Published online 19 October 2016 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/sec.1660

RESEARCH ARTICLE

# A novel feature extraction approach in SMS spam filtering for mobile communication: one-dimensional ternary patterns

Yilmaz Kaya[1]* and Ömer Faruk Ertuğrul[2]

[1] Department of Computer Engineering, Siirt University, 56100 Siirt, Turkey
[2] Department of Electrical and Electronics Engineering, Batman University, 56100 Batman, Turkey

## ABSTRACT

The importance and utilization of mobile communication are increasing day by day, and the short message service (SMS) is one of them. Although SMS is a widely used communication way, it brings together a major problem, which is SMS spam messages. SMS spams do not only use vain in the mobile communication traffic but also disturb users. Based on this fact, blacklisting methods, statistical methods which are built on the frequency of occurrence of words or characters, and machine learning methods have been employed. Because punishments and legal laws are not enough to solve this problem and the Group Special Mobile number of SMS spam can easily be changed, a content-based approach must be proposed. Content-based methods showed high success in spam e-mail filtering, but it is hard in the SMS spam filtering because SMS messages are extremely short and generally contains many abbreviations. In this study, an image processing method, local ternary pattern was improved to extract features from SMS messages in the feature extraction stage. In the proposed one-dimensional ternary patterns, firstly, text message was converted to their UTF-8 values. Later, each character (its UTF-8 value) in the message was compared with its neighbors. Two different feature sets were extracted from the results of these comparisons. Finally, some machine learning methods were employed to classify these features. In order to validate the proposed approach, three different SMS corpora were used. The achieved accuracies and other employee performance measures showed that the proposed approach, one-dimensional ternary patterns, can be effectively employed in SMS spam filtering. Copyright © 2016 John Wiley & Sons, Ltd.

**KEYWORDS**
SMS; ternary patterns; spam filtering; text mining

***Correspondence**
Yilmaz Kaya, Department of Computer Engineering, Siirt University, 56100 Siirt, Turkey.
E-mail: yilmazkaya1977@gmail.com

## 1. INTRODUCTION

Although, short message service (SMS) and multimedia message services are the most commonly used mobile communication services, unfortunately, these services also suffer from a major problem, which is called SMS spam [1]. Nowadays, SMS communication takes a major part of the mobile communication traffic and SMS spams took a considerable part of this SMS traffic, which is 69.6% in 2013 [2]. Ahmet *et al.* [3] noticed that the number of received SMS spam message is higher than received ham messages. Although, SMS messages are not free (their cost may be lower than an SMS that is sent by a standard user, because of utilizing SMS packages from Group Special Mobile (GSM) operators), there is a high intensity of SMS spams. Because the spam filtering methods in SMS are not developed or employed like spam e-mail filtering techniques, SMS users need to check their SMS box and read each SMS in order to understand whether it is a spam or ham (this process is disturbing and takes the time); otherwise, with a non-successful SMS spam filtering method, an important ham message may easily drop to spam box. Despite legal laws, the SMS spam problem is increasing day by day. To solve this issue, an increasing number of methods, which can be classified into blacklisting, statistical methods that are based on the frequency of occurrence of words or characters and machine learning methods, have been proposed [4–7]. Although the source of an SMS spam may be easily searched or detected and their content are generally about advertisement, making money, adult products, or losing weight, an efficient solution has not been proposed yet [6,8–10]. Feature extraction can be considered as a major stage in SMS spam filtering, similar to spam e-mail filtering [11]. On the other hand, unlike to emails, it is very hard to extract relevant features in SMS messages, because they are generally short and may contain many abbreviations [12,13].

In this study, a novel feature extraction approach, one-dimensional ternary patterns (1D-TP), was proposed to extract relevant features from SMS messages. 1D-TP is a statistical approach that was built on the order of occurrence of the characters. In 1D-TP, patterns were formed from the comparisons of Unicode values of the characters in SMS messages with the Unicode values of their neighbors. Also, in 1D-TP, different micro/macro patterns can be generated via $P$ and $\beta$ parameters of 1D-TP. In order to evaluate and validate the proposed approach, three different benchmark SMS corpora were utilized and machine learning methods were employed to detect/filter spam. Achieved accuracies (based on 10-folds cross-validation) showed that the proposed approach can be employed in SMS spam filtering.

The rest of the paper is organized as follows. Section 2 gives related works on SMS spam filtering. Employed benchmark datasets were described in Section 3. In Section 4, 1D-TP method and its implementation procedure were introduced. Section 5 reports 1D-TP performances evaluation results. Finally, conclusions are presented in Section 6.

## 2. RELATED WORKS

Spam messages are not only a problem in the mobile communication but also a trouble in e-mail communication on the Internet. Many methods have been proposed and successfully employed in filtering spam e-mails [14,15]. Nevertheless, these methods may not show the same success in the SMS communication, because SMS texts are generally shorter than e-mail messages (a standard SMS is limited to 160 characters). Furthermore, there is not any standard text format, and generally, abbreviations are used in a message. For instance, instead of expressing "How are you" the users generally type only "how r u" [2]. Because of its importance, many methods have been introduced in the literature, and the proposed SMS spam filtering methods are reviewed by Delany *et al.* [16] in detail. Generally, blacklisting and content-based methods were employed in spam filtering [17]. In blacklisting, the GSM numbers of SMS spam senders or keywords were stored in a list, and when an SMS is received from one of the GSM number in the blacklist or it consists one of the black keywords, this SMS is labeled as spam and stored in the spam box. But unfortunately, these two blacklisting methods have their own drawbacks such as a GSM number can be easily changed and a word can be written in many formats that may be different from the stored keyword by abbreviations. Another type of SMS spam filtering method is content-based method. In these methods, frequencies of words or characters are employed as a key of spam searching. The proposed SMS spam filtering methods are summarized in Table I.

**Table I.** Related studies on short message service spam filtering.

| Reference | Author(s)/year | Method |
|---|---|---|
| [18] | Xiang, Chowdhury, and Ali (2004) | Support vector machine (SVM) |
| [19] | Healy, Delany, and Zamolotskikh (2004) | k-nearest neighbors (kNN) |
| [20] | Cai, Tang, and Hu (2008) | Winnow algorithm |
| [21] | Wu, Wu, and Chen (2008) | Bayes |
| [22] | Longzhen, An, and Longjun (2009) | kNN |
| [7] | Almeida, Gómez Hidalgo, and Yamakami (2011) | SVM, kNN, decision trees, C4.5, PART, |
| [23] | Deng and Peng (2006) | Naïve Bayes (NB) |
| [24] | Rafique and Farooq (2010) | Hidden Marcov Models (HMM) |
| [25] | Cormack, Gómez Hidalgo, and Sánz (2007) | Bigrams features |
| [26] | Sohn, Lee, and Rim (2009) | Stylistic features |
| [27] | He *et al.* (2008) | White/black listing |
| [4] | Healy *et al.* (2005) | kNN, SVM, and NB |
| [23] | Deng and Peng (2006) | SMS message characteristics such as its length |
| [1] | Yoon *et al.* (2010) | Content-based filtering and challenge response. |
| [28] | Gómez Hidalgo *et al.* (2006) | Lexical features (character and word n-grams) |
| [29] | Sohn *et al.* (2012) | Lexical features and stylistic features |
| [2] | Chen *et al.* (2015) | Characters based on the weight values and also the length of words |
| [3] | Ahmed *et al.* (2015) | Apriori + NB |
| [30] | Ali *et al.* (2015) | Dendritic cell algorithm |
| [31] | Onashoga *et al.* (2015) | Artificial immune system |
| [32] | Ho *et al.* (2013) | Graph-based kNN algorithm |

# 3. DATASETS

To make a fair comparison, three publicly available SMS spam corpora were used to evaluate and validate the proposed 1D-TP approach. The essential properties of these three corpora are summarized as follows:

(1) SMS Spam Corpus v.0.1 (DS1) is a set of SMS tagged messages that have been collected for an SMS spam research and employed by many researchers such as Hidalgo *et al.* [28] and Cormack *et al.* [33]. DS1 contains a total of 1002 legitimate messages and 322 spam messages. The average number of words per message is 15.72, and the average length of a word is 4.44 characters long [33].
(2) British English SMS Corpora (DS2) was obtained from GrumbleText, which is a website that collects SMS spam that was uploaded by people who complain receiving SMS spams (http://www.grumbletext.co.uk). This corpus contains a total of 450 legitimate and 425 spam.
(3) The last corpus (DS3) is generated by Almeida *et al.* [34]. This dataset contains 747 spam and 4827 ham.

Short message services that are longer than 30 characters were employed in this study, and others were excluded. The length of final employed datasets was summarized in Table II.

# 4. METHODS

## 4.1. One-dimensional ternary patterns

One-dimensional ternary patterns approach was built on the local ternary pattern, which is a widely used image processing method. Unlike local ternary pattern, 1D-TP was employed for extracting features in one-dimensional series. The value of each member of the 1-D series was compared with its neighbors, and the result of these comparisons was expressed as a decimal number [11,35]. For text messages, the comparisons were carried out after converting the characters to their UTF-8 values. In 1D-TP, the number of neighbors of a character was determined by the $P$ parameter. $P/2$ characters previously and after of central character were assigned as neighbors of that character. For instance, the implementation process of 1D-TP in an SMS sample for the case of $P$ is 8 (the neighbors of each character (central character, $P_c$) are four characters at the left (previous) of the $P_c$ and four characters at the right (late) of it) as illustrated in Figure 1.

As seen in Figure 1(A) and (B), for an SMS sample "whereru:)", the neighbors at the left are $P_L = (P_0, P_1, P_2, P_3)$ and ones at the right are $P_R = (P_4, P_5, P_6, P_7)$. Totally, neighbors for each $P_c$ are $P = \{P_0, P_1, P_2, P_3, P_4, P_5, P_6, P_7\}$. Addition to the number of neighbors, a threshold parameter ($\beta$), which was a user-defined parameter, was also utilized in the comparisons. The comparisons were carried out depending on the following equation as seen in Figure 1(C).

$$TP = \begin{cases} 1 & Pc > Pi + \beta \\ 0 & Pc \le Pc + \beta \text{ and } Pc \ge Pi - \beta \\ -1 & Pc < Pi - \beta \end{cases} . \quad (1)$$

Patterns extracted by 1D-TP depend on the employed $\beta$ value. The increase of the $\beta$ yields lower sensitivity, especially for text messages. Because the UTF-8 values of the widely used characters are near to each other. The local changes, which are in the range of ±$\beta$ (i.e., the value of $P_i$ is inside $P_c \pm \beta$), cannot be observed; therefore, also, some patterns cannot be extracted. Also, this process can be described as a band-stop filtering process, while $\beta$ shows that the band-stop bandwidth is ±$\beta$. Via this parameter, the value of the central character ($P_c$) is adjusted according to ±$\beta$ value. Therefore, for each $P_c$, its neighbors are filtered depending on $\beta$, as seen in Figure 2.

After the filtering process by $\beta$, which is illustrated in Figure 2, two different decimal numbers were generated based on the comparison results for each $Pc$ (Figure 1(**D**)). Negative ones ($Pc < Pi - \beta$) were employed to generate the low features while the positive ones ($Pc > Pi + \beta$) were used for extract up features. Obtained decimal numbers (up–low) were used instead of the UTF-8 values of the text messages (Figure 1(**E**)) in two feature sets (up–low features). This process continues for each data in the message, and at the end of this process, two different 1D signals, which are extracted low and up values, were obtained. Later, two different histograms, which are lower and upper histograms, were formed from these 1D signals. For the case of $P$ is 8 (the total number of utilized neighbors is 8), four neighbors are taken from left and four from right. In this case, the values in the 1D signals vary from 0 to 255 ($2^8 = 256$). A variation in $P$ also causes a change in the length of extracted histograms, such that the length of

**Table II.** Properties of datasets.

| Dataset | # SMS in the dataset | # SMS spam | # SMS ham | Ratio of spam in dataset (%) | Ratio of ham in dataset (%) |
|---|---|---|---|---|---|
| DS1 | 913 | 321 | 592 | 35.16 | 64.84 |
| DS2 | 786 | 419 | 367 | 53.31 | 46.69 |
| DS3 | 3902 | 740 | 3162 | 18.96 | 81.04 |

SMS, short message service.

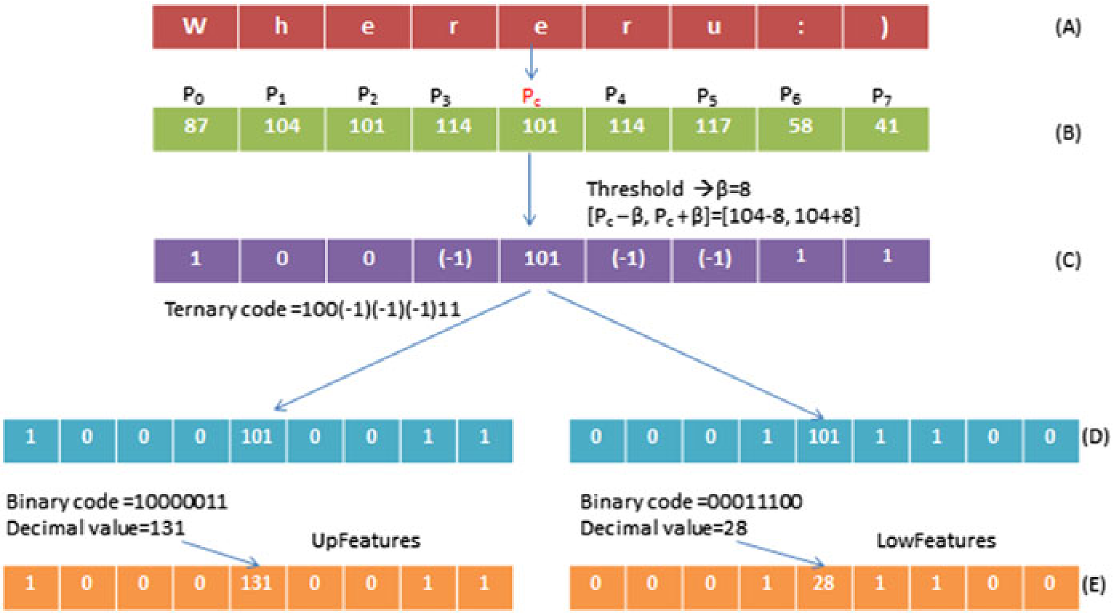

**Figure 1.** Feature extraction steps. (A) An SMS sample, (B) UTF-8 values of characters in the sample SMS, (C) comparison of *Pc* with neighbors (*Pi*), (D) separation positive and negative values, and (E) conversion of binary values to decimal.

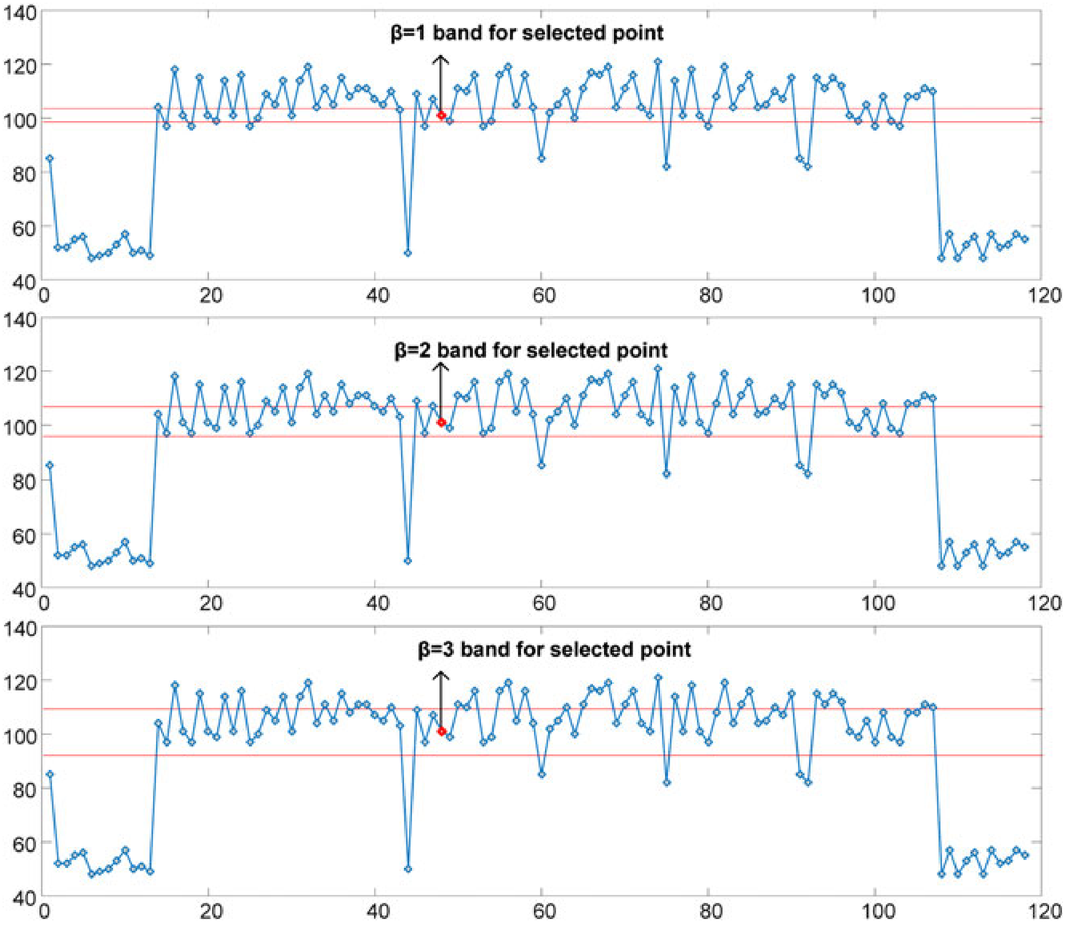

**Figure 2.** UTF-8 values of a text message for three different $\beta$ values.

histograms will be 64 ($2^6$), 256 ($2^8$), and 1024 ($2^{10}$) if $P$ is assigned as 6, 8, and 10, respectively. As a summary, a variation in both $P$ and $\beta$ causes detecting different patterns. The optimal values for both of them can be determined by trials or an expert view.

## 4.2. Criteria for performance evaluation

To compare the results, accuracy (rate of correctly classified samples), spam caught (SC; rate of correctly classified spam samples), blocked hams (BH; rate of incorrectly classified ham samples), and Matthews correlation coefficient (MCC), which are performance measures that are widely adopted in evaluating spam e-mail classification, were employed [7,36]. These rates are defined as follows:

$$Accuracy\ Rate\ (Acc) = \frac{N_{\mathrm{TP}} + N_{\mathrm{TN}}}{N_{\mathrm{F}} + N_{\mathrm{T}}} \tag{2}$$

$$\text{Spam Caught}\ (SC) = \frac{N_{\mathrm{TP}}}{N_{\mathrm{TP}} + N_{\mathrm{FP}}} \tag{3}$$

$$\text{Blocked Hams}\ (BH) = \frac{N_{\mathrm{TN}}}{N_{\mathrm{TN}} + N_{\mathrm{FN}}} \tag{4}$$

$$MCC = \frac{N_{\mathrm{TP}}xN_{\mathrm{TN}} - N_{\mathrm{FP}}xN_{\mathrm{FN}}}{\sqrt{(N_{\mathrm{TP}+}N_{\mathrm{FP}})(N_{\mathrm{TP}} + N_{\mathrm{FN}})(N_{\mathrm{TN}} + N_{\mathrm{FP}})(N_{\mathrm{TN}} + N_{\mathrm{FN}})}} \tag{5}$$

where $N_{\mathrm{T}}$ is the total number of spam, $N_{\mathrm{F}}$ is the total number of ham, $N_{\mathrm{TP}}$ is the total number of correctly classified spam, $N_{\mathrm{FP}}$ is the total number of SMS hams, which are falsely classified as spam, and $N_{\mathrm{FN}}$ is the total number of SMS spams, which are falsely classified as ham. A false positive is mistakenly classified a ham as a spam, and a false negative is mistakenly classified a spam as a ham.

## 4.3. Proposed method

In this study, a novel SMS spam filtering approach, in which statistical features were extracted from the characters in the text, was employed. The implementation procedure of the proposed approach is given in Figure 3.

Block 1: In this block, special characters, such as punctuation, the blank, carriage return, and newline characters, were removed, and the remaining part was converted to Unicode. For example, for the message:

> A Novel Feature Extraction Approach in SMS Spam Filtering for Mobile Communication: One-Dimensional Ternary Patterns

First, the unwanted characters were removed; the remaining message was

> ANovelFeatureExtractionApproachinSMSSpamFilteringf-orMobileCommunication:
> OneDimensionalTernaryPatterns

After this, the message was converted to the UTF-8 values of the characters in the text (Figure 4(A)):

65 78 111 118 101 108 70 101 97 116 117 114 101 69 120 116 114 97 99 116 105 111 110 65 112 112 114 111 97 99 104 105 110 83 77 83 83 112 97 109 70 105 108 116 101 114 105 110 103 102 111 114 77 111 98 105 108 101 67 111 109 109 117 110 105 99 97 116 105 111 110 58

Block 2: The features in the SMS messages were extracted by employing 1D-TP as explained in the previous section. By applying this procedure, a 1D-TP signal was formed, which had values in the range of 0 to 255 for $P = 8$, and extracted up and low features are illustrated in Figure 4(B). As seen in this figure and also from Equation (1), two different 1D-TP signals can be extracted from an SMS message.

Block 3: The histograms of the transformed 1D-TP signal, which shows how often each of the patterns appear in its corresponding signal, were determined in this block. If $P$ takes the value of 8, there will be 256 features in 1D-TP for classification. The extracted low and up histograms for the example given in Block 1 are illustrated in Figure 4(C).

Block 4: In the last stage, different classification techniques such as Naïve Bayes, Bayesian network, random forest (RF), k nearest neighbors, and artificial neural network methods were employed to classify SMS messages into two classes: spam or ham. In this stage,

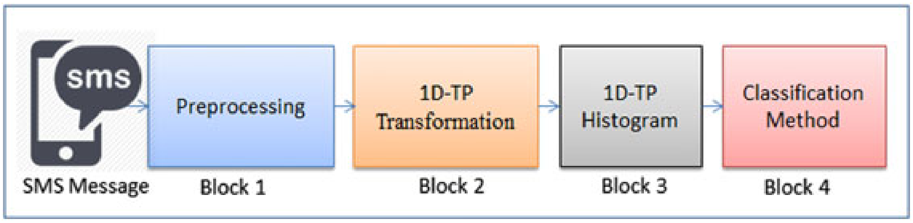

**Figure 3.** The block diagram of the proposed approach. SMS, short message service; 1D-TP, one-dimensional ternary patterns.

DOI: 10.1002/sec

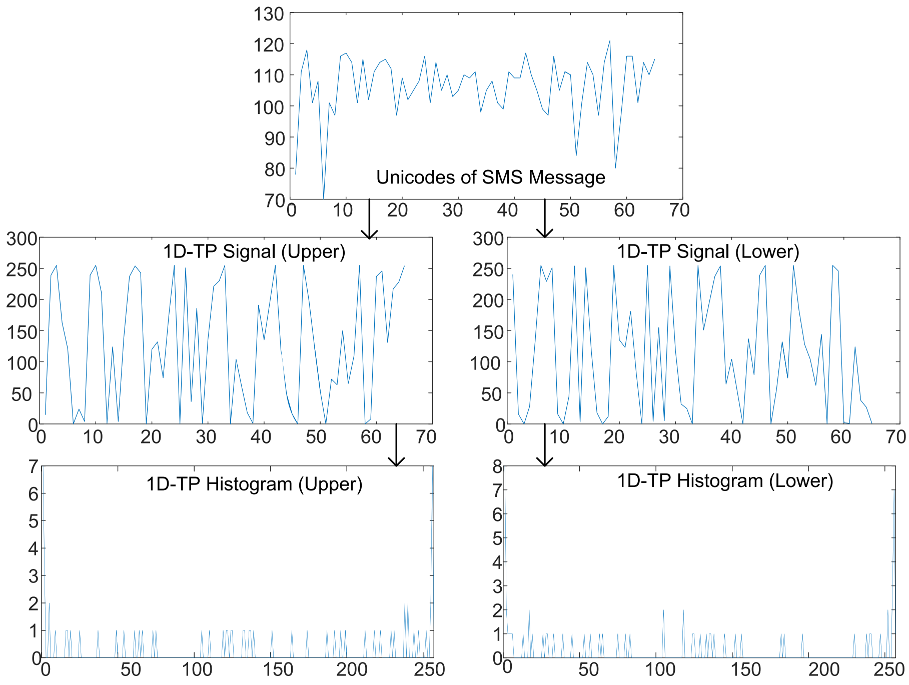

**Figure 4.** Selecting neighbors stage in obtaining a signal of a sample SMS for $P = 8$. SMS, short message service; 1D-TP, one-dimensional ternary patterns.

the validation process was applied based on 10-folds cross-validation.

## 5. RESULTS

### 5.1. Determining optimal parameters of ternary patterns

In this study, a novel feature extraction approach (1D-TP), which is based on the UTF-8 values of the characters in an SMS, was proposed in order to filter SMS spam messages by classifying an SMS into ham or spam. In 1D-TP, two different feature sets (low and up features) can be extracted from a signal for each threshold value ($\beta$) and the number of neighbors ($P$). 1D-TP is sensitive to small local changes and via $P$ and $\beta$ different patterns can be detected. Therefore, both of $P$ and $\beta$ are important parameters to detect patterns that can distinguish the SMS messages. The optimal values of both $P$ and $\beta$ can be determined by trials or an expert view. Achieved accuracies by RF (when $\beta$ is 4) in the trials, while $P$ is 6, 8, and 10 are summarized in Table III.

As seen in Table III, no correlation was found between obtained classification accuracy and assigned $p$ value. But lengths of up and low histograms, which are the extracted features, are depended on the $P$ parameter (which is $2^P$). Therefore, larger $p$ values yield higher computational cost, but this does not mean to say that having higher accuracy. In general, higher $p$ values yield larger comparison areas around the $P_c$. In the further steps, $P$ was assigned as 8 and only the $\beta$ was utilized to filter SMS spam. The features extracted from a spam and a ham SMS by employing different $\beta$ values are illustrated in Figure 5 and 6.

**Table III.** Achieved classification accuracies by different *p* values.

| Dataset | *p* | #Features | Low features | Up features |
|---|---|---|---|---|
| DS1 | 6 | 64 | 90.3614 | 89.4852 |
| | 8 | 256 | 93.3187 | 91.6758 |
| | 10 | 1024 | **93.9759** | 89.7043 |
| DS2 | 6 | 64 | 85.1145 | 82.9517 |
| | 8 | 256 | 87.1501 | 83.8422 |
| | 10 | 1024 | **89.313** | 88.0407 |
| DS3 | 6 | 64 | 93.7212 | 92.8754 |
| | 8 | 256 | **94.1056** | 92.5679 |
| | 10 | 1024 | 93.8493 | 92.2348 |

Bold entries that were given in each table showed achieved highest accuracy of that case.

As seen from these figures and also in Equation (1), a variation in $\beta$ changes the detected patterns, which are also extracted features in both a spam and a ham SMS. Neighbors ($P$i) that have values near $P_c$ in the range of $\pm\beta$ cannot be observed or in other words, they are lost. Therefore, the energy of the histograms increases at the edges of histograms as seen in Figures 5 and 6. To make it clearer, low features extracted from both a ham and SMS spam are given in Figure 7. It is obvious from this figure that, the histograms extracted from ham and SMS spams are different from each other.

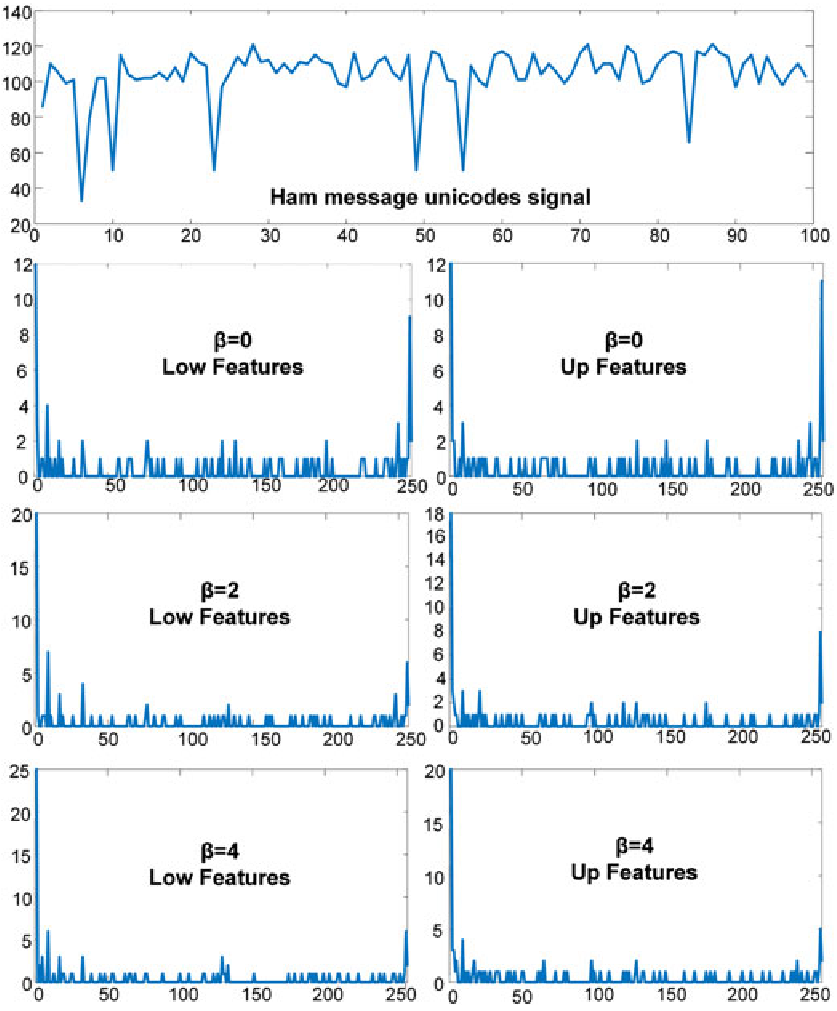

**Figure 5.** Extracted histograms for different $\beta$ values from a ham short message service.

## 5.2. Classification results

In order to evaluate and validate the proposed approach, three different SMS corpora were employed. The features were extracted by 1D-TP, and later, extracted features were employed to classify SMS messages into spam and ham. In the classification process, popular machine learning methods such as the Bayesian network, Naïve Bayes, radial basis feed forward neural network, k nearest neighbors, and RF were employed. The classifications were carried out by WEKA (University of Waikato, New Zealand) [36] based on 10-folds cross-validation. Achieved accuracies for various $\beta$ values (while P equals to 8) for DS1, DS2, and DS3 are summarized in Tables IV–VI, respectively.

As seen from Tables IV–VI, different accuracies were obtained by employing different $\beta$ values, because in each case, a different pattern was detected. In DS1 dataset, achieved highest accuracies were 93.31% and 91.68% by low and up features that are extracted by 1D-TP$_{p=8,\beta=4}$. Obtained accuracies were varied from 92.11% to 93.31% in low features based on employed $\beta$ values. The achieved highest accuracies by low and up features that were extracted from DS2 dataset were 87.15% and 85.24%, respectively. In DS3 dataset, achieved highest accuracies were 94.11% and 93.11% by low and up features, respectively. Although the highest accuracies for each case were achieved by RF (except up features extracted from DS2 dataset), obtained accuracies by other employed machine learning methods are also acceptable. In addition to accuracy, other obtained performance measures by RF were summarized in Table VII.

Although accuracy shows the overall performance of the proposed approach, other calculated performance measures (SC, MCC, and BH) are important measures to determine the success of the SMS spam filtering. As seen in Table VII, the proposed feature extraction approach (1D-TP) with RF showed high success, because SC and MCC ratios, which show the ratio of correctly blocked SMS spams, are high and BH, which shows ham ratio of messages that are labeled as spam incorrectly, is low enough to be accepted. In SMS spam filtering, the ratio of incorrectly filtered SMS (BH) is much more important than

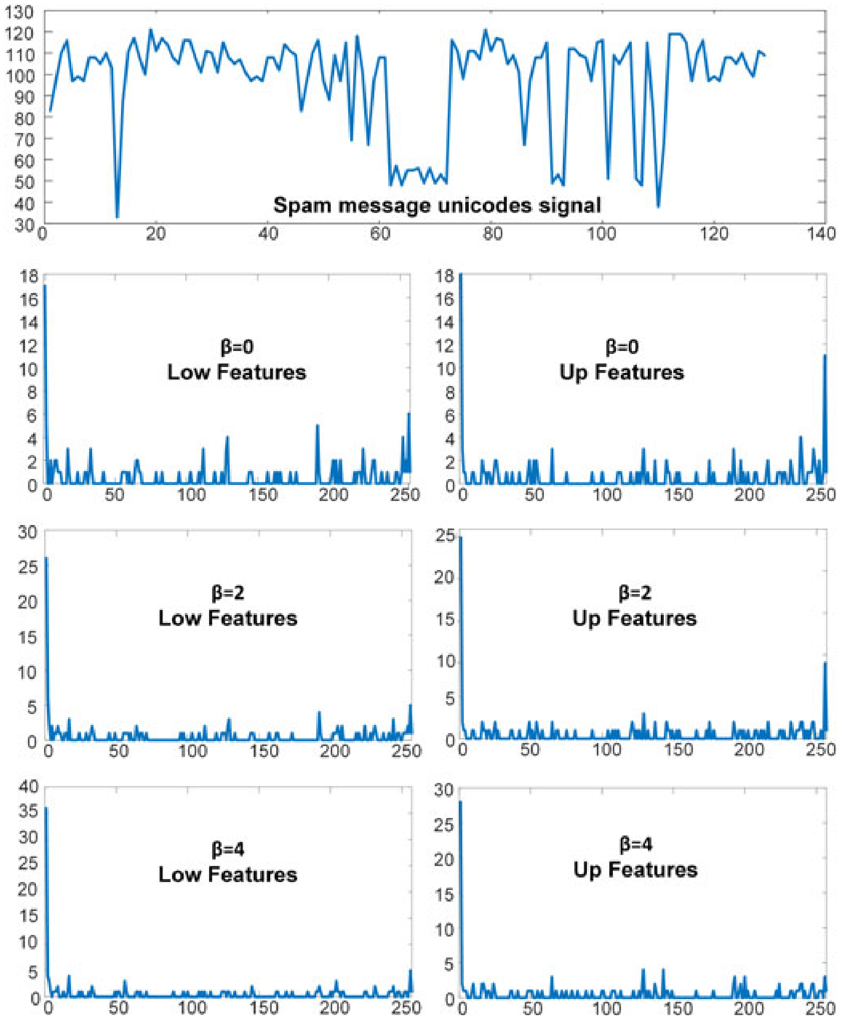

**Figure 6.** Extracted histograms for different $\beta$ values from an short message service spam.

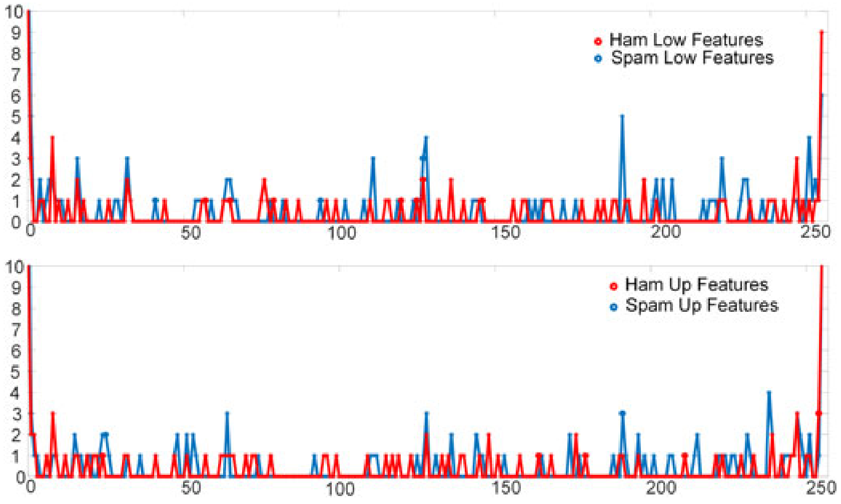

**Figure 7.** Obtained histograms for Low and Up features from a ham and short message service spam ($\beta = 0$).

**Table IV.** Achieved accuracies in DS1 dataset.

| Features | Model | $\beta = 0$ | $\beta = 1$ | $\beta = 2$ | $\beta = 3$ | $\beta = 4$ |
|---|---|---|---|---|---|---|
| Low Features | BN | 86.8565 | 89.2662 | 88.3899 | 88.0613 | 91.0186 |
| | NB | 86.3089 | 87.0756 | 88.1709 | 87.2946 | 89.3757 |
| | RBFN | 87.4042 | 87.2946 | 87.4042 | 86.5279 | 88.0613 |
| | kNN | 87.6232 | 87.0756 | 88.2804 | 86.8565 | 86.4184 |
| | RF | 92.1139 | 92.1139 | 92.7711 | 92.6616 | **93.3187** |
| Up Features | BN | 87.1851 | 86.8565 | 88.1709 | 87.5137 | 89.4852 |
| | NB | 85.9803 | 86.4184 | 86.9660 | 87.9518 | 86.6375 |
| | RBFN | 84.6659 | 85.4326 | 86.4184 | 86.3089 | 86.5279 |
| | kNN | 86.5279 | 86.7470 | 85.9803 | 87.0756 | 86.8565 |
| | RF | 90.2519 | 89.9233 | 91.2377 | 90.9091 | **91.6758** |

Bold entries that were given in each table showed achieved highest accuracy of that case.

BN, Bayesian network; kNN, k nearest neighbors; NB, Naïve Bayes; RBFN, radial basis feed forward neural network; RF, random forest.

**Table V.** Achieved accuracies in DS2 dataset.

| Features | Model | $\beta = 0$ | $\beta = 1$ | $\beta = 2$ | $\beta = 3$ | $\beta = 4$ |
|---|---|---|---|---|---|---|
| Low Features | BN | 85.2417 | 84.7328 | 84.6056 | 86.514 | 86.514 |
| | NB | 84.8601 | 84.4784 | 85.6234 | 84.6056 | 85.4962 |
| | RBFN | 83.3333 | 84.9873 | 84.6056 | 82.9517 | 82.3155 |
| | kNN | 79.3384 | 79.3384 | 78.1934 | 80.1018 | 79.4656 |
| | RF | 83.5878 | 86.3868 | 86.514 | 85.4962 | **87.1501** |
| Up Features | BN | 83.8422 | **85.2417** | 83.0789 | 83.3333 | 82.9517 |
| | NB | 84.2239 | 84.6056 | 83.4606 | 84.7328 | 84.9873 |
| | RBFN | 82.3155 | 81.5522 | 81.5522 | 80.0254 | 83.5878 |
| | kNN | 78.3206 | 80.3562 | 79.5929 | 80.3562 | 79.7201 |
| | RF | 84.9873 | 83.2061 | 84.2239 | 84.0967 | 83.8422 |

Bold entries that were given in each table showed achieved highest accuracy of that case.

BN, Bayesian network; kNN, k nearest neighbors; NB, Naïve Bayes; RBFN, radial basis feed forward neural network; RF, random forest.

**Table VI.** Achieved accuracies in DS3 dataset.

| Features | Model | $\beta = 0$ | $\beta = 1$ | $\beta = 2$ | $\beta = 3$ | $\beta = 4$ |
|---|---|---|---|---|---|---|
| Low Features | BN | 84.0082 | 85.1102 | 85.8022 | 86.1866 | 86.6479 |
| | NB | 85.7509 | 86.6735 | 87.673 | 87.1092 | 88.0318 |
| | RBFN | 85.6228 | 85.9047 | 87.1604 | 86.8785 | 86.9042 |
| | kNN | 92.0554 | 92.2091 | 92.2091 | 92.8242 | 92.7217 |
| | RF | 93.2599 | 93.4905 | 93.5418 | 93.9518 | **94.1056** |
| Up Features | BN | 83.0087 | 82.8549 | 83.5982 | 82.6499 | 84.2132 |
| | NB | 84.6489 | 85.2896 | 85.1358 | 84.9821 | 85.9303 |
| | RBFN | 84.572 | 85.6484 | 84.4439 | 84.8283 | 84.3157 |
| | kNN | 91.9528 | 92.5935 | 91.9785 | 91.9272 | 92.5679 |
| | RF | 92.6448 | 92.9011 | **93.1061** | 92.4910 | 92.5679 |

Bold entries that were given in each table showed achieved highest accuracy of that case.

BN, Bayesian network; kNN, k nearest neighbors; NB, Naïve Bayes; RBFN, radial basis feed forward neural network; RF, random forest.

**Table VII.** Obtained performance measures in DS1, DS2, and DS3 datasets.

| Features / Model | Dataset | SC | BH | Accuracy | MCC |
|---|---|---|---|---|---|
| Low Features($\beta = 4$) + RF | DS1 | 97.29 | 0,143 | 93.3187% | 0.853 |
| Low Features($\beta = 4$) + RF | DS2 | 89.97 | 0,160 | 87.1501% | 0.721 |
| Low Features($\beta = 4$) + RF | DS3 | 80.9% | 0.004 | 94.1056% | 0.800 |

BH, blocked hams; MCC, Matthews correlation coefficient; RF, random forest; SC, spam caught.

SC, MCC, and accuracy, because wrongly labeled SMS spam can easily be deleted with a time lose, but the wrongly labeled ham message can be unread by the user, which means the SMS may be deleted after an automatic process.

In addition to obtained acceptable accuracy, SC, MCC, and BH performance metrics, the proposed approach has some other advantages over the employed methods in the literature. It is not based on the frequency of occurrence of words or characters; therefore, the proposed approach is faster than this type of method. Furthermore, the proposed approach does not require any update, akin to blacklisting or black keywords methods. Moreover, the proposed approach can be employed in any type of text such as short text and/or a text that contain abbreviations, because it is built on the UTF-8 values of the characters in the text. On the other hand, to employ the proposed approach effectively, the parameters of 1D-TP ($P$ and $\beta$) must be optimized. This is a limitation of the proposed approach, which can be overcome by trials.

## 6. CONCLUSION

The increase in mobile communication technology yields various outcomes to human life and one of them is SMS communication, which is one of the most popular communication ways nowadays. Similar to other popular communication ways, SMS also suffers from spam messages. The total ratio of SMS spam overall SMS traffic is increased day by day. SMS spams do not only take the time of users but also takes an important part of mobile traffic. Nevertheless, spam filtering in SMS communication is not as easy as in e-mail communication, because of abbreviations and shortage of the SMSs. In this study, a novel approach, 1D-TP, was proposed to extract features from SMS messages and spam that are filtered by employing machine learning methods that utilized extracted features. 1D-TP is a statistical feature extraction method that is based on the comparisons of characters with their neighbors according to their UTF-8 values. Two different features set, low and up features, were extracted from the results of comparisons. In order to evaluate and validate the proposed approach, three different SMS corpora were employed. The achieved accuracies, which were 93.318%, 87.15%, and 94.10%, showed that the proposed approach can be successfully employed in feature extraction in SMS filtering instead of the statistical methods that are built on the frequency of occurrence of words or characters. Additionally, the proposed approach was not required to be constantly updated akin to blacklisting or black keywords-based methods. The 1D-TP is based on the UTF-8 values of the characters in the text, and therefore, it is effective for each type of messages such as short texts and/or texts that contain abbreviations. But determining optimal parameters of 1D-TP ($P$ and $\beta$) is a limitation of the proposed approach, which can be overcome by trials. Based on achieved successful results and the implementation procedure of 1D-TP, the proposed approach can be easily employed in mobile phones to prevent their users from SMS spam. Moreover, the authors claim that 1D-TP has also a high potential to be employed in natural language processing applications.

## REFERENCES

1. Yoon JW, Kim H, Huh JH. Hybrid spam filtering for mobile communication. *Computers & Security* 2010; **29**(4):446–459.
2. Chen L, Yan Z, Zhang W, Kantola R. TruSMS: a trustworthy SMS spam control system based on trust management. *Future Generation Computer Systems* 2015; **49**:77–93.
3. Ahmed, I, Ali, R, Guan, D, Lee, YK, Lee, S, Chung, T Semi-supervised learning using frequent itemset and ensemble learning for SMS classification. *Expert Systems with Applications*, 2015, **42**(3): 1065–1073.
4. Healy, M, Delany, S, Zamolotskikh, A. An assessment of case-based reasoning for short text message classification. In N Creaney (ed.) *Proceedings of 16th Irish Conference on Artificial Intelligence and Cognitive Science, (AICS-05)*, University of Ulster: Portstewart, Northern Ireland, United Kingdom, 2005; 257–266.
5. Idris I, Selamat A, Omatu S. Hybrid email spam detection model with negative selection algorithm and differential evolution. *Engineering Applications of Artificial Intelligence* 2014; **28**:97–110.
6. Su M-C, Lo H-H, Hsu F-H. A neural tree and its application to spam e-mail detection. *Expert Systems with Applications* 2010; **37**(12):7976–7985.
7. Almeida TA, Hidalgo JMG, Yamakami A. Contributions to the study of SMS spam filtering: new collection and results. In *Proceedings of the 11th ACM Symposium on Document Engineering*. ACM: New York, NY, USA, 2011; 259–262.
8. Delany SJ, Buckley M, Greene D. SMS spam filtering: methods and data. *Expert Systems with Applications* 2012; **39**(10):9899–9908.
9. Uysal AK, Gunal S, Ergin S, Sora Gunal E. The impact of feature extraction and selection on SMS spam filtering. *Elektronika ir Elektrotechnika* 2012; **19**(5):67–72.
10. Guzella TS, Camınhas WM. A review of machine learning approaches to spam filtering. *Expert Systems with Applications* 2009; **36**(7):10206–10222.
11. Kaya Y, Ertuğrul ÖF. A novel approach for spam email detection based on shifted binary patterns. *Security and Communication Networks* 2016. doi:10.1002/sec.1412.
12. Liu W, Wang T. Index-based online text classification for sms spam filtering. *Journal of Computers* 2010; **5**(6):844–851.

13. Cormack GV, Lynam TR. Online supervised spam filter evaluation. *ACM Transactions on Information Systems (TOIS)* 2007; **25**(3):11.
14. Hou YT, Chang Y, Chen T, Laih CS, Chen CM. Malicious web content detection by machine learning. *Expert Systems with Applications* 2010; **37**(1):55–60.
15. Wang AH. Detecting spam bots in online social networking sites: a machine learning approach. In *Data and Applications Security and Privacy XXIV*. Springer: Berlin Heidelberg, 2010; 335–342.
16. Delany SJ, Buckley M, Greene D. SMS spam filtering: methods and data. *Expert Systems with Applications* 2012; **39**(10):9899–9908.
17. Taufiq M, Abdullah MFA, Kang K, Choi D. A survey of preventing, blocking and filtering short message services (SMS) spam. In *Proceedings of International Conference on Computer and Electrical Engineering*. IIEEE: NY, USA, 2010; 462–466.
18. Xiang, Y; Chowdhury, M; Ali, S. Filtering mobile spam by support vector machine. In *CSITeA'04: Third International Conference on Computer Sciences, Software Engineering, Information Technology, E-Business and Applications*. International Society for Computers and Their Applications (ISCA): Cairo, Egypt, 2004; 1–4.
19. Healy M, Delany SJ, Zamolotskıkh A. An assessment of case base reasoning for short text message classification. In Conference papers. 2004: 42.
20. Cai, J; Tang, Y; Hu, R. Spam filter for short messages using winnow. In *Advanced Language Processing and Web Information Technology*, *2008*. *ALPIT'08*. *International Conference on*. IEEE, 2008: 454–459.
21. Wu, N; Wu, M; Chen, S. Real-time monitoring and filtering system for mobile SMS. In Industrial Electronics and Applications, 2008. ICIEA 2008. 3rd IEEE Conference on, 2008: 1319–1324.
22. Duan L, Li N, Huang L. A new spam short message classification. In IEEE *2009 First International Workshop on Education Technology and Computer Science*, 2009; 168–171.
23. Deng W-W, Peng H. Research on a naive bayesian based short message filtering system. In *Machine learning and cybernetics*, *2006 international conference on*. IEEE, 2006; 1233–1237.
24. Rafique MZ, Farooq M. Sms spam detection by operating on byte-level distributions using hidden markov models (hmms). In Proceedings of the 20th virus bulletin international conference. 2010.
25. Cormack GV, Gómez Hidalgo JM, Sanz EP. Spam filtering for short messages. In *Proceedings of the Sixteenth ACM Conference on Conference ON İnformation and Knowledge Management*. ACM: New York, NY, USA, 2007; 313–320.
26. Sohn D-N; Lee, J-T; Rim, H-C. The contribution of stylistic information to content-based mobile spam filtering. In Proceedings of the ACL-IJCNLP 2009 Conference Short Papers. Association for Computational Linguistics, 2009; 321–324.
27. He P, Sun Y, Zheng W, Wen X. Filtering short message spam of group sending using CAPTCHA. In *Knowledge Discovery and Data Mining*, *2008*. *WKDD 2008*. *First International Workshop on*. IEEE, 2008: 558–561.
28. Gómez Hidalgo JM, Bringas GC, Sánz EP, García FC. Content based SMS spam filtering. In *Proceedings of the 2006 ACM symposium on Document engineering*. ACM, 2006; 107–114.
29. Sohn DN, Lee JT, Han KS, Rim HC. Content-based mobile spam classification using stylistically motivated features. *Pattern Recognition Letters* 2012; **33** (3):364–369.
30. Al-Hasan AA, El-Alfy E-SM. Dendritic cell algorithm for mobile phone spam filtering. *Procedia Computer Science* 2015; **52**:244–251.
31. Onashoga AS, Abayomi-Alli OO, Sodiya AS, Ojo DA. An adaptive and collaborative server-side SMS spam filtering scheme using artificial ımmune system. *Information Security Journal: A Global Perspective* 2015; **24**(4–6):133–145.
32. Ho TP, Kang H-S, Kim S-R. Graph-based KNN algorithm for spam SMS detection. *Journal of Universal Computer Science* 2013; **19**(16):2404–2419.
33. Cormack GV, Hidalgo JMG, Sánz, EP. Feature engineering for mobile (SMS) spam filtering. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2007: 871–872.
34. Almeida TA, Yamakamı A, Almeıda J. Evaluation of approaches for dimensionality reduction applied with Naive Bayes anti-spam filters. In *Machine Learning and Applications*, *2009*. *ICMLA'09*. *International Conference on*. IEEE, 2009; 517–522.
35. Kaya Y, Uyar M, Tekin R, Yıldırım S. 1D-local binary pattern based feature extraction for classification of epileptic EEG signals. *Applied Mathematics and Computation* 2014; **243**:209–219.
36. Witten IH, Frank E. *Data Mining: Practical Machine Learning Tools and Techniques* (2nd edn). Morgan Kaufmann: San Francisco, CA, 2005.