Journal of Soft Computing and Artificial Intelligence

Journal homepage: https://dergipark.org.tr/en/pub/jscai

International Open Access

Volume 03
Issue 01

June, 2022

Research Article

Classification of Unwanted Emails (Spam) with Turkish Text by Different Algorithms in the Weka Program

***Hüseyin Şimşek*[1], *Emrah Aydemir*[2*]**

[1,2] *Management Information Systems, Institute of Business. Sakarya University, 54000, Sakarya, Turkey*

ARTICLE INFO

Article history:
Received **April 17, 2022**
Revised **April 25, 2022**
Accepted **May 1, 2022**

Keywords:
E-mail
Classification Algorithm
Spam email
Weka

ABSTRACT

Recently, with the widespread use of the internet, electronic communication tools have also been widely used. One of these tools is emails. Emails are easy to use and provide the opportunity to reach thousands of people at the same time. This advantage causes some bad uses. Email users are faced with dozens of unsolicited emails (spam) against their will. In this study, 1017 mails collected from about 20 different Gmail and Hotmail accounts were classified as spam or regular email using the algorithms in the Weka program, and the success of the algorithms was compared. In the study, 45 different algorithms were tested. The highest classification success was obtained with the Naive Bayes Multinominal and Naive Bayes Multinominal Updateable algorithms with 94.7886% correct classification. Among other classifier algorithms, Random Forest algorithm 93.6087%, Multi-Class Classifier and SGD 92.4287%, SMO 91.7404%, Random Committee 91.0521%, Naive Bayes and Naive Bayes Updateable 90.3638% classification success.

1. Introduction

One of the basic needs of people is communication. Communication is the sharing of feelings, thoughts, ideas, and information between people. Today, new communication tools have emerged with the development of knowledge and internet technologies. One of them is emails that provide electronic communication and communication. Email is the adaptation of classical mailboxes to the electronic environment. The electronic mail system is inspired by correspondence, one of the communication tools used in the past and today and is reflected in the electronic environment with the development of today's internet technology [1]. An email address can be personal or corporate. Email service providers such as Gmail, Hotmail, Yahoo are available. An email address is created in the format "nickname@domainname" [2]. Text, audio, visual, video, file, etc. contents can be shared easily with emails. They are easy to use and meager cost. In addition, it is a great convenience that content can be transmitted to thousands of different people or [1] institutions simultaneously.

The ease of use of emails and the ability to reach thousands of people simultaneously has brought some disadvantages. At the top of these disadvantages are unsolicited (spam) messages. The abuse of electronic messaging systems to send random, unsolicited emails is called spam [3]. Thanks to the cost and speed advantage, emails are used for purposes such as advertising, promotion, marketing, creating public opinion, sharing inappropriate content, and obtaining personal information by sending malicious software, and dozens of spam emails fall into their mailboxes every

* Corresponding author
e-mail: emrahaydemir@sakarya.edu.tr
DOI: 10.55195/jscai.1104694

day. According to the results of a study, approximately 269 billion emails were sent and received worldwide in 2017, 281 billion emails in 2018, and 293 billion emails in 2019 [4]. While this causes a waste of time and effort for users, it also causes unnecessary occupation of network traffic. In addition, from the point of view of enterprises, it is seen that it causes enormous financial losses.

Many different methods and techniques are used to filter unwanted emails, and successful results are obtained. Despite this, they continue to use email systems by developing new strategies to overcome the filters applied to spam email users. For this reason, it is essential to carry out recent studies in this field, develop different methods and techniques, create different data sets, and support the analyses.

This study aims to contribute to the spam filtering studies and the literature by identifying the most successful algorithms in the Bayes, Trees, Meta, Lazy, Functions, Misc., Rules classifiers in the WEKA program by using a Turkish data set collected from different email addresses.

2. Mail System and Spam Mail

Electronic mail (email) is the name given to an electronic message, usually in the form of a simple text message, that a user writes on a computer system and transmits to another user who can read it over a computer network [5]. Email messages consist of a header and a body. The title contains the sender (From), recipient user's ID (To), subject header (Subject), date (Date), received (Received), and content number (Message-ID). There is the content of the message in the body part and the part where attachments (Attachment) will be made [2]. Simple Mail Transport Protocol-SMTP protocol is used for the transmission of emails.

Spam emails are messages sent in bulk by people or bot accounts that are not known. These can also be defined as messages sent to the accounts against the will of the person. Unwanted emails are used for purposes such as advertising, promotion, and propaganda. When we open email addresses, we come across dozens of advertising messages every day, and most of them come from addresses we do not know. In addition, some spam messages can send viruses to capture our personal information and bank account information. They can get our information by copying trusted web addresses and making us trust them. Another reason why we are faced with spam messages today is due to the email trade. Email addresses belonging to thousands of people are marketed to different businesses, and they cause us to receive spam messages from companies we do not know. While businesses are always looking for ways to communicate with their customers more accessible, cheaper, or faster, the internet offers all three [6]. In this case, the marketing of email addresses is one of the reasons for the increase in the number of spam emails.

When examining spam messages, we can list some of their features as follows [2]. The same content is sent to multiple recipients.

- They are sent for promotional purposes.
- Often their content is misleading.
- They may talk about religious beliefs or human feelings and may want the email to be forwarded to many people.
- Address information such as sender, who is not in a proper format, and letter errors are standard as random fakes are usually produced.
- Email message header information is destroyed, making it difficult to trace back.
- Recipients do not have a valid or functional return address to indicate that they do not wish to receive email from this distribution.
- In general, their content is up to date

Today, many different methods are used, and new methods are being developed to filter unsolicited (spam) emails. Some of these methods are Word filtering, Rule-Based Filtering, Blacklists, DNS MX Record Lookup, Reverse DNS Lookup, So Reverse DNS Lookup Honeypots (Honeypots), Hashing Systems, Antivirus Scanning, Fingerprinting (fingerprint), Challenge-Response (challenge) systems and Bayesian filters [5].

3. Aim and Contribution

Today, although technological developments bring great convenience to our lives, they also bring some negativities along with these conveniences. The email has entered our lives with the development of internet technology and has brought a different dimension to communication. Many data such as interpersonal information, documents, pictures, and audio files can be shared quickly and inexpensively via emails. Since emails are a fast and low-cost communication tool, we encounter unsolicited emails, and we are faced with situations such as

endangering people's time, workforce, or personal information. Unnecessary occupancy of network traffic is another problem.

To avoid these problems, it is of great importance to develop new methods, test existing methods with different data sets, and determine successful strategies. This study it is aimed to test different classification algorithms in the Weka program and to choose the most successful classification method by using a data set with Turkish content that has not been used before. In addition, it is thought that it is essential that the more different data are used, the more successful the fight against spam will be.

4. Literature Review

When the literature on filtering and classification of spam is examined, it is seen that methods such as Spam classification with Machine Learning and Word Set technique, Phishing email detection with Deep Learning Models, text mining applications, Decision trees, Bayesian Classifiers, artificial immune system, and spam filtering are examined. In addition, a new approach based on Binary Patterns, filtering methods such as Word2Vec, Support Vector Machines are used. It has been observed that generally successful results have been obtained in the studies carried out.

In the study called Filtering Spam Emails Using the Bayesian Method in 2006, 2387 emails with Turkish content were used. Two different models were tested with the Bayesian method, and it was seen that the first model was classified as spam emails at a rate of 81%, 92%, and 84%, and 93.2%, respectively [5]. In the study conducted within the scope of SMTP Protocol and Spam Mail Problem, DNSBL technique was used, and it was seen that many mails could escape from DNSBL [3]. Tekeli and Aşlıyan analyzed a data set in the UCI machine learning repository in the Weka program in their studies on the detection of spam emails with the multi-layered Perceptron, KNN, and C4.5 Methods and obtained a 92.8% successful classification with the C4.5 algorithms [7].

Cahide Ünal and İsmail Şahin designed a rule-based expert system for the detection of unsolicited emails and aimed to detect spam emails over content and IP addresses. In the study, a data set consisting of 4601 emails obtained from the Hewlett-Packard laboratory was used, and a total of 57 features were extracted from this data set. The developed Expert system examines the emails according to these 57 features and gives feedback to the user about whether the email is a slur [8]. Nazlı Nazlı tested the Word2Vec vector and SVM(Poly) algorithm on a dataset of 300 emails in her Machine Learning-Based Spam Filtering Techniques study and achieved 98.33% successful results [9].

A new spam filtering approach, using binary patterns obtained by comparing the UTF-8 values of characters with each other by Kaya and Özdemir, who tried to detect spam with a new system based on scrolling binary patterns, shifted one-dimensional local binary patterns has been suggested. A proposed C1W-LBS method is a statistical approach based on low-level information obtained because of comparisons of each value on the signal with its neighbors. A benchmark (spamassian) and a dataset created by us were used to test our method. According to the results obtained, it has been seen that the proposed method is a successful method for feature extraction from text-based emails. 92.34% success was achieved in the filtering performed using the Weka Program [10].

Çıtlak, Doğru, and Dörterler In a Spam Detection System Study with Short Links, it has been determined that websites marked as spam in the Google Safe Browsing database can hide by using short link services. In the study, temporary link addresses were first listed, and then software was developed that converts quick link addresses to long web addresses. In the software made, these addresses are automatically checked whether they are spam or not spam in the Google Safe Browsing data set [11].

In the study titled an Analysis of Various Algorithms for Text Spam Classification and Clustering Using RapidMiner and Weka, NB, SVM, KNN algorithms were tested using Weka and RapidMiner programs. In the study, in the tests made using 5572 messages in the UCI Machine Learning database, the NB algorithm in the Weka Program was 94.56%, the SVM algorithm 98.21, the KNN algorithm 94.80% accurate classification success, while the RapidMiner program made the NB algorithm 84.79, SVM algorithm 96.64 and KNN algorithm 94. It was observed that 74 successful classifications were made. The study shows that the Weka Program makes better predictions than RapidMiner [12].

In 2017, the algorithm's success was tested using the Naive Bayes algorithm using two different data sets in Malaysia. In the study, 9324 Spam datasets collected from various email addresses and SPAMBASE dataset consisting of 4601 emails taken from UCi machine learning database were used. Five hundred features were extracted for Spam dataset, and 58 features were extracted for Spabase dataset.

In the results of the analysis, it was seen that the Naive Bayes algorithm made 91.13% correct predictions for the Spam dataset and 82.54% for the Spambase dataset, and it was seen that the collection of the data set from many different sources increased the percentage of correct predictions [13].

Eryılmaz and Kılıç examined the methods used for the detection of spam emails, basically examined artificial intelligence-based and non-artificial intelligence-based filtering techniques, and that non-artificial intelligence-based filtering methods (blacklist, whitelist, gray list, content review, etc.) can be passed, they stated that they have negative sides such as constant updating. Previously, non-AI-based methods would have been effective. But spam detection has improved as a result of the increase in machine learning algorithms. Thus, artificial intelligence-based systems have become more used [4].

Aman Kumar used 4601 email datasets from the UCI machine learning database to compare algorithms for spam filtering. 1813 pieces of data constitute spam emails. The data has a total of 58 features, 57 continuous and one nominal. As a result of the test, it was seen that the J48 Algorithm was the highest correct classification with 92.7624%, and the other algorithms were followed by CART with 92.632%, ADTree with 90.915%, and ID3 with 89.111%, respectively. According to the test results, it has been determined that the J48 algorithm is more successful than CART, ADTree, and ID3 in spam classification [14]. Again, in a similar study, Naive Bayes, Bayesnet, J48, and LAZY-IBK algorithms were compared, and it was seen that the J48 Algorithm was more successful than other algorithms, with a success rate of 85.06% [15].

While many different algorithms are being developed for spam blocking, spammers continue to send spam with new solutions. One of the methods used for spam is to send the message by embedding the content in the picture. When the content is embedded in the image, it does not get caught in the content-based spam filters, and the users continue to receive unwanted messages. In his study, which used 180 data sets, 60 from Google, 60 from Flickr, and 60 from spam, for the detection of spam messages containing images during the daytime, he extracted the histograms of the images and classified the spammy images at a rate of 81%. When the histogram of spammy images is examined, it has been determined that they usually contain few colors, and the value of '0' is relatively high for colors that are not used in the histogram [16].

Looking at the studies, it is clearly seen that artificial intelligence-based machine learning algorithms give successful results in classifying spam emails. Despite this, it is seen that spammers continue to send unsolicited emails with new solutions. In addition, it is seen that the studies are generally studied on English data sets. For this reason, the creation of new data sets and the use of data sets in different languages are important in combating spam.

5. Material and Method

5.1 Data Collection

A total of 1017 emails were collected to be tested in this study. While 502 of the emails were regular mails, 517 of them were spam emails. The data set does not consist of any readymade data set but consists of emails sent to and from 20 different email addresses. Only the title and content parts of the emails with Turkish content were included in the data set, and each email was recorded separately in a text file. For regular mails, the names are norm1.txt, norm2.txt, …….. norm502. spam1, spam2, ….., spam517 filenames are saved as spam. The dataset here has been uploaded publicly to its address so that it can be used by other researchers (https://www.kaggle.com/datasets/emrahaydemr/turkish-mail-dataset-normalspam).

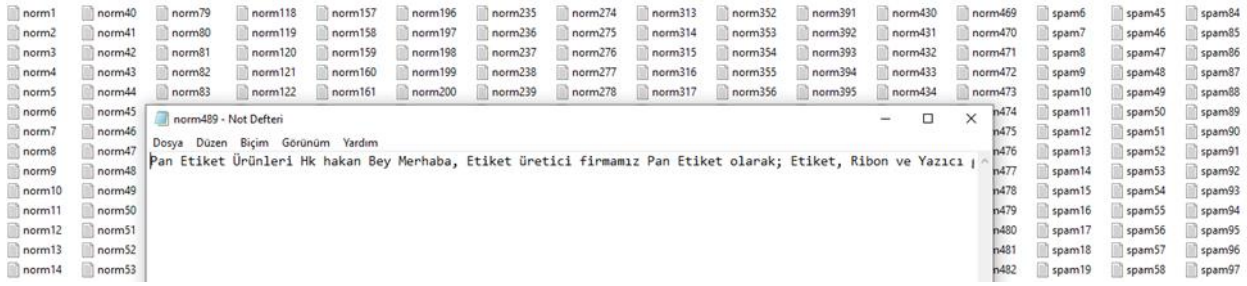

Figure 1 Created Dataset Pool

5.2 Data Analysis

During the analysis of the data, some operations were carried out in order to process the data in the Weka program. At the beginning of these processes, the data set collected in separate txt files was converted into a single norm_spam.txt file using the Python programming language, and punctuation marks, memorable characters, and numbers in the text were cleaned from the text. The created text file has been converted into a format that the Weka program can analyze as an arff file.

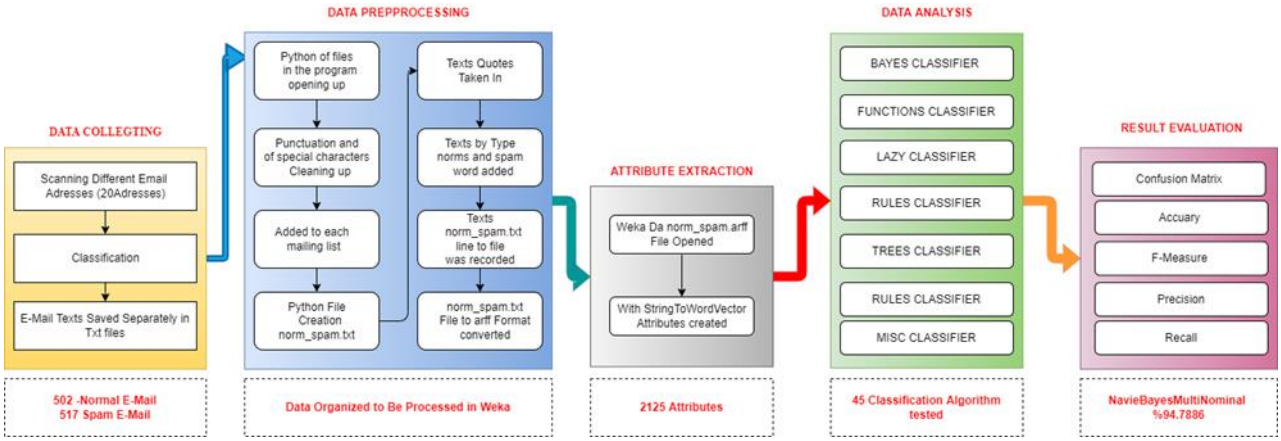

Figure 2 Email Analysis Workflow Chart

In the continuation of these processes, the norm_spam.arff file we created first was opened in the Explorer window of the Weka program with the OpenFile tab, and the attributes of the data were extracted with the StringToWordVector filter from the filter tab. A total of 2125 word vectors belonging to the data set were extracted as features.

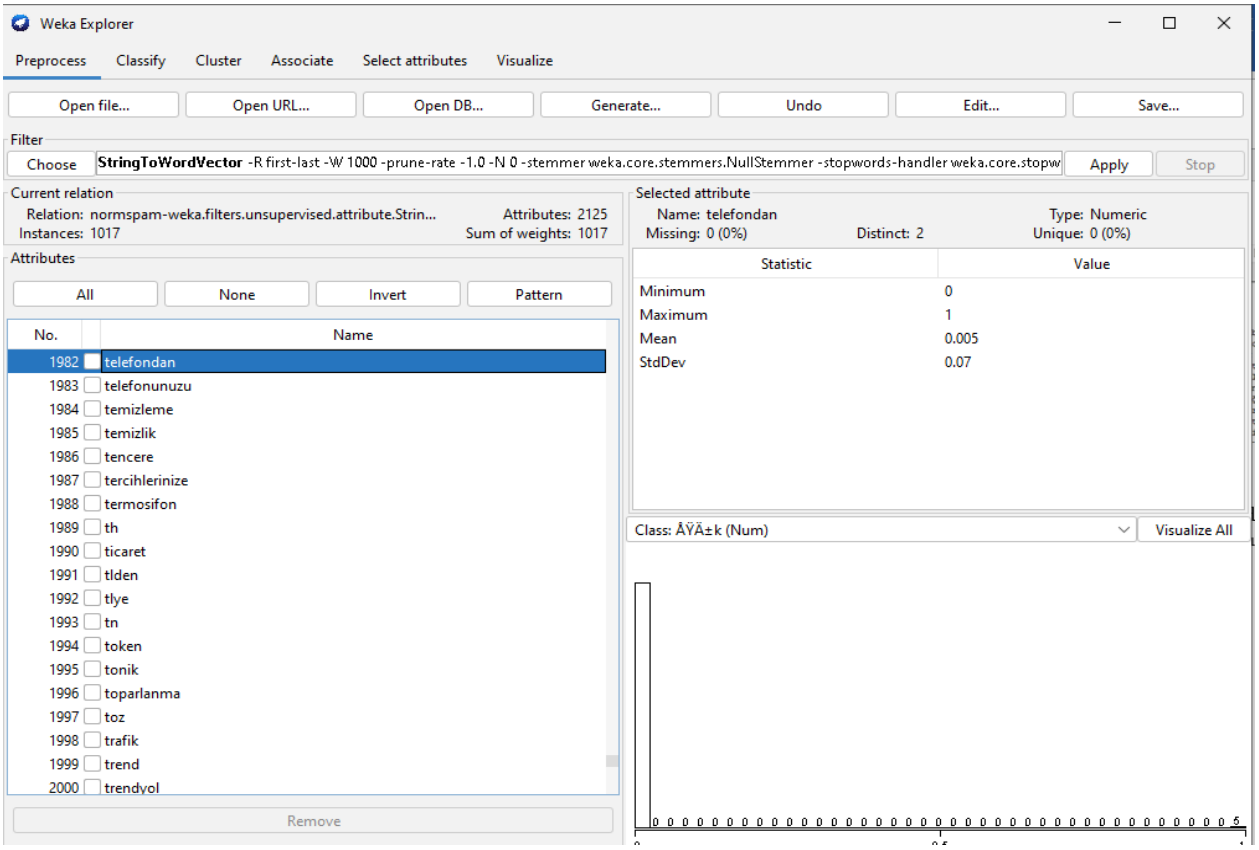

Figure 3 Attribute Extraction Screen

5.2.1 Weka Program

Aydemir introduced the Weka Program in the Artificial Intelligence Book with Weka as follows. Weka is a software developed at the University of Waikato in New Zealand and is licensed under the GNU and GPL. That is, it is software that is available to the public, free and open source. It is named after the initials of the phrase "Waikato Environment for Knowledge Analysis". The name of this Java-based program is also the name of a flightless and endangered bird found in the islands of New Zealand. This software, which is used for data mining and machine learning applications, contains almost all frequently used algorithms. It has been produced in order to be able to try on datasets quickly by using the existing methods themselves. In addition to this feature, it allows the analysis of the results. Through the program, basic data mining operations such as classification, clustering, and association can be performed in general. It can run on all systems, especially Linux, Windows, and Macintosh [17].

5.2.2 String to Word Vector

Word vectors simply focus on the relationships between words. Semantic analyzes are made based on the relationships of these words [18]. StringToWordVector Generates a numeric attribute showing the frequencies of the words in the String data type [17]. For natural language processing algorithms to understand text, texts must be represented as numbers. There are standard methods for this. The StringToWordVector filter used in the Weka program also converts texts into numerical vectors with techniques such as TF-IDF (Term Frequency — Inverse Document Frequency) and n-gram. With the help of this filter, the number of terms in the text or data set is counted and their frequency is revealed. For example, if a word occurs 10 times in the text and the dataset contains 1000 texts, then the value of 10/1000, which is 0.01, is created for this word. In addition, the reverse document frequency determines how vital the searched word is. In other words, if the number of data in which the number of words searched is 10, and it is included in a total of 3 data, then the log(10/3) value of 0.52 is obtained. With this filter in Weka, word count, rooting, converting to lowercase, etc. It also provides features.

5.2.3 Algorithms Used in Data Analysis

For the analysis of the data, 45 different algorithms in the Classify window of the Weka program were tested, and the results were given in the findings section. The information on the classification algorithms used before proceeding to the findings is as follows.

- *Bayes Classifier:* Bayesian classifier is a classification algorithm that is used a lot in machine learning studies because it is fast and has a high success rate. Bayesian classifier is based on Bayes' theorem introduced by Thomas Bayes in 1763 [19]. This theorem, which deals with the events to be classified independently, predicts which class the data belong to [19]. Bayesian filters are one of the widely used methods of spam classification. To determine the probability that an email is a spam, filters use Bayesian analysis to compare the frequency of words or phrases in the email in previous (regular and spam) emails of the relevant user [5].
- *Naive Bayes Algorithm:* The logical foundations of the Naive Bayes algorithm are based on the approaches introduced by Thomas Bayes in the 18th century [20]. It is possible to have an idea about the direction of their content by analyzing the numerous news in the media or social media through text mining. Naive Bayes algorithm is one of the algorithms that can be used for this purpose [20].
- *Decision Trees:* Decision Trees in data mining are one of the most preferred methods because they are cheap to create, can be easily integrated with data systems, are safe, easy to interpret, and have a high comprehensibility [21]. When we look at the structure of decision trees, they consist of roots, branches, and leaves [21]. It resembles a tree with its structure. Decision trees that start with the root node divide many datasets into small groups and branches as they go down [21]. In decision trees, the first node is called the root node, the other nodes are called the leaf node, and the last is the decision node [22].
- *Lazy (Lazy Algorithms):* Lazy classifiers store the training samples and do no real work until it is time to classify [23]. The simplest lazy learning algorithm is the k-nearest neighbor classifier called IBk[17].
- *Meta Heuristics Algorithms:* The high-level heuristic approach includes methods that perform a probabilistic but conscious search in the solution space. These methods produce new solutions based on the solution set created at each step. Thus, by doing searches at the points close to the most suitable

one in the search space, it is tried to reach the most appropriate solution by getting rid of the local best point selection [24].Rules Algorithms: The rule inference system (RULES) family is an inductive learning family that includes several overlay algorithms. This family is used to construct a predictive model based on the given observation. It works based on the concept of dividing and conquers to create rules and knowledge pools directly from a specific training set [25].

5.2.4 Success Criteria

The table where we can interpret the classification successes of the algorithms, we used in the research in an understandable way is the confusion matrix. We can compare the actual values with the estimated values with the confusion matrix. In the data set we used, we divided regular emails into two separate classes as norm spam and spam. The confusion matrix will allow us to interpret the results of the algorithm in 4 classes.

Table 1 Confusion Matrix

		Prediction	
		Regular	**Spam**
Real	**Regular**	**TP**	**TN**
	Spam	**FN**	**FP**

- **True Positive (TP):** Indicates the number of correctly classified emails that are actually regular emails.
- **True Negative (TN):** The number of emails classified as spam even though it is actually regular email.
- **False Negative (FN):** Number of spam emails classified as regular email even though they are actually spam.
- **False Positive (FP):** Shows the number of emails that are actually spam emails and are correctly classified as spam by the algorithm used.

6. Findings

After the dataset's attributes we used were extracted with StringToWordVector in the Weka Program Preprocess window, 45 algorithms were tested with the generally accepted 10-fold cross-validation (Cross-Validation Folds 10) method in classification processes in the Classify window, and the findings are given in the tables below.

Table 2 Findings

Classifier	Algorithm	Confusion Matrix	Accuracy	Precision	TP rate	FP rate	F-Measure	Recall	Class
BAYES	Bayes Net	a b <-- classif 437 65 | a = norm 72 443 | b = spam	86.529	0,859	0,871	0,140	0,864	0,871	Norm
				0,872	0,865	0,129	0,866	0,860	Spam
	Naive Bayes	a b <-- classif 465 37 | a = norm 61 454 | b = spam	90,3638	0,884	0,926	0,118	0,905	0,926	Norm
				0,925	0,882	0,074	0,903	0,882	Spam
	Naive Bayes Multinominal	a b <-- classif 477 25 | a = norm 28 487 | b = spam	94,7886	0,945	0,950	0,054	0,947	0,950	Norm
				0,951	0,946	0,050	0,948	0,946	Spam
	Naive Bayes Multinominal Text	a b <-- classi 0 502 | a = norm 0 515 | b = spam	50,6391	?	0,000	0,000	?	0,000	Norm
				0,506	1,000	1,000	0,672	1,000	Spam
	Naive Bayes Multinominal Updateable	a b <-- classif 477 25 | a = norm 28 487 | b = spam	94,7886	0,945	0,950	0,054	0,947	0,950	Norm
				0,951	0,946	0,050	0,948	0,948	Spam
	Naive Bayes Updateable	a b <-- classif 465 37 | a = norm 61 454 | b = spam	90,3638	0,884	0,926	0,118	0,905	0,926	Norm
				0,925	0,882	0,074	0,903	0,882	Spam
FUNCTIONS	Logistic	a b <-- classifi 427 75 | a = norm 65 450 | b = spam	86,234	0,868	0,851	0,126	0,859	0,851	Norm
				0,857	0,874	0,149	0,865	0,874	Spam
	Simple Logistic	a b <-- classif 432 70 | a = norm 42 473 | b = spam	88,9872	0,911	0,861	0,82	0,885	0,861	Norm
				0,871	0,918	0,139	0,894	0,918	Spam
	SMO	a b <-- classif 456 46 | a = norm 38 477 | b = spam	91,7404	0,923	0,908	0,074	0,916	0,908	Norm
				0,912	0,926	0,092	0,919	0,926	Spam
	Voted Perceptron	a b <-- classif 427 75 | a = norm 61 454 | b = spam	86,6273	0,875	0,851	0,118	0,863	0,851	Norm
				0,858	0,882	0,149	0,870	0,882	Spam
	SGD Text	a b <-- classif 0 502 | a = norm 0 515 | b = spam	50,6391	?	0,000	0,000	?	0,000	Norm
				0,506	1,000	1,000	1,000	1,000	Spam
	SGD	a b <-- classif 460 42 | a = norm 35 480 | b = spam	92,4287	0,929	0,916	0,068	0,923	0,916	Norm
				0,920	0,932	0,084	0,926	0,92	Spam
LAZY	IBk	a b <-- classif 373 129 | a = norm 109 406 | b = spam	76,5978	0,774	0,743	0,212	0,758	0,743	Norm
				0,759	0,788	0,257	0,773	0,773	Spam
	KStar		80,0393	0,836	0,741	0,142	0,786	0,741	Norm

		a b <-- classif 372 130 | a = norm 73 442 | b = spam		0,773	0,858	0,259	0,813	0,858	Spam
	LWL	a b <-- classif 496 6 | a = norm 418 97 | b = spam	58,088	0,543	0,988	0,812	0,701	0,988	Norm
				0,942	0,188	0,012	0,314	0,188	Spam
RULES	Decision Table	a b <-- classif 428 74 | a = norm 171 344 | b = spam	75,9095	0,715	0,853	0,332	0,777	0,853	Norm
				0,823	0,668	0,147	0,737	0,668	Spam
	JRip	a b <-- classif 326 176 | a = norm 73 442 | b = spam	75,5162	0,817	0,649	0,142	0,724	0,649	Norm
				0,715	0,858	0,351	0,780	0858	Spam
	OneR	a b <-- classif 454 48 | a = norm 396 119 | b = spam	56,3422	0,534	0,904	0,769	0,672	0,904	Norm
				0,713	0,231	0,096	0,349	0,231	Spam
	PART	a b <-- classif 392 110 | a = norm 76 439 | b = spam	81,7109	0,838	0,781	0,148	0,808	0,781	Norm
				0,800	0,852	0,219	0,825	0,852	Spam
	ZeroR	a b <-- classif 0 502 | a = norm 0 515 | b = spam	50,6391	?	0,000	0,000	?	0,000	Norm
				0,506	1,000	1,000	0,672	1,000	Spam
TREES	Decision Stump	a b <-- classif 496 6 | a = norm 425 90 | b = spam	57,6205	0,539	0,988	0,825	0,697	0,988	Norm
				0,938	0,175	0,012	0,295	0,175	Spam
	Hoeffding Tree	a b <-- classif 0 502 | a = norm 0 515 | b = spam	50,6391	?	0,000	0,000	?	0,000	Norm
				0,506	1,000	1,000	0,672	1,000	Spam
	J48	a b <-- classif 381 121 | a = norm 81 434 | b = spam	80,1377	0,825	0,759	0,157	0,790	0,759	Norm
				0,782	0,843	0,241	0,811	0,843	Spam
	LMT	a b <-- classif 433 69 | a = norm 37 478 | b = spam	89,5772	0,921	0,863	0,072	0,891	0,863	Norm
				0,874	0,928	0,137	0,900	0,928	Spam
	Random Forest	a b <-- classif 466 36 | a = norm 29 486 | b = spam	93,6087	0,941	0,928	0,056	0,935	0,928	Norm
				0,931	0,944	0,072	0,937	0,944	Spam
	Random Tree	a b <-- classif 407 95 | a = norm 85 430 | b = spam	82,3009	0,827	0,811	0,165	0,819	0,811	Norm
				0,819	0,835	0,189	0,827	0,835	Spam
	REP Tree	a b <-- classif 376 126 | a = norm 64 451 | b = spam	81,3176	0,855	0,749	0,124	0,798	0,749	Norm
				0,782	0,876	0,251	0,826	0,876	Spam
META	AdaBoostM1	a b <-- classif 234 268 | a = norm 65 450 | b = spam	67,2566	0,783	0,466	0,126	0,584	0,466	Norm
				0,627	0,874	0,534	0,730	0,874	Spam
	Attribute Selected Classifier	a b <-- classif 381 121 | a = norm 131 384 | b = spam	75,2212	0,744	0,759	0,254	0,751	0,759	Norm
				0,760	0,746	0,241	0,753	0,746	Spam
	Bagging	a b <-- classifi 401 101 | a = norm 47 468 | b = spam	85,4474	0,895	0,799	0,091	0,844	0,799	Norm
				0,822	0,909	0,201	0,863	0,909	Spam
	Classication Via Regression	a b <-- classifi 361 141 | a = norm 87 428 | b = spam	77,5811	0,806	0,719	0,169	0,760	0,719	Norm
				0,752	0,831	0,281	0,790	0,831	Spam
	CV Parameter Selection	a b <-- classif 0 502 | a = norm 0 515 | b = spam	50,6391	?	0,000	0,000	?	0,000	Norm
				0,506	1,000	1,000	0,672	1,000	Spam
	Filtered Classifier	a b <-- classifi 393 109 | a = norm 54 461 | b = spam	83,9725	0,879	0,783	0,105	0,828	0,783	Norm
				0,809	0,895	0,217	0,850	0,895	Spam
	Iterative Classifier Optimizer	a b <-- classifi 351 151 | a = norm 88 427 | b = spam	76,4995	0,800	0,699	0,171	0,746	0,699	Norm
				0,739	0,829	0,301	0,781	0,829	Spam
	Logit Boost	a b <-- classifi 351 151 | a = norm 88 427 | b = spam	76,4995	0,800	0,699	0,171	0,746	0,699	Norm
				0,739	0,829	0,301	0,781	0,829	Spam
	Multi Class Classifier	a b <-- classif 427 75 | a = norm 65 450 | b = spam	86,234	0,868	0,851	0,126	0,859	0,851	Norm
				0,857	0,874	0,149	0,865	0,874	Spam
	Multi Class Classifier Updateable	a b <-- classif 460 42 | a = norm 35 480 | b = spam	92,4287	0,929	0,916	0,068	0,923	0,916	Norm
				0,920	0,932	0,084	0,926	0,932	Spam
	Random Committee	a b <-- classif 467 35 | a = norm 56 459 | b = spam	91,0521	0,893	0,930	0,109	0,911	0,930	Norm
				0,929	0,891	0,070	0,910	0,891	Spam
	Randomizable Filtered Classifier	a b <-- classifi 281 221 | a = norm 177 338 | b = spam	60,8653	0,614	0,560	0,344	0,585	0,560	Norm
				0,605	0,656	0,440	0,629	0,656	Spam
	Random Sub Space	a b <-- classif 429 73 | a = norm 39 476 | b = spam	88,9872	0,917	0,855	0,076	0,885	0,855	Norm
				0,867	0,924	0,145	0,895	0,924	Spam
	Stacking	a b <-- classif 0 502 | a = norm 0 515 | b = spam	50,6391	?	0,000	0,000	?	0,000	Norm
				0,506	1,000	1,000	0,672	1,000	Spam
	Vote	a b <-- classif 0 502 | a = norm 0 515 | b = spam	50,6391	?	0,000	0,000	?	0,000	Norm
				0,506	1,000	1,000	0,672	1,000	Spam
	Weighted Instances Handler Wrapper	a b <-- classif 0 502 | a = norm 0 515 | b = spam	50,6391	?	0,000	0,000	?	0,000	Norm
				0,506	1,000	1,000	0,672	1,000	Spam
	Multi Schema	a b <-- classif 0 502 | a = norm 0 515 | b = spam	50,6391	?	0,000	0,000	?	0,000	Norm
				0,506	1,000	1,000	0,672	1,000	Spam
MISC	Input Mapped Classifier	a b <-- classifi 0 502 | a = norm 0 515 | b = spam	50,6391	?	0,000	0,000	?	0,000	Norm
				0,506	1,000	1,000	0,672	1,000	Spam

7. Conclusion

This study classified regular email and unsolicited email (spam) using various algorithms. 502 regular emails and 517 spam emails collected from different email addresses were tested in the Weka program with 45 different classification algorithms. The highest classification success was obtained with the Naive Bayes Multinominal and Naive Bayes Multinominal Updateable algorithms with 94.7886% correct classification. Among other classifier algorithms, Random Forest algorithm 93.6087%. It is seen from the results that Multi Class Classifier and SGD 92.4287%, SMO 91.7404%, Random Committee 91.0521%, Naive Bayes Updateable 90.3638% classification success. In the study, it was seen that the data set without classification success was clustered in a single class (spam), while the Meta (Stacking, Vote, Weighted Instances Handler Wrapper, Multi Schema, CV Parameter Selection), Hoeffding Tree, Rules Zero R, SGD Text, Naive Bayes Multinominal Text algorithms 50.6391.

When the results are examined, successful results can be obtained by using Random Forest, SMO, Multi Class Clasifier Updateable, and Random Committee Algorithms, where bayesian classifiers show higher success in classifying spam. When the studies on filtering spam emails are examined, it is a fact that although successful results have been obtained, spammers are constantly developing new methods. To continue the struggle with this reality, the creation of more Turkish data sets is of great importance for future studies.

References

[1]. C. Özdemir, M. Ataş, Ve A. B. Özer, "Türkçe İstenmeyen Elektronik Postaların Yapay Bağışıklık Sistemi İle Sınıflandırılması, Signal Processing and Communications Applications Conference (SIU), 2013.

[2]. C. Özdemir, "Yapay bağışıklık sistemi ile spam filtreleme", Master's Thesis, Fen Bilimleri Enstitüsü, 2013.

[3]. M. E. Yüksel ve Ş. D. Odabaşı, "SMTP Protokolü ve Spam Mail Problemi", Akad. Bilişim, 2010.

[4]. E. E. Eryilmaz ve E. Kılıç, "İstenmeyen Epostaların Tespiti için Kullanılan Yöntemlerin İncelenmesi", Dicle Üniversitesi Mühendis. Fakültesi Mühendis. Derg., c. 11, sy 3, ss. 977-987, 2020.

[5]. C. Altunyaprak, "Bayes yöntemi kullanarak istenmeyen elektronik postaların filtrelenmesi", PhD Thesis, Yüksek Lisans Tezi, Muğla Üniversitesi Fen Bilimleri Enstitüsü, 2006.

[6]. Y. Gedik, "E-Posta Pazarlama: Teorik Bir Bakış", Uluslar. Önetim Akad. Derg., c. 3, sy 2, ss. 476-490, 2020.

[7]. K. Tekeli ve R. Aşlıyan, "Çok Katmanlı Algılayıcı, K-NN ve C4. 5 Metotlarıyla İstenmeyen E-postaların Tespiti", Adnan Menderes Üniversitesi, 2016.

[8]. Ü. Cahide ve İ. Şahin, "İstenmeyen Elektronik Postaların (SPAM) Filtrelenmesi için Bir Uzman Sistem Tasarımı ve Gerçekleştirilmesi", Politek. Derg., c. 20, sy 2, ss. 267-274, 2017.

[9]. N. Nazlı, "Analysis of machine learning-based spam filtering techniques", Master's Thesis, 2018.

[10]. Y. Kaya ve C. Özdemir, "Spam filtrelemek için kaydırmalı ikili örüntüler tabanlı yeni bir yaklaşım". XVIII. Akademik Bilişim Conference, 2016.

[11]. O. Çitlak, İ. A. Doğru, ve M. Dörterler, "A spam detection system with short Link Analysis". 10. Uluslararasi Bilgi Güvenliği Ve Kriptoloji Konferansi, Ankara, Türkiye, 20 - 21 Ekim 2017.

[12]. K. Zainal, N. F. Sulaiman, ve M. Z. Jali, "An analysis of various algorithms for text spam classification and clustering using RapidMiner and Weka", Int. J. Comput. Sci. Inf. Secur., c. 13, sy 3, s. 66, 2015.

[13]. N. F. Rusland, N. Wahid, S. Kasim, ve H. Hafit, "Analysis of Naive Bayes algorithm for email spam filtering across multiple datasets", içinde IOP conference series: materials science and engineering, 2017, c. 226, sy 1, s. 012091.

[14]. A. K. Sharma ve S. Sahni, "A comparative study of classification algorithms for spam email data analysis", Int. J. Comput. Sci. Eng., c. 3, sy 5, ss. 1890-1895, 2011.

[15]. G. H. AL-Rawashdeh ve R. B. Mamat, "Comparison of four email classification algorithms using WEKA", Int. J. Comput. Sci. Inf. Secur. IJCSIS, c. 17, sy 2, ss. 42-54, 2019.

[16]. H. C. Gündüz, "Spam 2.0, Tespit ve Engelleme Yöntemleri", s. 6, 2007.

[17]. E.Aydemir, Weka İle Yapay Zeka, 2.Baskı. Ankara: Seçkin, 2019.

[18]. E. Koç, S. Çalışkan, S. A. Yazıcıoğlu, U. Demirci, ve Z. Kuş, "Yapay Sinir Ağları, Kelime Vektörleri ve Derin Öğrenme Uygulamaları", 2018.

[19]. G. AKSOY, "Ağırlıklı Bayes sınıflandırıcıda ağırlıkların optimizasyonu/Optimization of the weights of weighted Naive Bayesian classifier", 2018.

[20]. A. Suat, "KNN, Naive Bayes ve Karar Ağaci Makine Öğrenme Algoritmalari, Bu Algoritmalarin Sosyal Bilimlerde Kullanim Imkânlari", 2020.

[21]. Ş. Demirel ve S. G. Yakut, "Karar Ağacı Algoritmaları ve Çocuk İşçiliği Üzerine Bir

Uygulama", Sos. Bilim. Araşt. Derg., c. 8, sy 4, ss. 52-65, 2019.

[22]. B. S. Kuzu ve S. G. Yakut, "Destek Vektör Makineleri Yardimiyla Imalat Sanayisinde Mali Başarisizlik Tahminlerinin Teknoloji Yoğunluğuna Göre Incelenmesi", Osman. Korkut Ata Üniversitesi İktisadi Ve İdari Bilim. Fakültesi Derg., c. 4, sy 2, ss. 36-54, 2020.

[23]. D. Akmaz ve M. S. Mamiş, "Bayes, Lazy, Trees, Rules Sınıfı Makine Öğrenme Algoritmaları". 2nd International Mediterranean Science and Engineering Congress, 2017.

[24]. Seckin, "MetaSezgisel Algoritmalar", Bir Yazılımcının Günlüğü, 16 Mayıs 2017. https://biryazilimciningunlugu.wordpress.com/2017/05/16/metasezgisel-algoritmalar/ (erişim 16 Nisan 2022).

[25]. "Rules extraction system family", Wikipedia. 27 Kasım 2019. Erişim: 16 Nisan 2022. [Çevrimiçi]. Erişim adresi: https://en.wikipedia.org/w/index.php?title=Rules_extraction_ system_family&oldid=928196017