Spam Detection for Turkish Short Message Service (SMS)

Zeynep Çakmak
Istanbul Kultur University
Computer Engineering Department
Istanbul/Turkey
zeynepcakmak37@hotmail.com

Mehmet Salih Çifçi
Istanbul Kultur University
Computer Engineering Department
Istanbul/Turkey
m.salih_cifci@hotmail.com

***Abstract*— Short Message Service (SMS) is a mobile messaging tool for individuals to communicate, and billions of people use their phones to send and receive messages. Due to the lack of suitable message filtering techniques, this sort of communication is insecure. Machine learning and deep learning methods are applied to detect the spam SMS. In this study, we will suggest a spam detection approach for categorization of ham and spam communications based on machine learning methods. Different methods will be applied to detect spam such as K Nearest Neighbors, Support Vector Machines, Naïve Bayes, Decision Trees. The confusion matrix, which will be constructed for each strategy, is the most common way for evaluating the performance of models derived from data sets with predetermined target data in machine learning. To compare machine learning algorithms, accuracy, precision, recall, and f-measure will be examined. According to the values we obtained as a result of our study, we achieved the highest accuracy values in Support Vector Machine (99.03%) with the 1st preprocessing and in Logistic Regression and Random Forest (98.07%) with the 2nd preprocessing.**

Keywords—sms classification, machine learning, spam, deep learning

I. INTRODUCTION

Many people use SMS in mobile communication. SMS has great importance in people's lives. However, undesirable situations may occur in incoming messages. The most important of these are spam messages. These spam messages are usually in the promotion category. Messages sent by many banks, betting, holiday and similar companies are classified as spam. SMS Spam Classification has been made to prevent these situations. Thanks to this application, the messages sent to the mobile phone are divided into 2 sections by understanding whether they are spam or not. Spam ones are detected through the algorithm and are not shown to the user. Thus, information pollution is avoided.

In this study, it is aimed to classify SMS spam using two different Turkish datasets. Messages are divided into 2 categories, ham and spam, by means of machine learning techniques, passing through the improvement stages such as data preprocessing. In this study, accuracy values were calculated using K-Nearest Neighbors, Multinomial Naive Bayes, Logistic Regression, Support Vector Machines, Random Forest, Decision Tree Classifier, Ada Boost Classifier, XGBoost Classifier, Stochastic Gradient Descent, Artificial Neural Network, Convolutional Neural Network and Long Short-Term Memory.

In this work, it is aimed to prevent people from being disturbed by spam messages. The rest of this paper is regularized as follows: The summaries of other articles related to our project are presented in the second part, the explanation of the datasets, the algorithms and methods used are explained in the third part; and the comparison of the classification results are given in the fourth part. The discussion and results are shown in the fifth and sixth parts.

II. RELATED WORK

Onur Karasoy and Serkan Ballı classified Turkish SMS as spam and ham by using ML and DL algorithms in their work in 2021 [2]. Turkish SMS records are a collection of messages from people of different ages and regions. Turkish SMS records have 5 particular constitutional properties, 2 new properties created using WordTwoVec and fourty-five properties produced using the word index value of every SMS. Consequently, the algorithm that gave the highest accuracy value was the convolutional neural network with 99.9% accuracy. D.Suhartono, T.K.Prasetyo, A.Theodorus and R.Hartono created a model to categorize spam, raw, promotional dispatches grounded on Indonesian dispatches in their work in 2021 [1]. The model was trained with message 4.125, and tested with message 1.260. 10 fold cross validation procedure was used to measure the classifiers and the outcomes indicate MLR(93.57%), XGBoost (93.52%), SVM(93.38%) and RF(93.62%). In the work of M. Arulprakash and K. R. Jansi in 2021, [3] the structure of SMS transmission is fundamentally splited into two layers; first one is the Access Layer and the second of all is the Service Provider Layer. SMS spam detection filters and techniques are used in either of these two layers. As a result, the accuracy values in grouping raw and spam messages are Naïve Bayes algorithm(95.2%), SVM(88%), KNN(85%) and Decision Tree(83%). In the study of Kemal Ozkan, Yildiray Anagun, Zuhal Kurt and Sahin Isik in 2020, [4] different deep learning methods are implemented to 2 feature selection methods. First of all, feature selection methods inclusive of MI and WMI are implemented to decrease the number of terms chosen. Second of all, the feature vectors were created using the B.O.W Model. After all, the productivity of the system was measured using BILSTM,

LSTM and ANN models. When the results were interpreted, we observed a competition between the detection results of MI and WMI use. Experimental results show that combining LSTM and BILSTM with MI or WMI gives 100% accurate results for spam and legitimate email. However, for certain cross validation, WMI's performances is higher than the M.I's specification when it comes to email grouping. Given the high detection scores, WMI and MI with DL architectures have proven to be more robust against spam email detection. Özlem ÖRNEK, in her study in 2020, [5] used TurkishSMS message and UCI SMS Spam collections in the study, spam and ham classification was made for SMS with Turkish and English content. It's determined to provide the topnotch algorithm by using the Orange 3 application for spam messages classification and identification. Based on the accuracy and error rates, it has been determined that Neural Networks (98.4%) algorithm for TurkishSMS dataset and Naive Bayes (98.4%) algorithm for UCI SMS Spam dataset have great accuracy and less miss rate. Dima Suleiman, Ghazi Al-Naymat and Mariam Itriq in their work in 2020, [6] the main part of the research is to suggest a new message detection classifier propped up using the H2O platform. Three classifiers are used these are RF, DL, and NB. Two validation models, TEN-fold and THREE-fold cross validation, were used. As a result, 10x cross validation significantly improves accuracy, recall, precision, and FMeasure results. Looking at the run time, we found that triple the experiment was optimal. As a result, it showed that the RF is the excellent classifier with accuracy rate %0.977. Looking at the accuracy of Naive Bayes, it is seen that it is the worst, but in terms of running time, it gives the best result with a value of 0.6 seconds. In the study of J.P.Singh, S.Banerjee, and P.K.Roy in 2020, [7] CNN and LSTM models were used. It recorded 55.79% accuracy with Naïve Bayes (NB), 51.67% accuracy with Stochastic Gradient Descent (SGD), 90.21% accuracy with Gradient Boosting (GB), 86.70% accuracy with Random Forest (RF) and 56.27% accuracy with Logistic Regression (LR). Next, we tested two different deep learning models CNN and LSTM. As a result, the CNN model achieved an accuracy of 98.44%, confirming that it better the LSTM model. Ö.F.Ertuğrul and Y.Kaya in their work in 2016, [13] The LTP image processing method was developed to subtract features from the message during the feature extraction phase. In the recommended 1DTP, the text message was first converted to a UTF8 value. Then each letter in the SMS was compared to the adjacent letter. 2 distinct property sets were subtracted from the outcomes of these contrasts. In the end, some ML procedures are used to categorize these properties. Three SMS spam structures were used to interpret and verify the 1DTP. The 94.10%, 93.318%, and 87.15% accuracy obtained show that the recommended approach can be successfully used for message filtering feature extraction, rather than a statistical method of aggregating the frequency of occurrence of letters and words. In the work of H.Sajedi, F.Akbari, and G.Z.Parast in 2016, [11] [11] it is a study prepared to test 15 different algorithms on the most common SMS datasets on the internet from 5 different countries and obtain the results. In fact, this article is the most advanced study on SMS Spam Classification. Because a wide range of results was obtained by using 15 different algorithms on 5 different countries. As a result, by examining the accuracy values of 15 different algorithms, it is seen that Dendritic Cell Algorithm (DCA) reached the highest value with 99.95%. In the study of M.P.Patil, S.R. Kolhe and Ajay U. Surwade in 2016, [12] Indian researchers have studied Spam Mail. Three different datasets were used for this research: Personal mail, Enron, and LingSpam. NB, SVM, and Semantic Similarity with Edge Based Classifier(SSC) are used as algorithm for two component Machine Learning Based Classifier (MLC). The results of the algorithms used are concluded in 5 separate sections as 1-Spam Precision, 2-Spam Recall, 3 Accuracy, 4-False Positive, and 5-False negative. As a result, in the Enron dataset, 96.36% accuracy values were obtained with NB, 97.68% and 99.9% accuracy values were obtained with SVM in LingSpam and PEM datasets, respectively. With SSC, accuracy values of 94.5% were obtained in Enron, 90.5% in LingSpam and 99% in PEM. Cüneyt Özdemir and Yılmaz Kaya in their work in 2018, [9] a new approach to Spam Sms has been developed and a new approach study has been made through motif patterns. In this study, 3 different datasets, VS1 VS2 VS3, were studied. Success rates of 93.76%(LR), 90.07%(LR) and 94.29%(RF) were observed for the three Datasets, respectively. According to the observed results, the proposed method was found to be an accomplished property extraction procedure from SMS in spam filtering. In this study, different ML methods such as Aggregating One-Dependence Estimators (A1DE), ANN, SVM, LR, RF, and Functional Tree were used. SMSs are classified as spam or non-spam with machine learning methods using motif patterns. In the work of Dr. Kavita S. Oza and Dr. Dipak R. Kawade in 2018, [10] they worked on a single dataset. Numerous feature extraction and feature selection methods are used in the article to identify Spam SMS in Turkish and English languages. The model framework has taken into account features and structural features in the Bag of Words for identification purposes. Based on different text features, they applied 4 different classification algorithms and then combined them to create a collaborative algorithm. In the study, 4 different classification algorithms, namely SVM, NB, NN, and Relevance Vector Machine were used for Spam SMS classification. Experimental performance is counted based on different training dataset size and extracted feature size. As a result of the experiment, it has been shown that the NN algorithm is not suitable for Spam SMS identification. The performance of SVM and Relevance Vector Machine algorithms was good but between these two algorithms, Relevance Vector Machine is faster than SVM. The results of the study shows that the Relevance Vector Machine algorithm is most suitable for filtering Spam SMS. The current algorithm correctly identifies 732 spam SMS. The percentage of correctly matched Spam SMS is 98.12% and unmatched Spam SMS is 1.88%. In the study of Xuemin Chen and Tian Xia in 2020, [8] two different datasets, English and Chinese, were used. Machine learning methods such as Vector Space Model, NB, SVM, LSTM, and CNN were used for these datasets. B.O.W model and Hidden Markov model (HMM) were tested. Looking at the results of the experiments and Hidden Markov model (HMM) showed us that it was the highest Spam SMS capture model in both datasets and proved that it was not language sensitive. The Hidden Markov model (HMM) achieved an accuracy rate of 98.5% in the Chinese dataset.

III. METHODOLOGY

A. *Overview of the Dataset*

Our first dataset named SmsTurkish [14] consists of FourHundred-Thirty ham and FourHundred-Twenty spam messages collected from different people. As seen in figure 3.1, 50.6% of the messages are ham and 49.4% are spam messages in the dataset. This dataset is the fundamental Turkish dataset in articles written about sms spam classification.

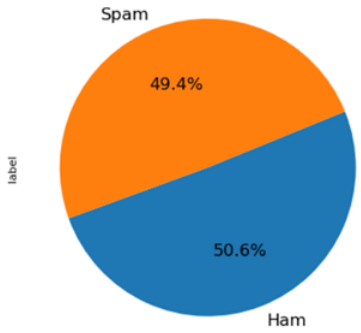

Figure 3.1: smsTurkish dataset

Our second dataset, named TurkishSMSCollection [2], consists of 2536 spam and 2215 ham messages. As seen in figure 3.2, 46.6% of the messages are ham and 53.4% are spam messages in the dataset. While creating the dataset, messages from different living areas and age groups were collected. SMS's were gathered from seventy-six they are living in diverse provinces of Turkiye, in Ankara, Istanbul and Muğla.

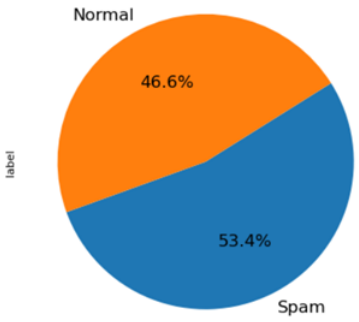

Figure 3.2: turkishSmsCollection dataset

By combining the SmsTurkish and TurkishSMSCollection datasets, we created a third dataset called BigTurkishSms. This dataset includes 2956 spam 2645 ham messages. As seen in figure 3.3, 47.2% of the messages are ham and 52.8% are spam messages in the dataset.

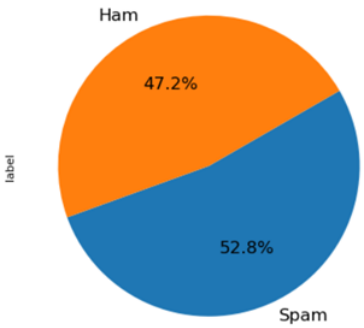

Figure 3.3: bigTurkishSms dataset

B. *Tools and Technology*

1)Machine Learning and Deep Learning

ML is the process of extracting knowledge from the past and using that knowledge to make future predictions. DL is a type of ML with more functions by reason of it tries to imitate the neurons of the human brain. Attempts to learn the phenomenon as a nested hierarchy of concepts. To each one is defined by a simpler concept.

The ML system consists of three main parts:

Model: A system that makes predictions or identifications.

Parameters: The signal or factor that the model uses to make the decision.

Learner: A system that adjusts parameters, or models, by seeing the difference between predicted and actual results.

ML is divided into mainly two types, which are:

Supervised ML: The machine is trained with a "labeled" dataset, and the device predicts output based on the training. A flagged date indicates that some input has been assigned to the production environment. That is, train the machine with the information and the corresponding output, and ask the machine to predict the outcome.

The main purpose of the supervised learning method is to match the input variable (x) to the output variable (y). Examples include risk assessment, fraud detection, spam filtering, and other real-world supervised learning applications.

Unsupervised ML: UL, as the name implies, differs from SL in that it does not require a teacher. In UML, machines are trained on unlabeled datasets and predict output for free.

The main goal of UL algorithm is to group or categorize unsorted datasets based on similarities, patterns, and differences. The machine is instructed to find a hidden path in the input dataset.

2)NB

NB is one of the supervised learning algorithm depending on Bayes theorem and used to solve classification difficulties. It is principally used for text classification using higher dimensional training dataset. The NB is one of the simplest and most efficient classification algorithm for building prompt ML models that can make fast estimations. The most common variations are known as sentiment analysis and spam filtering.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

3)SVM

The SVM algorithm is one of the supervised learning algorithms used primarily in classifications. The purpose of the SVM is to easily categorize new data points that will occur in the future and generate optimal judgment boundaries that can classify n dimensional space. These decision boundaries are called HyperPlane. Support Vector Machine is divided into 2; they are used for linear and non-linear data. The most common variations are known as image classification and text classification and face detection.

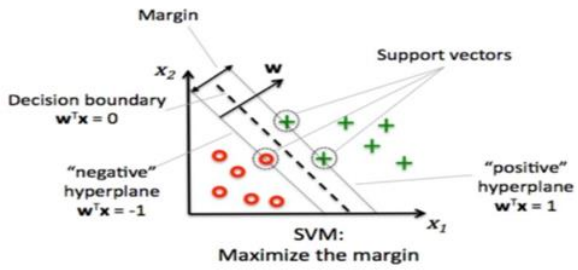

Figure 3.4: SVM

4)RF

RF algorithm is one of the common ML algorithm, which is a supervised learning method. This falls under the notion of community learning, which is used to reach the solution of a complex problem and improve model performance. Random Forest does not depend on a single result or decision but evaluates multiple predictions and obtains a final result. The

more predictions in RF, the higher accuracy margin. Less training time is required compared to other algorithms. If it works correctly, it will get the right result regardless of the size of the data set.

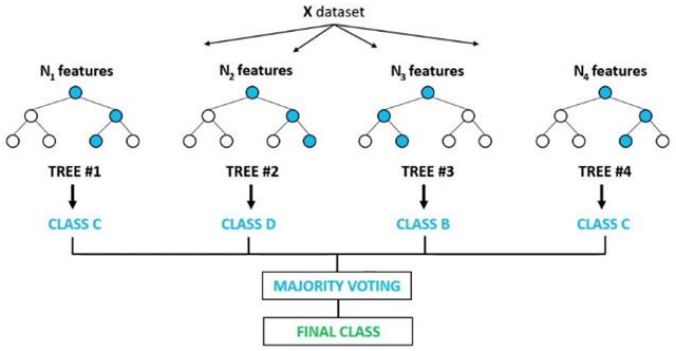

Figure 3.5: RF

5) LR

LR algorithm is one of the most common algorithm in supervised ML techniques. This algorithm helps us to find the categorically dependent variable using a certain non-dependent dataset. In Logistic Regression, the results vary between 0 and 1. In this algorithm, the concept of predictive modeling and numerical data are prioritized.

6) KNN

KNN is considered the simplest algorithms used in the supervised ML technique. KNN algorithm is depending on the relationship parity between the data in the current state and categorizes the new state data according to the parity rates. The KNN algorithm is non parametric and doesn't make presumes on basic data. The KNN algorithm performs operations on the dataset it categorizes.

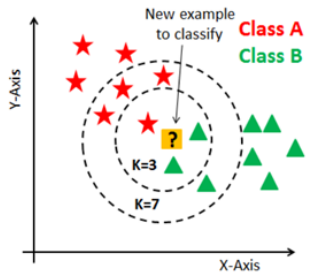

Figure 3.6: KNN

7) DT

DT Classification is one of the supervised learning algorithms used primarily in classification difficulties. DT can effortlessly process combinations of numerical and categorical properties, and can categorize data that lacks properties as well. The DT algorithm divides the data into decision nodes and creates a tree construction. The decision node symbolizes the reference question that further divides the data into two or more child nodes. The tree is built until the datas of a particular child node is impeccable.

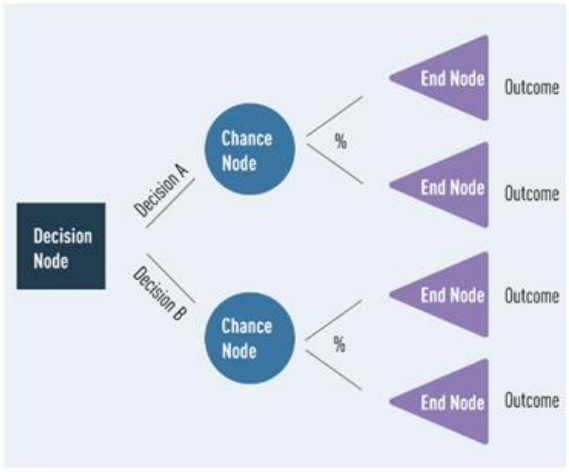

Figure 3.7: DT

8) AdaBoost

AdaBoost, which stands for "Adaptive Boosting", is ML algorithm referred by R.Schapire and Y.Freund. It works on the element that learners grow in turn. With the exception of the initial learner, each posterior learner grew up from a formerly adult learner. Simply put, a feeble learner turns into a powerful learner. AdaBoost studies on the identical element as boosting, with one subtle variation.

9) XGBoost

The XGBoost or Extreme Gradient Boosting algorithms use a transaction named boosting to enhance productivity. This is a DT based ML algorithm. To overcome overfitting, veritably common difficulty when training models, XGBoost uses more regulated model format to control overfitting and improve productivity. This algorithm was improved to operatively decrease calculation time with high model productivity.

10) SGD

SGD is a veritably common algorithm used in a variety of ML algorithms, especially forming the base of NN. SGD randomly selects several samples for each replication, rather than the entire dataset.

11) ANN

ANNs are networks that correlate the inputs and outputs of a system, and these networks have been successfully used to map nonlinear input and output connections in different domains. ANNs are used to model nonlinear problems and provide the output values of specific input parameters from training values. ANNs take delivery of input signals from the outside world in the form of patterns and images in the form of vectors. Then each input is multiplied by its own weight.

12) CNN

CNNs are neural networks with one or further convolutional layers and are primarily used for classification, image processing, segmentation and other autocorrelation data. Like NNs, CNNs are made up of neurons with learnable heaviness and strains. Each neuron take delivery of multiple inputs, takes a heavy sum across them, passes them through an activation function and replies with an output.

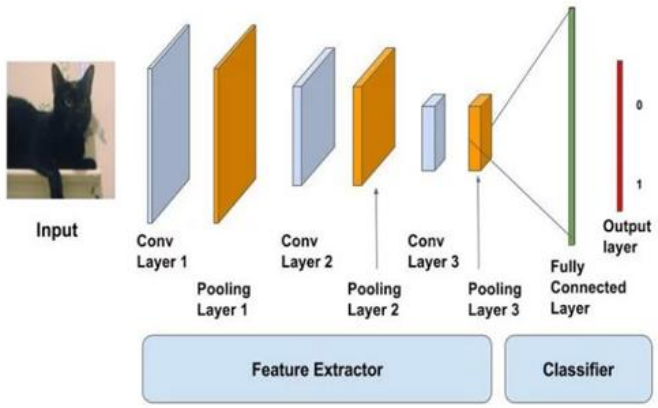

Figure 3.8: CNN

13) LSTM

LSTM nets are RNN armatures capable of learning long term dependences. LSTM act like a RNN cell. At a high level, The Long Short Term Memory cells consist of 3 sections; The first of the LSTM chooses whether to remember the information from the former timestamp or to forget it without using it. Secondly, the cell learns new information from the input. Finally, where the cell propagates the streamlined information from the extant timestamp to the coming one.

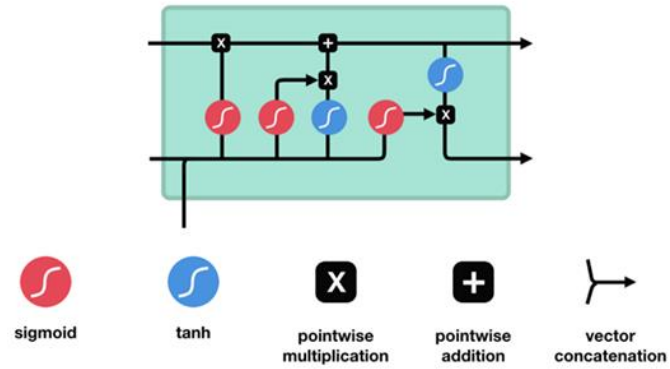

Figure 3.9: LSTM

The above 3 sections of the LSTM cell are the gates. Primary section is forget gate, while secondary is the input gate, and third section is output gate.

C. Proposed Approach

1)Data Preprocessing

a. First Preprocessing

Firstly, we transform all the messages into lowercase. Afterwards we remove http, www, .com, email address, link, punctuation, digits and also delete white spaces from the text.

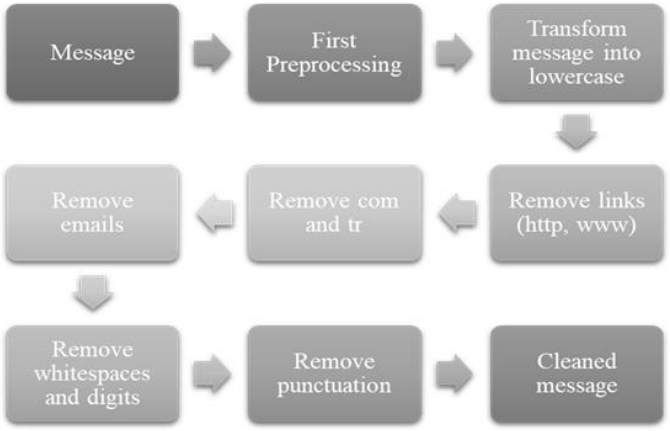

Figure 3.10: First preprocessing steps

b. Second Preprocessing

When we applied the first preprocessing, we get a low accuracy value in the KNN algorithm. At the same time, our LSTM model did not give high results. For this reason, we created a new preprocess that splits the data into words.

In the preprocess, first we remove non alphabetic characters and transform all the messages into lowercase. Secondly, we split the message data into words using the word tokenize function. We delete stopwords in the messages. In the end, we found the first 50 most repeated words in the dataset using the bag of words.

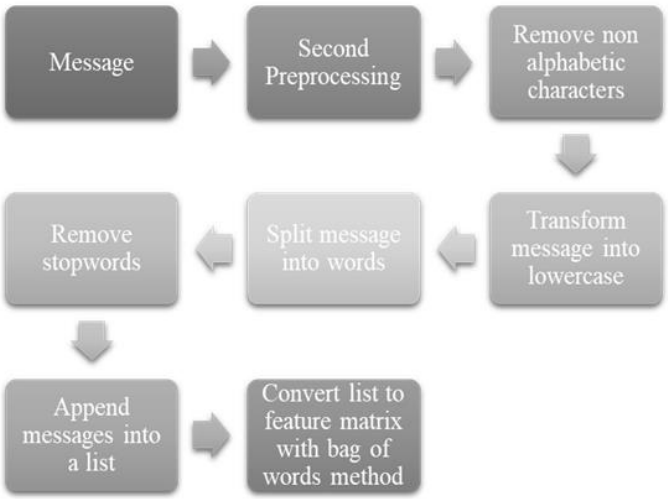

Figure 3.11: Second preprocessing steps

2)Cross Validation

Cross validation is statistical resampling method used to measure the productivity of the ML model on data it does not see, as objectively and accurately as possible.

Procedure takes a one variable called k. This shows the count of groups that need to divide a significant data instance. For this reason, this method is often cited as K fold cross validation.

Cross validation is firstly used in administered ML and uses hidden data to predict the functionality of ML models. In other words, a limited instance is used to predict how a model may ordinarily act if it is used to make estimates of data that wasn't used while training of the model.

3)Confusion Matrix

Confusion matrix is a productivity indicator for ML classification difficulty, and the output can be in further than one class. It is a table of four distinct combining estimated and real values.

Table 3.1: Confusion matrix

		Actual Values: Positive (1)	Actual Values: Negative (0)
Predicted Values	Positive (1)	TP	FP
Predicted Values	Negative (0)	FN	TN

It is excessive practical for evaluating Accuracy, Precision, Recall and F1 Score.

TP: If prediction is positive and it's true.
TN: If prediction is negative and it's true.
FP: If prediction is positive and it's false.
FN: If prediction is negative and it's false.

Accuracy: From all the classes, how many of them we have predicted correctly.

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

Precision: From all the classes we have predicted as positive, how many are actually positive.

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

Recall: From all the positive classes, how many we predicted correctly.

$$Recal = \frac{TP}{TP + TN}$$

F1 Score: F-score helps to measure Recall and Precision at the same time. It uses Harmonic Mean in place of Arithmetic Mean by punishing the extreme values more.

$$F1\ Score = \frac{Precision * Recall}{Precision + Recall} * 2$$

IV. EXPERIMENTAL RESULTS

A. Comparing Algorithms

In Smsturkish dataset we used first preprocessing method and we showed accuracy, precision, recall and F1 scores of the algorithms in the table 4.2. SGD is the algorithm that gives the best performance with an accuracy value of 96%. On the other hand, KNN gave the lowest performance with a value of 73%.

Table 4.2: Comparison of algorithms in the first preprocessed Smsturkish dataset

	Accuracy	Precision score	Recall score	F1 score
KNN	0.7353	0.7385	0.7353	0.7338
RandomForest	0.9412	0.9422	0.9412	0.9412
Logistic Regression	0.9412	0.9422	0.9412	0.9412
Multinomial Naive Bayes	0.9412	0.9472	0.9412	0.9409
Support Vector Machine	0.9471	0.9477	0.9471	0.9471
Decision Tree Classifier	0.8588	0.8762	0.8588	0.8576
Ada Boost Classifier	0.9471	0.9487	0.9471	0.9471
XGBoost Classifier	0.9235	0.9266	0.9235	0.9235
Stochastic Gradient Descent	0.9647	0.9649	0.9647	0.9647

In Smsturkish dataset we used second preprocessing method and we showed accuracy, precision, recall and F1 scores of the algorithms in the table 4.3. RF and MNB are the algorithms that gives the best performance with an accuracy values of 94%. On the other hand, DT gave the lowest performance with a value of 87%.

Table 4.3: Comparison of algorithms in the second preprocessed Smsturkish dataset

	Accuracy	Precision Score	Recall Score	F1 Score
KNN	0.8941	0.8999	0.8941	0.8938
RandomForest	0.9470	0.9475	0.9470	0.9470
Logistic Regression	0.9294	0.9304	0.9294	0.9294
Multinomial Naive Bayes	0.9470	0.9475	0.9470	0.9470
Support Vector Machine	0.9411	0.9413	0.9411	0.9411
Decision Tree Classifier	0.8705	0.8813	0.8705	0.8699
Ada Boost Classifier	0.9352	0.9353	0.9352	0.9353
XGBoost Classifier	0.9352	0.9358	0.9352	0.9353
Stochastic Gradient Descent	0.9235	0.9236	0.9235	0.9235

In TurkishSMSCollection dataset we used first preprocessing method and we showed accuracy, precision, recall and F1 scores of the algorithms in the table 4.4. SGD is the algorithm that gives the best performance with an accuracy value of 98%. On the other hand, KNN gave the lowest performance with a value of 66%.

Table 4.4: Comparison of algorithms in the first preprocessed TurkishSMSCollection dataset

	Accuracy	Precision Score	Recall Score	F1 Score
KNN	0.6635	0.7474	0.6635	0.6101
RandomForest	0.9284	0.9322	0.9284	0.9278
Logistic Regression	0.9747	0.9757	0.9747	0.9748
Multinomial Naive Bayes	0.9747	0.9756	0.9747	0.9746
Support Vector Machine	0.9831	0.9837	0.9831	0.9832
Decision Tree Classifier	0.9158	0.9196	0.9158	0.9161
Ada Boost Classifier	0.9663	0.9664	0.9663	0.9663
XGBoost Classifier	0.9674	0.9678	0.9674	0.9674
Stochastic Gradient Descent	0.9863	0.9867	0.9863	0.9863

In TurkishSMSCollection dataset we used second preprocessing method and we showed accuracy, precision, recall and F1 scores of the algorithms in the table 4.5. Ada Boost is the algorithm that gives the best performance with an accuracy value of 98%. On the other hand, MNB gave the lowest performance with a value of 66%.

Table 4.5: Comparison of algorithms in the second preprocessed TurkishSMSCollection dataset

	Accuracy	Precision Score	Recall Score	F1 Score
KNN	0.9652	0.9667	0.9652	0.9653
RandomForest	0.9821	0.9821	0.9821	0.9821
Logistic Regression	0.9737	0.9740	0.9737	0.9737
Multinomial Naive Bayes	0.6624	0.7640	0.6624	0.6039
Support Vector Machine	0.9821	0.9821	0.9821	0.9821
Decision Tree Classifier	0.9463	0.9495	0.9463	0.9465
Ada Boost Classifier	0.9842	0.9842	0.9842	0.9842
XGBoost Classifier	0.9779	0.9779	0.9779	0.9779
Stochastic Gradient Descent	0.9810	0.9810	0.9810	0.9810

In BigTurkishSms dataset we used first preprocessing method and we showed accuracy, precision, recall and F1 scores of the algorithms in the table 4.6. SVM is the algorithm that gives the best performance with an accuracy value of 98%. On the other hand, KNN gave the lowest performance with a value of 69%.

Table 4.6: Comparison of algorithms in the first preprocessed BigTurkishSms dataset

	Accuracy	Precision Score	Recall Score	F1 Score
KNN	0.6975	0.7482	0.6975	0.6726
RandomForest	0.9393	0.9422	0.9393	0.9390
Logistic Regression	0.9803	0.9807	0.9803	0.9803
Multinomial Naive Bayes	0.9741	0.9751	0.9741	0.9740
Support Vector Machine	0.9857	0.9859	0.9857	0.9857
Decision Tree Classifier	0.9072	0.9129	0.9072	0.9073
Ada Boost Classifier	0.9634	0.9643	0.9634	0.9634
XGBoost Classifier	0.9607	0.9611	0.9607	0.9607
Stochastic Gradient Descent	0.9848	0.9849	0.9848	0.9848

In BigTurkishSms dataset we used second preprocessing method and we showed accuracy, precision, recall and F1 scores of the algorithms in the table 4.7. RF and Ada Boost are the algorithms that gives the best performance with an accuracy values of 97%. On the other hand, MNB gave the lowest performance with a value of 64%.

Table 4.7: Comparison of algorithms in the second preprocessed BigTurkishSms dataset

	Accuracy	Precision Score	Recall Score	F1 Score
KNN	0.9536	0.9575	0.9536	0.9536
RandomForest	0.9759	0.9761	0.9759	0.9759
Logistic Regression	0.9661	0.9673	0.9661	0.9661
Multinomial Naive Bayes	0.6440	0.7630	0.6440	0.5830
Support Vector Machine	0.9696	0.9697	0.9696	0.9696
Decision Tree Classifier	0.9295	0.9344	0.9295	0.9295
Ada Boost Classifier	0.9759	0.9763	0.9759	0.9759
XGBoost Classifier	0.9705	0.9709	0.9705	0.9705
Stochastic Gradient Descent	0.9705	0.9706	0.9705	0.9705

B. Cross Validation

In Smsturkish dataset, we used KNN, MNB, LR, SVM, RF, DT, Ada Boost Classifier, XGBoost Classifier, SGD, ANN, CNN and LSTM algorithms. We showed the scores we obtained by giving the values of 3, 5 and 10 to the cross validation in the table 4.8. Except for Ada Boost, when the cross validation value is 10, we get higher results in all other algorithms.

Table 4.8: Algorithms in smsturkish dataset according to cv values

	cv=3	cv=5	cv=10
KNN	0.5888	0.6426	0.6981
RandomForest	0.9672	0.9603	0.9726
Logistic Regression	0.9656	0.9716	0.9722
Multinomial Naive Bayes	0.9548	0.9545	0.9676
Support Vector Machine	0.9661	0.9710	0.9770
Decision Tree Classifier	0.9156	0.9147	0.9172
Ada Boost Classifier	0.9561	0.9589	0.9545
XGBoost Classifier	0.9406	0.9445	0.9483
Stochastic Gradient Descent	0.9396	0.9441	0.9649
ANN	0.9543	0.9762	0.9941
CNN	0.9078	0.9385	0.9588
LSTM	0.4983	0.5189	0.5353

To evaluate the performance of the algorithms in other datasets, we performed the same operations on a dataset of English SMS messages. We showed the scores we obtained by giving the values of 3, 5 and 10 to the cross validation in the table 4.9. Except for Ada Boost and MNB, when the cross validation value is 10, we get higher results in all other algorithms.

Table 4.9: Algorithms in english dataset according to cv values

	cv=3	cv=5	cv=10
KNN	0.8024	0.8277	0.8315
RandomForest	0.9395	0.9459	0.9479
Logistic Regression	0.9424	0.9480	0.9592
Multinomial Naive Bayes	0.8909	0.8841	0.8825
Support Vector Machine	0.9347	0.9518	0.9548
Decision Tree Classifier	0.7949	0.7893	0.8070
Ada Boost Classifier	0.8846	0.9003	0.8944
XGBoost Classifier	0.9324	0.9339	0.9448
Stochastic Gradient Descent	0.9389	0.9457	0.9475
ANN	0.9448	0.9683	0.9776
CNN	0.9378	0.9591	0.9767
LSTM	0.4918	0.5029	0.5348

C. KNN testing accuracy

Looking at the results, we reached lower values in the KNN algorithm compared to the others. Therefore, we examined the effect on the accuracy value by changing the number of neighbors in the KNN. We reached the highest value by giving neighbors a value of 3.

Table 4.10: Comparing KNN according to neighbors values

	Testing Accuracy
n_neighbors = 3	0.7352
n_neighbors = 4	0.6882
n_neighbors = 5	0.7235
n_neighbors = 6	0.6352
n_neighbors = 7	0.6588
n_neighbors = 8	0.5882
n_neighbors = 9	0.6176
n_neighbors = 10	0.5411

D. Final Comparison

We tested 12 algorithms on 3 different datasets. Since we had two different preprocesses, we compared the cross validation scores under 2 different headings for each dataset.

When 1st preprocessing was applied to the Smsturkish dataset, we obtained the highest value with ANN (99.41%). When the 2nd preprocessing was applied, we obtained the highest value with SVM (96.60%).

When the 1st preprocessing was applied to the SMSCollection dataset, we obtained the highest score with the SVM (99.24%). When the second preprocessing was applied, we obtained the highest value with RF (98.85%).

When the 1st preprocessing was applied to the BigDataset dataset, we obtained the highest score with SVM (99.03%). When the second preprocessing was applied, we obtained the highest values with LR and RF (98.07%).

Table 4.11: Final comparison of algorithms

	1.preprocess	2.preprocess	1.preprocess	2.preprocess	1.preprocess	2.preprocess
	Smsturkish (cv=10)	Smsturkish (cv=10)	Turkish SMS Collection veri seti	Turkish SMS Collection veri seti	Smsturkish + Turkish SMS Collection (bigDataset)	Smsturkish + Turkish SMS Collection (bigDataset)
KNN	0.6981	0.9436	0.6459	0.9790	0.6942	0.9767
RandomForest	0.9726	0.9537	0.8977	0.9885	0.8972	0.9807
Logistic Regression	0.9722	0.9606	0.9892	0.9878	0.9888	0.9807
Multinomial Naive Bayes	0.9676	0.9472	0.9874	0.6501	0.9786	0.6506
Support Vector Machine	0.9770	0.9660	0.9924	0.9863	0.9903	0.9744
Decision Tree Classifier	0.9172	0.9127	0.9573	0.9779	0.9477	0.9620
Ada Boost Classifier	0.9545	0.9464	0.9787	0.9848	0.9785	0.9788
XGBoost Classifier	0.9483	0.9521	0.9802	0.9868	0.9760	0.9801
Stochastic Gradient Descent	0.9649	0.9505	0.9915	0.9870	0.9901	0.9749
ANN	0.9941	0.9412	0.9632	0.9853	0.9563	0.9741
CNN	0.9588	0.9412	0.9579	0.9832	0.9572	0.9715
LSTM	0.9000	0.9294	0.9769	0.9779	0.5987	0.9634

V. CONCLUSIONS AND FUTURE WORK

In this project, we will suggest a spam detection approach for classification of ham and spam based on machine learning methods. In our project, we applied two different preprocessing techniques to 3 different datasets. We compared the accuracy values of 12 algorithms with these datasets. We tested the accuracy rates of these models using cross validation. When the 1st preprocessing was applied, we compared the three datasets and obtained the highest value with ANN(99.41%). When the 2nd preprocessing was applied, we compared the three datasets and obtained the highest value with RF(98.85%).

REFERENCES

[1] Theodorus, T. K. Prasetyo, R. Hartono and D. Suhartono, "Short Message Service (SMS) Spam Filtering using Machine Learning in Bahasa Indonesia," 2021 3rd East Indonesia Conference on Computer and Information Technology (EIConCIT), 2021, pp. 199-203

[2] Karasoy, O., Ballı, S. Spam SMS Detection for Turkish Language with Deep Text Analysis and Deep Learning Methods. Arab J Sci Eng (2021).

[3] Arulprakash, M. "Eshort Message Service Spam Detection and Filtering Using Machine Learning Approach." Turkish Journal of Computer and Mathematics Education (TURCOMAT) 12.9 (2021): 721-727.

[4] Isik, Sahin, et al. "Spam E-mail Classification Recurrent Neural Networks for Spam E-mail Classification on an Agglutinative Language." International Journal of Intelligent Systems and Applications in Engineering 8.4 (2020): 221-227.

[5] Örnek, Özlem. "Orange 3 ile Türkçe ve İngilizce SMS Mesajlarında Spam Tespiti." Eskişehir Türk Dünyası Uygulama ve Araştırma Merkezi Bilişim Dergisi 1.1 (2019): 1-4.

[6] Suleiman, Dima, Ghazi Al-Naymat, and Mariam Itriq. "Deep SMS Spam Detection using H2O Platform." International Journal 9.5 (2020).

[7] Roy, Pradeep Kumar, Jyoti Prakash Singh, and Snehasish Banerjee. "Deep learning to filter SMS spam." Future Generation Computer Systems 102 (2020): 524-533.

[8] Xia, Tian, and Xuemin Chen. "A discrete hidden Markov model for SMS spam detection." Applied Sciences 10.14 (2020): 5011.

[9] Yılmaz, K. A. Y. A., and Cüneyt Özdemir. "Spam SMS'lerin filtrelenmesinde yeni bir yaklaşım: Motif örüntüler." Gazi Üniversitesi Fen Bilimleri Dergisi Part C: Tasarım ve Teknoloji 6.2: 436-450.

[10] Kawade, Kavita Oza, and Kavita S. Oza. "Content-based SMS spam filtering using machine learning technique." International Journal of Computer Engineering and Applications 7 (2018): 4.

[11] Sajedi, Hedieh, Golazin Zarghami Parast, and Fatemeh Akbari. "Sms spam filtering using machine learning techniques: A survey." Machine Learning Research 1.1 (2016): 1-14.

[12] Surwade, Ajay U., M. P. Patil, and S. R. Kolhe. "Effective and adaptive technological solution to block spam E-mails." 2016 International Conference on Advances in Human Machine Interaction (HMI). IEEE, 2016.

[13] Kaya, Yilmaz, and Ömer Faruk Ertuğrul. "A novel feature extraction approach in SMS spam filtering for mobile communication: one-dimensional ternary patterns." Security and communication networks 9.17 (2016): 4680-4690.

[14] Uysal, A. K., Gunal, S., Ergin, S., & Sora Gunal, E. (2013). The Impact of Feature Extraction and Selection on SMS Spam Filtering. Elektronika Ir Elektrotechnika, 19(5), 67-72.https://doi.org/10.5755/j01.eee.19.5.182