of 97.82%. In the Enron 5 dataset, a Bi-LSTM model

trained with Keras Tokenizer word vectors excels with an accuracy of 99.16%. The LSTM model training with Word2Vec vectors has the highest performance in the Enron 6 dataset, with an accuracy of 99.13%. In the combined dataset, the Bi-LSTM model training with Word2Vec vectors achieves the highest accuracy of 99.26%. Various models are created for each dataset for spam email detection in the Turkish language. The initial step involves preprocessing the data and converting it into numerical vectors using various methods such as Keras Tokenizer, Word2Vec, Glove, and BERT Tokenizer. After analyzing the results, all the datasets are merged, and model training is conducted once more. During the model training process, hyperparameters are tuned to attain the highest accuracy values. The best results of the models developed, and their respective outcomes can be found in Table 4.

Table 3
ML Methods Results.

Dataset	Model	Accuracy	Recall	Precision	F1-Score	Train Time (s)	Prediction Time (s)	Total Time (s)	Best Parameters
Enron1	LR TF-IDF	99.09%	99.09%	99.10%	99.10%	0.82	0.001	0.82	penalty: l2, solver: lbfgs
Enron2	MNB BoW	99.74%	99.74%	99.74%	99.74%	0.01	0.001	0.01	alpha: 0.01
Enron3	MNB TF-IDF	99.62%	99.62%	99.62%	99.62%	0.05	0.01	0.06	alpha: 0.001
Enron4	LR BoW	98.88%	98.88%	98.89%	98.88%	0.62	0.001	0.62	C: 100.0, penalty: l2, solver: lbfgs
Enron5	MNB BoW	99.51%	99.51%	99.51%	99.51%	0.02	0.001	0.02	alpha: 0.01
Enron6	K-NN TF-IDF	98.08%	98.08%	98.09%	98.06%	0.01	0.39	0.41	n neighbors: 3
Combined Enron Dataset	SVM TF-IDF	99.51%	99.51%	99.51%	99.51%	2433.18	42.38	2475.57	C: 10, gamma: 0.1, kernel:rbf
Kaggle	MNB TF-IDF	94.55%	94.55%	94.63%	94.53%	0.005	0.001	0.006	alpha: 0.01
UCI	LR TF-IDF	93.75%	93.75%	94.09%	93.68%	0.58	0.007	0.58	C: 1000.0, penalty: l2,solver: saga
Combined Turkish Dataset	MNB TF-IDF	95.08%	95.08%	95.13%	95.06%	0.01	0.03	0.01	alpha: 0.1

Table 4
DL Methods Results

Dataset	Model	Tokenizer	Optimizer	Epoch	Batch	Train Time (s)	Prediction Time (s)	Total Time (s)	Accuracy	Loss
Enron1	Bİ-LSTM	Word2Vec	Adam	15	32	290	2	292	97.49%	0.06
Enron2	Bİ-LSTM	Keras Tokenizer	Adam	10	64	1013	12	1025	98.12%	0.07
Enron3	Bİ-LSTM	Word2Vec	Adam	10	32	209	2	211	97.82%	0.06
Enron4	Bİ-LSTM	Word2Vec	Adam	10	32	209	2	211	97.82%	0.06
Enron5	LSTM	Word2Vec	Adam	20	32	69	0	69	99.13%	0.03
Enron6	LSTM	Word2Vec	Adam	20	32	46	0	46	98.42%	0.05
Combined Enron Dataset	Bİ-LSTM	Word2Vec	Adam	20	32	2333	16	2349	99.22%	0.03
Kaggle	BERTurk	Bert Tokenizer	AdamW	5	16	150	10	160	97.24%	0.05
UCI	BERTurk	Bert Tokenizer	AdamW	10	16	110	10	120	96.07%	0.07
Combined Turkish Dataset	BERTurk	Bert Tokenizer	AdamW	12	16	319	24	343	97.72%	0.03

Hybrid, Ensemble, and the Combination of Those Learning Methods

In this section, we analyze the methods ensemble methods in addition to hybrid feature selection methods and hybrid methods including the best ML and DL algorithms.

Ensemble Learning is an ML approach designed to enhance overall prediction performance by aggregating the predictions of multiple models. This strategy aims to achieve more accurate predictions by leveraging the strengths of different models while mitigating their shortcomings. It operates on the concept that combining weak models can create a more powerful and robust predictive model. Ensemble Learning has various applications, including improving decision reliability, optimizing or near-optimizing feature selection, data fusion, incremental learning, handling non-stationary learning scenarios, and error correction [52]. The core idea behind this technique is rooted in the understanding that when individual forecasts are appropriately combined, the collective decision of the ensemble provides better overall accuracy than any individual committee member's prediction [53-55].

In the study, new models are trained by combining ML and DL models using voting, stacking, and simple average ensemble techniques. The VC is a type of ensemble learning technique that allows

for the combination of multiple diverse ML models. Once each model selected for ensemble learning is trained, their outputs are subjected to a voting process. This voting can be performed in two main ways: hard or soft. In the hard voting process, also known as majority voting, the final output is determined by taking most of the individual model outputs. In the soft voting approach, the output values are averaged, and the prediction corresponds to the output with the highest average. After completing the voting process, the ensemble produces a prediction result by aggregating the decisions of its constituent models. This technique is beneficial because it often results in improved prediction accuracy compared to using individual models in isolation [56]. The SC is an ensemble learning technique that combines multiple ML models. Once the selected models for ensemble learning are trained, their predictions are collected. Subsequently, the final predictions are generated using a meta-model, which is constructed based on the combined predictions of these models. The SC is particularly useful when different models have unique strengths and weaknesses, as it can exploit the complementary aspects of these models to improve overall prediction performance [57]. In the simple average ensemble method, the trained models can be combined to improve the predictions. In this method, first, the models to be predicted are trained. Then, the prediction results of the trained models are averaged. The resulting average value is considered as the result predicted by the model. After all model developments are completed with ML and DL algorithms, model developments are carried out by hybridizing the three ML algorithms that gave the best results for both Turkish and English data sets, DL algorithms, and finally ML and DL algorithms together. Voting and stacking ensemble methods are used in ML and ML hybrid models, the stacking ensemble method is used in DL and DL hybrid models, and the simple average ensemble method is used in ML and DL hybrid models.

Ensemble, Hybrid and Hybrid Ensemble Learning Results

For the Enron 1 dataset, the LR and MNB hybrid model using TF-IDF vectors created by stacking have the best outcomes. In the Enron 2 dataset, the LR model utilizing TF-IDF vectors depending on stacking gives the highest accuracy. The MNB and SVM hybrid model, along with LR trained with TF-IDF vectors using stacking, achieves the best results in the Enron 3 dataset. For the Enron 4 dataset, the MNB hybrid model and the LR and SVM hybrid model, both trained with TF-IDF vectors, provide the best results. The K-NN and MNB hybrid model using TF-IDF vectors delivers the top results In the Enron 5 dataset. For the Enron 6 dataset, the MNB and SVM hybrid model, based on TF-IDF vectors, provides the best performance. The MNB and SVM hybrid model, consisting of TF-IDF vectors, yields the best results for the combined dataset. These outcomes suggest that different hybrid models have superior performance on different datasets. It can be said that these hybrid approaches have led to increased accuracy in most datasets, with the MNB model performing exceptionally well in the Enron 2 dataset. This demonstrates the versatility of ensemble methods and hybridization in improving model performance across a range of data scenarios.

For the Kaggle dataset, the best performance belongs to the SVM and MNB hybrid model trained with TF-IDF vectors created through voting. However, for the UCI dataset, the LR and MNB hybrid model using TF-IDF vectors depending on voting delivered the highest accuracy. If we look at the results of the combined dataset, stacking with LR and MNB hybrid models trained with TF-IDF vectors resulted in the best performance. In the light of the results, it is observed that different hybrid models have more successful results for different datasets. In the case of the Kaggle and UCI datasets, the hybrid models achieve the same accuracy value as the MNB and LR models for each respective dataset, while underscoring the effectiveness of ensemble techniques. Additionally, the MNB model performed exceptionally well in the combined dataset. This demonstrates the flexibility and utility of hybridization and ensemble methods in achieving high accuracy across various data scenarios. The best ML ensemble models, and their respective outcomes can be found in Table 5.

Table 5
ML-based ensemble model results.

Dataset	Model	Ensemble	Accuracy	Recall	Precision	F1-Score	Train Time (s)	Prediction Time (s)	Total Time (s)
Enron1	LR + MNB TF-IDF	Stacking	99.19%	99.19%	99.21%	99.20%	320.23	1.60	321.84
Enron2	LR + MNB + SVM TF-IDF	Stacking	99.65%	99.65%	99.65%	99.65%	414.28	2.91	417.20
Enron3	LR + MNB TF-IDF	Stacking	99.71%	99.71%	99.71%	99.71%	72.22	0.02	72.25
Enron4	LR + SVM TF-IDF	Stacking	99.48%	99.48%	99.48%	99.48%	357.28	2.55	359.84
Enron5	KNN + MNB TF-IDF	Stacking	99.90%	99.90%	99.90%	99.90%	26.25	0.01	26.26
Enron6	KNN + MNB TF-IDF	Stacking	98.99%	98.99%	98.99%	98.99%	2.13	0.51	2.64
Combined Enron Dataset	SVM + MNB TF-IDF	Stacking	99.62%	99.62%	99.62%	99.62%	45.66	0.15	45.82
Kaggle	SVM + MNB TF-IDF	Voting	94.55%	94.55%	94.63%	94.53%	0.10	0.002	0.10
UCI	LR + MNB TF-IDF	Voting	93.75%	93.75%	94.09%	93.68%	0.16	0.001	0.17
Combined Turkish Dataset	LR + MNB TF-IDF	Stacking	94.79%	94.79%	94.82%	94.78%	0.16	0.01	0.17

The experimental results generated by one of the DL-based methods for the English language, which is the BERT + Bi-LSTM hybrid model has an accuracy value of 94.49% in the Enron 1 data set, 94.62% in the Enron 2 data set, and 95.83% and 97.97% accuracy in the Enron 3 and Enron 4 data sets, respectively. BERT + LSTM hybrid model with 97.20% accuracy on the Enron 5 dataset, BERT + LSTM hybrid model with 95.67% accuracy on the Enron 6 dataset, and BERT + Bi with 95.61% accuracy on the combined dataset. According to the results, the Bi-LSTM model provides the best performance. When analyzing the Turkish language, BERTurk + Bi-LSTM hybrid models show the best performance with an accuracy value of 87.19% in the Kaggle dataset, 89.02% in the UCI dataset, and 92.20% in the combined dataset, respectively. The best results of the DL ensemble models developed, and their respective outcomes can be found in Table 6.

Table 6
DL-based ensemble model results.

Dataset	Model	Tokenizer	Optimizer	Epoch	Batch	Train Time (s)	Prediction Time (s)	Total Time (s)	Accuracy	Loss
Enron1	BERT + Bİ-LSTM	BERT Tokenizer	Adam	20	32	1344	11	1355	94.49%	0.15
Enron2	BERT + Bİ-LSTM	BERT Tokenizer	Adam	10	64	1013	12	1025	98.12%	0.07
Enron3	Bİ-LSTM	BERT Tokenizer	Adam	20	32	1385	12	1397	95.83%	0.12
Enron4	Bİ-LSTM	BERT Tokenizer	Adam	15	32	1181	13	1194	97.67%	0.07
Enron5	LSTM	BERT Tokenizer	Adam	20	32	1325	11	1336	97.20%	0.09
Enron6	LSTM	BERT Tokenizer	Adam	20	32	1476	13	1489	95.67%	0.11
Combined Enron Dataset	Bİ-LSTM	BERT Tokenizer	Adam	17	32	6753	75	6828	95.61%	0.11
Kaggle	BERTurk	Bert Tokenizer	AdamW	20	32	75	0	75	87.19%	0.70
UCI	BERTurk	Bert Tokenizer	AdamW	20	32	65	0	65	89.02%	0.47
Combined Turkish Dataset	BERTurk	Bert Tokenizer	AdamW	20	32	97	0	97	92.20%	0.42

Table 7
Hybrid ensemble model results.

Dataset	Model	Accuracy	Recall	Precision	F1-Score	Train Time (s)	Prediction Time (s)	Total Time (s)
Enron1	LSTM + SVM	99.19%	99.34%	98.05%	98.69%	205.15	154.90	360.05
Enron2	LSTM + SVM	99.57%	99.31%	98.97%	99.14%	206.53	141.76	348.29
Enron3	LSTM + SVM	99.57%	99.31%	98.97%	99.14%	206.53	141.76	348.29
Enron4	LSTM + SVM	99.48%	100%	99.29%	99.64%	169.53	107.98	277.51
Enron5	LSTM + SVM	99.70%	100%	99.58%	99.79%	137.77	97.29	235.06
Enron6	LSTM + KNN	98.41%	99.77%	98.09%	98.92%	85.81	476.11	561.92
Combined Enron Dataset	Bi-LSTM + LR	99.39%	99.79%	99.01%	99.40%	2449.35	2029.55	4478.91
Kaggle	BERTurk + MNB	97.92%	99.41%	96.56%	97.96%	167.04	11.85	178.89
UCI	BERTurk + MNB	97.77%	98.83%	95.14%	96.95%	115.24	14.35	129.60
Combined Turkish Dataset	BERTurk + SVM	98.43%	98.82%	97.68%	98.25%	337.95	25.96	363.91

In the study conducted for the English language, the accuracy values are 99.19% in the Enron 1 data set, 99.57% in the Enron 2 data set, 99.14% in the Enron 3 data set, 99.48% in the Enron 4 data set, and 99.70% in the Enron 5 data set, respectively. The LSTM + SVM hybrid model shows the best performance, the LSTM + KNN hybrid model with an accuracy value of 98.41% on the Enron 6 dataset, and the Bi-LSTM + LR model with an accuracy value of 99.39% on the combined dataset. In the study

conducted for the Turkish language, BERTurk + MNB hybrid models showed the best performance with accuracy values of 97.92% and 97.77% in the Kaggle and UCI datasets, respectively, and BERTurk + SVM hybrid model with 98.43% accuracy in the combined dataset. The best results of the hybrid ensemble models developed, and their respective outcomes can be found in Table 7.

DISCUSSION AND PERFORMANCE EVALUATION

In this study, ML and DL algorithms are employed to address real-world problems, specifically related to spam email detection. The algorithms selected for this research are developed as potential solutions for the classification problem concerning spam emails. This study focuses on several key aspects, including pre-processing steps, vectorization of data, and the impact of the raw or spam ratio on performance. Experiments are conducted using a total of 1085 Turkish email texts and 33,716 English email texts. The datasets are open access. Each of these datasets is subjected to several key steps such as pre-processing and vectorization before being applied to various models. We extensively discuss these crucial steps in the Methodology section. To ensure accuracy in the comparison of these models, ML algorithms are assessed according to precision, recall, F1-score, and accuracy metrics, while DL algorithms are evaluated based on accuracy and loss values. The results are systematically recorded, and statistical information is employed to analyze and compare the findings. Based on the performance results obtained, the study conducts comparisons using different methods and presents comprehensive evaluations. This research contributes to the understanding of how various factors, including data pre-processing and vectorization, can impact the performance of ML and DL models in the context of spam email detection.

For English datasets, the best results are obtained while using one of the ML algorithms which is MNB models trained with TF-IDF vectors. Among the DL algorithms, it has been observed that Bi-LSTM models using Word2Vec vectors give the best results. Among the ensemble learning methods, the stacking method created with ML algorithms has been shown to produce the best results. Thus, spam mail detection in English is solved with 99.9% as the highest accuracy value, with at least 97% accuracy performance.

According to the Turkish data sets, the best results are obtained in spam mail detection with MNB models trained with TF-IDF vectors, one of the ML algorithms. Among the DL algorithms, it has been observed that BERTurk models trained with BERTurk tokenizer vectors give the best results. Among the ensemble learning methods, the simple average method created with ML and DL algorithms has achieved the best results. Thus, spam mail detection in the Turkish language is solved with 98.4% accuracy, with at least 93% accuracy value.

All in all, although this study does not produce a generic solution for spam mail detection on the languages used, the obtained results show that especially hybrid and ensemble learning methods and some ML algorithms provide more successful outcomes.

CONCLUSION

In recent years, the prevalence of spam emails has increased significantly due to the widespread use of email in business and personal communication, as well as the growth of social media platforms. Effectively detecting and categorizing this rising volume of spam emails is a critical concern to safeguard systems and prevent issues like fraudulent activities. This paper conducts a comparative analysis and literature review to identify the most efficient methods for distinguishing between spam and legitimate emails, using datasets in both Turkish and English. In addition to existing datasets, we've also created new datasets in these languages. In this study, we present a comparative and comprehensive study taking advantage of various approaches which are ML, DL, hybrid, and ensemble methods. These

methods are combined with NLP techniques, enhanced learning algorithms, optimized feature selection, and preprocessing. We compare our solutions with commonly used methods in the field such as MNB, SVM, LR, K-NN, DT, RF, VC, SC, and DL algorithms like LSTM, Bi-LSTM, BERT, and BERTurk. The results are validated by splitting the datasets into training and testing data with an 80:20 ratio and using 5-fold cross-validation for each model. Additionally, the hyperparameters of our models are fine-tuned using Grid search. The best feature selection method for the Enron dataset is the TF-IDF approach, while the simple average ensemble learning approach is for the Turkish dataset. The findings reveal that ensemble methods based on ML achieve the best performance, achieving an accuracy rate of 99.9% for the English dataset. In comparison, a hybrid ensemble approach with a simple average produces the highest accuracy of 98.43% for the Turkish dataset.

These proposed methods are well-suited for various platforms, including email and social media, to effectively detect and categorize spam content. In future work more efficient and real-time models can be proposed by using transfer learning approaches, large language models (LLMs), and federated learning approaches to detect spam mail so we can decrease the overhead of this problem on the Internet.

Author Contributions

Research Design (CRediT 1) E.N.C. (%34) – R.K. (%33) – A.E. (%33)
Data Collection (CRediT 2) E.N.C. (%34) – R.K. (%33) – A.E. (%33)
Research - Data Analysis - Validation (CRediT 3-4-6-11) E.N.C. (%39) – R.K. (%38) – A.E. (%23)
Writing the Article (CRediT 12-13) E.N.C. (%39) – R.K. (%23) – A.E. (%38)
Revision and Improvement of the Text (CRediT 14) E.N.C. (%39) – R.K. (%23) – A.E. (%38)

Conflict of interest

The authors declare that they have no conflict of interest.

REFERENCES

[1] J. Doshi, K. Parmar, R. Sanghavi, N. Shekokar, A comprehensive dual-layer architecture for phishing and spam email detection, *Computer & Security*. 133 (2023), 103378. doi:10.1016/j.cose.2023.103378

[2] N. Saidani, K. Adi, MS. Allili, A semantic-based classification approach for an enhanced spam detection. *Computer & Security.* 94 (2020), 101716. doi:10.1016/j.cose.2020.101716

[3] B. Feng, Q. Fu, M. Dong, D. Guo, Q. Li, Multistage and elastic spam detection in mobile social networks through deep learning, *IEEE Network*. 32(4) (2018), 15-21. doi:10.1109/MNET.2018.1700406

[4] A. Karim, S. Azam, B. Shanmugam, K. Kannoorpatti, M. Alazab, A comprehensive survey for intelligent spam email detection, *IEEE Access.* 7 (2019), 168261-168295. doi:10.1109/ACCESS.2019.2954791

[5] S. Gibson, B. Issac, L. Zhang, SM. Jacob, Detecting spam email with machine learning optimized with bio-inspired metaheuristic algorithms, *IEEE Access.* 8 (2020), 187914-187932. doi:10.1109/ACCESS.2020.3030751

[6] S. Rapacz, P. Chołda, M. Natkaniec, A method for fast selection of machine-learning classifiers for spam filtering, *Electronics*. 10(17) (2021), 2083. doi:10.3390/electronics10172083

[7] S. Magdy, Y. Abouelseoud, M. Mikhail, Efficient spam and phishing email filtering based on deep learning, *Computer Networks*. 206 (2022), 108826. doi:10.1016/j.comnet.2022.108826

[8] F. Ozen, R. Ortac Kabaoglu, T. V. Mumcu, Deep Learning Based Temperature and Humidity Prediction, *Necmettin Erbakan University Journal of Science and Engineering*. 5(2) (2023). 219-229. doi:10.47112/neufmbd.2023.20

[9] M. Hacıbeyoglu, M. Çelik, Ö. Erdaş Çiçek, Energy Efficiency Estimation in Buildings with K Nearest Neighbor Algorithm, *Necmettin Erbakan University Journal of Science and Engineering*, 5 (2) (2023), 65-74. doi:10.47112/neufmbd.2023.10

[10] A. Pektaş, O. İnan, Application of Tree Seed Algorithm on Clustering Problems, *Necmettin Erbakan University Journal of Science and Engineering*. 4(1) (2022), 1-10. doi:10.47112/neufmbd.2022.8

[11] F. Rustam, N. Saher, A. Mehmood, E. Lee, S. Washington, I. Ashraf, Detecting ham and spam emails using feature union and supervised machine learning models, *Multimedia Tools and Applications.* 82 (2023), 26545–26561. doi: 10.1007/s11042-023-14814-2

[12] E. E. Eryılmaz, D. Ö. Şahin, E. Kılıç, Türkçe İstenmeyen E-postaların Farklı Öznitelik Seçim Yöntemleri Kullanılarak Makine Öğrenmesi Algoritmaları ile Tespit Edilmesi, *Türkiye Bilişim Vakfı-Bilgisayar Bilimleri ve Mühendisliği Dergisi.* 13(2) (2020), 57-77.

[13] M. A. Shaaban, Y. F. Hassan, S. K. Guirguis, Deep convolutional forest: a dynamic deep ensemble approach for spam detection in text, *Complex & Intelligent Systems.* 8(6) (2022), 4897-4909. doi:10.1007/s40747-022-00741-6

[14] S. Kaddoura, O. Alfandi, N. Dahmani, A spam email detection mechanism for English language text emails using a deep learning approach, In: *29th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)*, *IEEE,* Bayonne, France, 2020: 193-198. doi:10.1109/WETICE49692.2020.00045

[15] T. Toma, S. Hassan, M. Arifuzzaman, An analysis of supervised machine learning algorithms for spam email detection, In: *International Conference on Automation, Control, and Mechatronics for Industry 4.0 (ACMI), IEEE,* Rajshahi, Bangladesh, 2021: 1-5. doi:10.1109/ACMI53878.2021.9528108

[16] C. M. Shaik, N. M. Penumaka, S. K. Abbireddy, V. Kumar, S. Aravinth, Bi-LSTM and conventional classifiers for email spam filtering, In: *Third International Conference on Artificial*

Intelligence and Smart Energy (ICAIS), *IEEE*, Coimbatore, India, 2023: 1350-1355. doi:10.1109/ICAIS56108.2023.10073776

[17] K. Debnath, N. Kar. Email spam detection using deep learning approach, In: *International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COM-IT-CON)*, *IEEE*, Faridabad, India, 2022: 37-41. doi:10.1109/COM-IT-CON54601.2022.9850588

[18] A. R. Yeruva, D. Kamboj, P. Shankar, U. S. Aswal, A. K. Rao, C. Somu, E-mail spam detection using machine learning—KNN, In: *5th International Conference on Contemporary Computing and Informatics (IC3I), IEEE*, Uttar Pradesh, India, 2022: 1024-1028. doi:10.1109/IC3I56241.2022.10072628

[19] N. Ahmed, R. Amin, H. Aldabbas, D. Koundal, B. Alouffi, T. Shah, Machine learning techniques for spam detection in email and IoT platforms: Analysis and research challenges, *Security and Communication Networks*. (1) (2022), 1-19. doi:10.1155/2024/7538203

[20] Z. B. Siddique, M. A. Khan, I. U. Din, A. Almogren, I. Mohiuddin, S. Nazir, Machine learning-based detection of spam emails, *Scientific Programming.* (1) (2021), 1-11. doi:10.1155/2021/6508784

[21] Z. Hassani, V. Hajihashemi, K. Borna, I. S. Dehmajnoonie, A classification method for E-mail spam using a hybrid approach for feature selection optimization, *Journal of Sciences, Islamic Republic of Iran*. 31(2), (2020), 165-173.

[22] A. Sheneamer, Comparison of deep and traditional learning methods for email spam filtering, *International Journal of Advanced Computer Science and Applications (IJACSA).* 12(1) (2021), 560-565. doi: 10.14569/IJACSA.2021.0120164

[23] S. Zavrak, S. Yilmaz, Email spam detection using hierarchical attention hybrid deep learning method, *Expert Systems with Applications.* 233 (2023), 120977. doi: 10.1016/j.eswa.2023.120977

[24] G. Hnini, J. Riffi, M. A. Mahraz, A. Yahyaouy, H. Tairi, MMPC-RF: a deep multimodal feature-level fusion architecture for hybrid spam E-mail detection, *Applied Sciences*. 11(24) (2021), 11968. doi: 10.3390/app112411968

[25] A. I. Taloba, S. S. Ismail, An intelligent hybrid technique of decision tree and genetic algorithm for e-mail spam detection, *In: IEEE 2019 Ninth International Conference on Intelligent Computing and Information Systems (ICICIS)*, *IEEE*, Cairo, Egypt, 2019: 99-104. doi:10.1109/ICICIS46948.2019.9014756

[26] K. Meena, S. R. Upadhyaya, A Privacy-Preserving machine learning ensemble for spam detection, *In: IEEE 5th International Conference on Inventive Research in Computing Applications (ICIRCA)*, *IEEE*, Coimbatore, India, 2023: 255-259.

[27] M. Sumathi, S. Raja, Machine learning algorithm-based spam detection in social networks, *Social Network Analysis and Mining.* 13(1) (2023), 104. doi:10.1007/s13278-023-01108-6

[28] M. Hina, M. Ali, A. R. Javed, F. Ghabban, L. A. Khan, Z. Jalil, SeFACED: Semantic-based forensic analysis and classification of e-mail data using deep learning, *IEEE Access.* 9 (2021), 98398-98411. doi:10.1109/ACCESS.2021.3095730

[29] S. Xu, Y. Li, W. Zheng, Bayesian multinomial naïve bayes classifier to text classification, *In: International Conference on Multimedia and Ubiquitous* Engineering*, Springer,* Singapore 2017: 347–352. doi:10.1007/978-981-10-5041-1_57

[30] R. O. Olanrewaju, S. A. Olanrewaju, L. A. Nafiu, Multinomial naïve bayes classifier: bayesian versus nonparametric classifier approach, *European Journal of Statistics*. 2 (8) (2022), 1-13. doi:10.28924/ada/stat.2.8

[31] U. K. B. Saravanan, M. Vijay, T. Shreedhar, G. Rajasekar, R. Yashwanth, P. Shakthipriya, Multinomial Naive Bayes Based Machine Learning Analysis of Twitter Sentiment, *In: IEEE 2nd International Conference on Edge Computing and Applications (ICECAA)*. Namakkal, India, 2023: 429-434. doi:10.1109/ICECAA58104.2023.10212150.

[32] Y. K. Zamil, S. A. Ali, M. A. Naser, Spam image email filtering using k-nn and svm, *International Journal of Electrical and Computer Engineering.* 9(1) (2019), 245-254. doi:10.11591/ijece.v9i1. 245-254.

[33] B. Trstenjak, S. Mikac, D. Donko, Knn with tf-idf based framework for text categorization, *Procedia Engineering.* 69 (2014), 1356-1364. doi:10.1016/j.proeng.2014.03.129

[34] Z. Yong, L. Youwen, X. Shixiong, An improved knn text classification algorithm based on clustering, *Journal of Computers*, 4(3) (2009), 230-237. doi:10.4304/jcp.4.3.230-237

[35] S. S. Ismail, R. F. Mansour, A. El-Aziz, M. Rasha, A. I. Taloba, Efficient e-mail spam detection strategy using genetic decision tree processing with NLP features, *Computational Intelligence and Neuroscience*. (2022), 1-16. doi:10.1155/2022/7710005.

[36] D. Jurafsky, J. H. Martin, Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models, (2024). https://web.stanford.edu/~jurafsky/slp3/5.pdf (accessed 21 September 2024).

[37] S. Jamshidi, M. Mohammadi, S. Bagheri, H. E. Najafabadi, A. Rezvanian, M. Gheisari, M. Ghaderzadeh, A. S. Shahabi, Z. Wu, Effective text classification using BERT, MTM LSTM, and DT. *Data & Knowledge Engineering*. 151 (2024), 102306. doi:10.1016/j.datak.2024.102306.

[38] Y. Yu, X. Si, C. Hu, J. Zhang, A review of recurrent neural networks: LSTM cells and network architectures, *Neural Computation.* 31 (7) (2019), 1235-1270. doi:10.1162/neco a 01199

[39] A. Purwarianti, I. A. P. A. Crisdayanti, Improving Bi-LSTM performance for Indonesian sentiment analysis using para44 graph vector, *In: IEEE 2019 International Conference of Advanced Informatics: Concepts, Theory and Applications (ICAICTA),* Bandung, Indonesia, 2019: 1-5. doi:10.1109/ICAICTA.2019.8904199.

[40] Y. Xiong, N. Wei, K. Qiao, Z. Li and Z. Li, Exploring Consumption Intent in Live E-Commerce Barrage: A Text Feature-Based Approach Using BERT-BiLSTM Model, *IEEE Access*, 12 (2024), 69288-69298. doi: 10.1109/ACCESS.2024.3399095.

[41] J. Wallat, F. Beringer, A. Anand, V. Anand, Probing BERT for Ranking Abilities. *In: Kamps, J., et al. Advances in Information Retrieval. ECIR 2023. Lecture Notes in Computer Science*, Springer, Cham, 2024: 13981. doi:10.1007/978-3-031-28238-6_17

[42] B. Aytan, C. O. Sakar, Comparison of transformer-based models trained in turkish and different languages on turkish natural language processing problems, *In: 2022 30th Signal Processing and Communications Applications Conference (SIU),* Safranbolu, Turkey, 2022: 1-4. doi:10.1109/SIU55565.2022.9864818

[43] E. Corp, W. W. Cohen. Enron Email Dataset, (2015). https://www.loc.gov/item/2018487913/ (accessed 23 September 2024).

[44] H. Simsek, E. Aydemir. Classification of unwanted e-mails (spam) with turkish text by different algorithms in weka program, *Journal of Soft Computing and Artificial Intelligence,* 3 (2022), 1-4. doi:10.55195/jscai.1104694

[45] UCI Machine Learning Repository. Turkish Spam V01 dataset, (2019). https://archive.ics.uci.edu/dataset/530/turkish+spam+v01 [accessed 15 December 2023].

[46] W. Qader, M. Ameen, B. Ahmed, An overview of bag of words;importance, implementation, applications, and challenges, *In: 2019 International Engineering Conference (IEC)*, Erbil, Iraq, 2019: 200-204. doi:10.1109/IEC47844.2019.8950616

[47] L. Havrlant, V. Kreinovich, A simple probabilistic explanation of term frequency-inverse document frequency (tf-idf) heuristic (and variations motivated by this explanation), *International Journal of General Systems*. 46 (2017), 27-36. doi:10.1080/03081079.2017.1291635

[48] A. Jalilifard, V. F. Carida, A. F. Mansano, R. S. Cristo, F. P. C. Fonseca, Semantic sensitive tf-

idf to determine word relevance in documents, *In: Advances in Computing and Network Communications,* 2021: 327–337. doi:10.1007/978-981-33-6987-0

[49] F. Zhang, W. Song, Product improvement in a big data environment: A novel method based on text mining and large group decision making, *Expert Systems with Applications*, 245 (2024), 123015, doi:10.1016/j.eswa.2023.123015.

[50] J. Pennington, R. Socher, C. Manning, Glove: global vectors for word representation, *In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP),* Doha, Qatar, 2014: 1532–1543. doi:10.3115/v1/D14-1162

[51] Z. Hua, Y. Tong, Y. Zheng, Y. Li, and Y. Zhang, PPGlove: Privacy-Preserving Glove for Training Word Vectors in the Dark, *IEEE Transactions on Information Forensics and Security*. 19 (2024), 3644-3658. doi:10.1109/TIFS.2024.3364080

[52] P. Bountakas, C. Xenakis, HELPHED: Hybrid ensemble learning phishing email detection, *Journal of Network and Computer Applications*. 210 (2023), 103545. doi:10.1016/j.jnca.2022.103545

[53] O. Sagi, L. Rokach, Ensemble learning: a survey, *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery.* 8 (4) (2018), e1249. doi:10.1002/widm.1249

[54] G. Wang, J. Sun, J. Ma, K. Xu, J. Gu, Sentiment classification: the contribution of ensemble learning, *Decision Support Systems.* 57 (2014), 77-93. doi:10.1016/j.dss.2013.08.002

[55] M. A. Ganaie, M. Hu, A. K. Malik, M. Tanveer, P. N. Suganthan, Ensemble deep learning: a review, *Engineering Applications of Artificial Intelligence*. 115 (2022), 105151. doi:10.1016/j.engappai.2022.105151.

[56] N. C. Yang, K. L. Sung, Non-intrusive load classification and recognition using soft-voting ensemble learning algorithm with decision tree, k-nearest neighbor algorithm and multilayer perceptron, *IEEE Access*. 11 (2023), 94506-94520. doi:/10.1109/ACCESS.2023.3311641

[57] A. Ghourabi, M. Alohaly, Enhancing spam message classification and detection using transformer-based embedding and ensemble learning, *Sensors.* 23(8) (2023), 3861. doi:10.3390/s23083861