IET Software

Research Article

IET Journals
The Institution of Engineering and Technology

Development of content-based SMS classification application by using Word2Vec-based feature extraction

ISSN 1751-8806
Received on 13th February 2018
Revised 12th September 2018
Accepted on 15th October 2018
E-First on 11th December 2018
doi: 10.1049/iet-sen.2018.5046
www.ietdl.org

Serkan Ballı[1] ✉, *Onur Karasoy*[1]

[1]*Department of Information Systems Engineering, Faculty of Technology, Muğla Sıtkı Koçman University, 48000 Mugla, Turkey*
✉ *E-mail: serkan@mu.edu.tr*

Abstract: While mobile instant messaging applications such as WhatsApp, Messenger, Viber offer benefits to phone users such as price, easy usage, stable, collective and direct communication, SMS (short message service) is still considered a more reliable privacy-preserving technology for mobile communication. This situation directs the institutions that want to perform the product promotion such as advertising, informing, promotion etc. to use SMS. However, spam messages sent from unknown sources constitute a serious problem for SMS recipients. In this study, a content-based classification model which uses the machine learning to filter out unwanted messages is proposed. From the selected dataset, the model to be used in the classification is created with the help of Word2Vec word embedding tool. Thanks to this model, two new features are revealed for calculating the distances of messages to spam and ham words. The performances of the classification algorithms are compared by taking these two new features into consideration. The random forest method succeeded with a correct accuracy rate of 99.64%. In comparison to other studies using the same dataset, more successful correct classification percentage is achieved.

1 Introduction

In today's technology world, the next generation mobile phones give performance almost as good as personal computers [1]. In this context, the number of mobile phone users is increasing every day. With the increasing number of users, the use of short message service (SMS) becomes widespread and it is preferred for personal messaging and authentication (mobile banking) method [2, 3].

Spam is the type of unwanted message that can be sent electronically. Spam e-mails are dispatched over the internet while SMSs are sent over the mobile network [4]. SMS messages are among the most convenient ways to deliver promotions and advertisements to users [5]. Spam SMSs are not only disturbing, but also pose security threats since they may contain links that redirect the users to malware. In some Asian countries such as South Korea and China, the traffic of spam SMS is superior to that of spam e-mail [6]. Therefore, spam filtering is a problem on which people are working for many years.

Numerous studies are carried out on this subject and some specific datasets are used for the detection of spam messages. One of these datasets is SMS Spam Collection Dataset [7], which is prepared by Almeida *et al.* [8]. It is a dataset consisting of 5574 messages in English with open access and it is frequently preferred in SMS classification studies. SMS Spam Corpus v.0.1 [9] is a dataset created by Hidalgo and Sanz and consists of 1324 messages in English. NUS SMS Corpus [10] consists of 10,000 messages collected from computer science students of the National University of Singapore. Dublin Institute of Technology (DIT) SMS Spam Dataset [11] consists of 1353 spam SMS messages from the UK website that collects consumer complaints. Turkish SMS dataset [12] consists of 420 Spam 420 Ham Turkish messages created by Uysal *et al.* These datasets are accessible for using in academic studies.

Some of these datasets are no longer available, and some of them are based on YouTube comments, so SMS Spam Collection dataset [7] is the cleanest one for this work. Also, this dataset is chosen because it is a combination of multiple datasets and it has more messages than others, and there is more work using this dataset for comparing the results.

In this study, SMS Spam Collection dataset prepared by Almeida *et al.* [8] is used. When the other studies using the same dataset are taken into consideration, the more successful accuracy rate is achieved with 99.64% by using the structural features of the message and two new features revealed by Word2vec. In addition to the previously used features, new features are extracted with the help of the Word2vec library by transforming the structural features of the message into text and adding them to the message. After that, the classification is carried out by traditional classification methods such as random forest, multi-layer perceptron (MLP), support vector machine (SVM), logistic regression and Naive Bayes methods and accuracy rate of 99.64% is achieved by random forest method.

2 Related work

Table 1 shows the years of SMS filtering and classification studies, the classification methods used, the number of features, the datasets used, the recommended classification method and the correct classification accuracy rates.

Almeida *et al.* [8] proposed an SVM-supported solution in their study. They create SMS Spam Collection by combining different datasets and use 30% of the dataset in training and 70% of the dataset in classification. Spam Capture (SC%), Blocked Ham Messages (BH%), Accuracy (Acc%) rates and Matthews correlation coefficient values are taken into consideration in order to compare the results. SVM is found to be the most successful technique with an accuracy rate of 97.5%.

Bozan *et al.* [13] present a solution proposal with classification methods with the help of an expert system. Features are extracted from the expressions and words in the dataset and, the features that are determined according to some criteria as they would not affect the result are eliminated. As a result, 6622 features are obtained and CfsSubset feature selection method is applied. The results obtained with the determined features with SVM, Naive Bayes (NB) and k-nearest neighbors (kNN) algorithms are compared and it is determined that SVM is the most successful classifier with 98.61% accuracy rate.

He *et al.* [14] proposed a linguistic attribute hierarchy (LAH) embedded with linguistic decision trees (LTD). The features extracted from the dataset are divided into subsets semantically. In the study, they are divided into three sub-levels and the LAHs are

Table 1 Summary of the studies

Reference number	Author(s)	Year	Classification methods used	Number of features	Dataset	Recommended method	Accuracy rate, %
[8]	Almeida *et al.*	2011	SVM, navie Bayes, boosted, C4.5, PART, MDL, kNN	81,175	SMS Spam Colleciton v1	SVM	97.50
[13]	Bozan *et al.*	2015	SVM, NB, kNN	59,016	SMS Spam Colleciton v1	SVM	98.61
[14]	He *et al.*	2017	Deep learning, LAHs based on decomposition	20	SMS Spam Colleciton v1	Deep learning	
[2]	Ho *et al.*	2014	Graph-based kNN		SMS Spam Colleciton v1	Graph-based kNN	98.90
[15]	Arifin *et al.*	2016	FP-growth and Naive Bayes classifier	—	- SMS Spam Collection v.1 - SMS Spam Corpus v. 0.1 Big	FP-growth and Naive Bayes classifier	98.59
[16]	Waheeb *et al.*	2015	ANN-SVG	—	- SMS Spam Colleciton v1 - DIT SMS spam Dataset - British English SMS - NUS SMS Corpus	ANN-SVGFuture-size:100	99.10
[17]	Ma *et al.*	2016	MTM (message topic model) SVM	—	- SMS Spam Colleciton v1 - DIT SMS spam Dataset	MTM	97
[18]	Fernandes *et al.*	2015	ANN-MLP, kNN, OPF with complete graph, OPF with kNN graph and SVM	—	SMS Spam Colleciton v1	OPF with complete	92.23
[19]	Akbari and Sajedi	2015	LPBoost, AdaBoost, TotalBoost, LogitBoost, GentleBoost, RobustBoost, RusBoost, SVM and NB	32	SMS Spam Colleciton v1	GentleBoost	98.30
[20]	Suleiman ve and Al-Naymat	2017	Random forest, navie Bayes and deep learning	10	SMS Spam Colleciton v1	Random forest	97.70
[21]	Nagwani	2014	NB, SVM, NMF LDA	40	SMS Spam Colleciton v1	SVM	93.45 (first-level) – 96.68 (second-level)
[22]	Uysal *et al.*	2013	kNN, SVM	2696(Tr)3185(En)	TurkishSMS, English SMS (425 spam, 450 legimate SMS)	SVM (Bow + SF2)SVM(BoW + SF1:SF6)	*F-1 Score*98 (Turkish)96 (English)
[23]	Uysal *et al.*	2012	Bayesian	10–50	SMS Spam Colleciton v1	—	—
[5]	Karasoy and Ballı	2016	Random forest, bagging, RandomSubspace	8	Turkish Real World Messages	Random forest	93.76

semantically constructed according to these separations. The performances in different separations are compared.

Ho *et al.* [2] proposed a solution by combining the graph-based text representation method and the kNN algorithm. The dataset is divided into different groups for performance evaluations; these groups are represented by graphs and created kNN entities. They compare it with their previous study and it is concluded that this suggestion is better with 98.9% accuracy.

Arifin *et al.* [15] focused on two important issues in data mining: classification and association. False positive (FP)-growth is used in determining the associations (frequent pattern), Naive Bayes is used in the classification process. Better results are achieved when FP-growth and NB are used together and 98.596% accuracy rate is achieved.

Waheeb *et al.* [16] analysed the performance of an artificial neural network (ANN) that they train with scaled conjugate gradient backpropagation algorithm. Gini index value is used in feature selection. They compare true positive (TP), FP, false negative (FN) and true negative (TN) values according to the number of selected features. The best result is obtained with 100 features. As a result of this study, 99.1% accuracy rate is achieved.

Ma *et al.* [17] proposed a message topic model (MTM) based on the latent semantic analysis probability theory and appropriate for SMS spam filtering. Compared to the existing spam SMS filtering technologies, MTM can eliminate the shortness problem of the messages. It often draws attention to the symbols seen in spam SMSs. A 97% accuracy rate is achieved with the MTM model.

Fernandes *et al.* [18] proposed a spam message filtering method with optimum-path forest classification model. SC%, BH%, ACC % and Matthews correlation coefficient values are used to compare the results. Even though the SVM provides best accuracy rate among the ANN-multi-layer perceptron (MLP), kNN, OPF with complete graph, OPF with kNN graph and SVM classification methods, OPF-based classifiers correctly classify all ham messages (BH%) 720 times faster.

Akbari and Sajedi [19] compared the LPBoost, AdaBoost, TotalBoost, LogitBoost, GentleBoost, RobustBoost, RusBoost, SVM and NB algorithms for SMS spam classification and propose the GentleBoost algorithm with the best accuracy rate of 98.30%. In addition, the probability of each word is calculated and compared with the others for feature extraction, and 124-word features are extracted. After that, it is reduced to 32-word features by removing the unused features without affecting the accuracy rate.

3.3 Machine Learning Methods

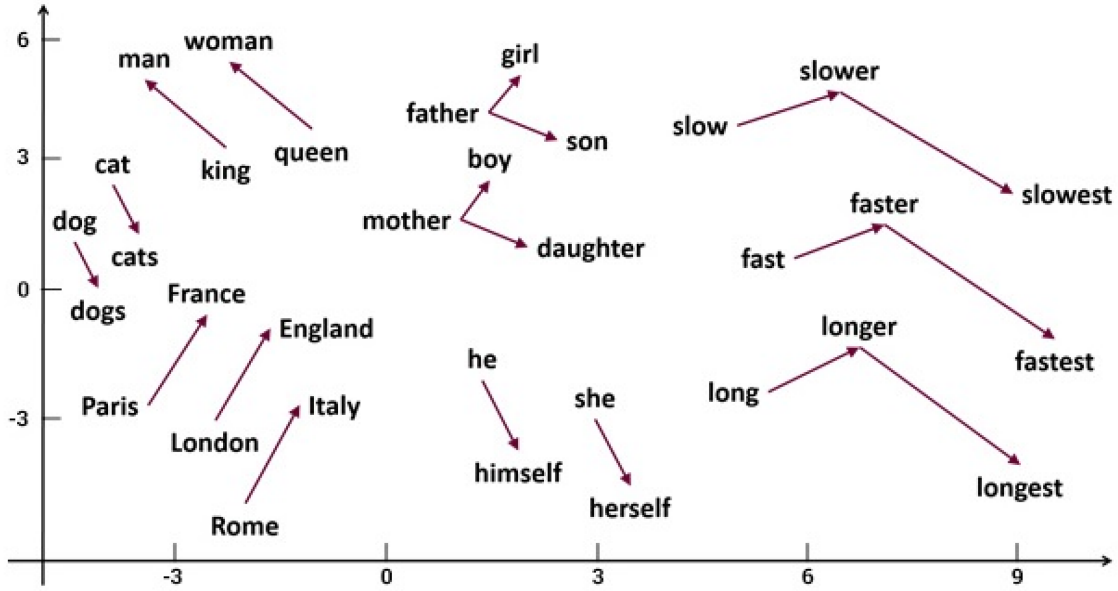

Fig. 1 *Semantic relations in vector space*

Suleiman and Al-Naymat [20] used the machine learning platform H2O in their study. The performances of NB, random forest (RF) and deep learning methods are compared with ten structural message features such as message length, word count, capital letter frequency, URL status etc. When Precision, Recall, F-measure and accuracy values are taken into consideration, the random forest method is proposed as the best algorithm with 97.7% accuracy.

Nagwani [21] proposed a two-level message classification method. According to this method, the incoming message is classified as spam and non-spam at the first level, and it is classified according to message priority at the second level. A total of 40 terms (20 spam messages and 20 ham messages), obtained by pre-processing from a total of 8629 terms extracted from the messages, are used as a feature. As a result of performance measurements, it is observed that the SVM algorithm produces the best result at both levels.

Uysal *et al.* [22] proposd kNN and SVM classification methods. Bow (Bag of Word) and collocation combinations of characteristic features are analysed in feature selection. (Bow – 2690 Turkish, 3179 English features are extracted + 6 characteristic features.) It is observed that the characteristic features give effective results even in different languages. The best result is achieved by SVM method with a 98% accuracy rate for the Turkish dataset and 96% accuracy rate for the English dataset.

Uysal *et al.* [23] developed an application that separates the incoming message as spam or ham message. First of all, the term ordering is performed with CHI2- and IG-based feature extractions for all terms in the dataset. Then, the classification is performed with feature subsets in different numbers (between 10 and 50) selected according to the binary and probabilistic model, and the results are compared.

In their study, Karasoy and Balli [5] created a new Turkish dataset and developed a mobile application for preventing unwanted messages. They use eight different features determined according to the characteristic properties of the messages in the dataset. Apart from spam and ham messages, the effect of the announcement messages on the result is also been compared. With 93.76% accuracy, random forest (with the two-class result) is proposed.

3 Methods

3.1 Deep learning

With advanced processor technologies, deep learning has become one of the most frequently preferred methods. This method, which is used in voice recognition, image recognition and natural language processing, is a set of algorithms that try to model data using model architectures composed of non-linear transformations [24]. Deep learning includes one or more hidden layers and it is a branch of machine learning used in deep neural network architectures.

One of the important processes of deep learning is to train neuron layers with Autoencoder (Diabolo) neural network. Autoencoder is a neural network which usually has a single hidden layer and is trained to produce similar output with the inputs it receives. It does not need labelled data to train; therefore, it provides learning without supervision. In addition, it has the ability to generate different representations of processed inputs. Owing to the advantages mentioned above, Autoencoder neural network is preferred in deep learning. Word2vec works like Autoencoder neural network. It may be trained using a big quantity of unlabelled input data [25].

3.2 Word2vec

Word2vec was published by Google in 2013 as a deep learning-based open source tool [26]. Thanks to this tool, words can be transformed into vectors and the distances between them can be calculated and an analogy can be established between the words. Word2vec is easy to understand and fast to train compared to other techniques. This technique is an easy to scale model that works with small and large datasets. Word2vec can find the semantic relationships between words in the sentence.

Examples made with the model consist of Google News texts [27], as shown in Fig. 1 [28], the words that are similar to each other are represented by close vectors. For example, dog and cat are represented in the areas close to each other. In addition, relations between words present close associations also in similar word phrases. Therefore, the relationship between ‘fast’, ‘faster’ and ‘fastest’ is similar to ‘long’, ‘longer’ and ‘longest’. Using the model created with Word2vec, the similarities of the words can be reached. In the model created with Google news, the closest word to ‘Man’ is ‘Woman’ (with a similarity value of 0.69). In a certain group of words, it can distinguish the irrelevant word with doesnt_match function. The command doesnt_match (‘blue red green yellow book’) returns the word ‘book’ in response.

Word2vec has a two-layer neural network that processes texts. CBOW and Skip-Gram are the main learning modes used in Word2vec. Fig. 2 shows the working principles of CBOW and Skip-Gram algorithms [29]. The CBOW model is used to predict contexts (neighbouring words) according to the word, and the Skip-Gram is used to predict the word from contexts [30–32].

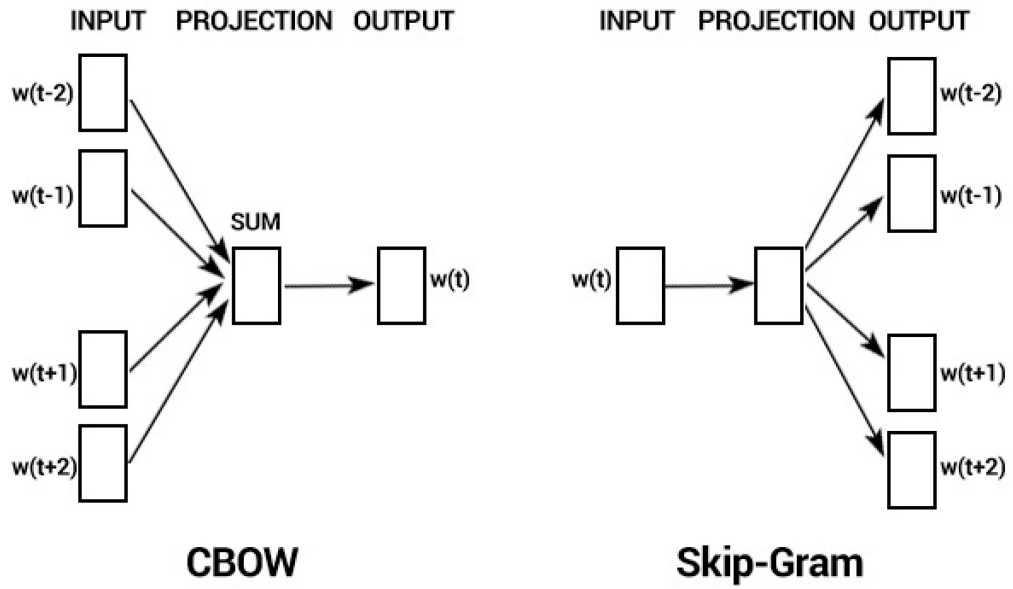

Fig. 2 *Working principles of CBOW and skip-gram models*

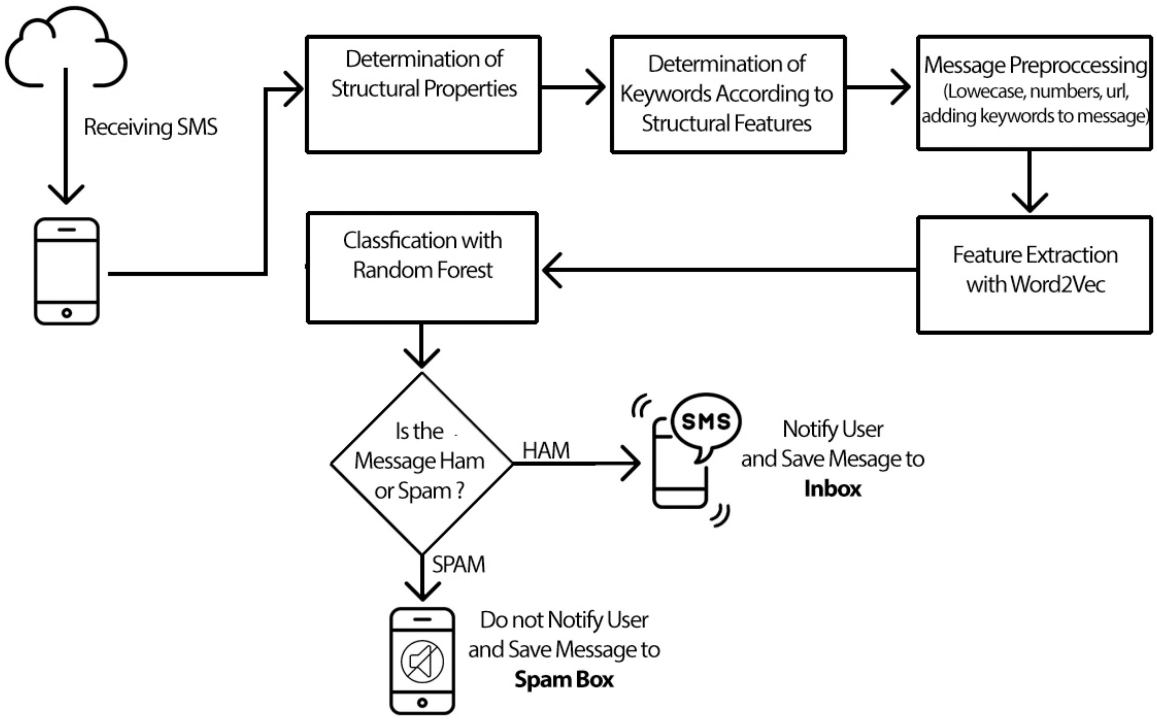

Fig. 3 *Flowchart of the designed system*

3.3 Machine learning methods

In this study, the model is built by utilising five different supervised machine learning methods: Random forest, Naive Bayes, MLP, logistic regression and SVM. These machine learning methods are chosen because they are frequently used in text classification and have successful accuracy rates in other studies. As these are widely known in the literature, they are briefly described as follows.

Random forest contains a combination of tree prognosticators. Every tree relies on the evaluations of a random vector sampled independently and with using the same allocation for every single tree of the forest. [33]

SVM is established upon a linear model. In linear model, firstly, the input space is transformed into a high-dimensional feature space by non-linear transformation. Then the optimal linear interface is searched in the new space [34].

In Naive Bayes algorithm, a probabilistic connection between features (attributes) and classes (labels) is actually employed. The naive part of this algorithm is that the assumption of independence between features may not carry out every time [35].

Logistic regression is a linear algorithm and utilised for binary classification. Parameters that maximise the likelihood of observing the sample values are selected for prediction in logistic regression while in ordinary regression, parameters that minimise the sum of squared errors are selected [36].

In MLP, neurons are interconnected in a few layers. It is a type of ANN and used to deal with the problems including the estimation (or fitting) of functions and the classification of non-linearly resolvable instances [37].

4 Word2Vec-based model for SMS classification

Fig. 3 shows the flowchart of the model designed in this study. Pre-processes of the incoming message, identification of features and classification process are explained in the sub-headings.

Table 2 Determining the structural features of the messages

No.	Message	ML	CWR	URL	Emoji	SMW	Class
1	I'm gonna be home soon and i don't want to talk about this stuff anymore tonight, k? I've cried enough today.	109	0.022	0	0	187	ham
2	SIX chances to win CASH! From 100 to 20,000 pounds txt> CSH11 and send to 87575. Cost 150p/day, 6days, 16 + TsandCs apply Reply HL 4 info	136	0.153	0	0	694	spam
3	URGENT! You have won a 1 week FREE membership in our £100,000 Prize Jackpot! Txt the word: CLAIM to No: 81010 T&C www.dbuk.net LCCLTD POBOX 4403LDNW1A7RW18	155	0.308	1	0	838	spam
4	I've been searching for the right words to thank you for this breather. I promise i won't take your help for granted and will fulfil my promise. You have been wonderful and a blessing at all times.	196	0.019	0	0	807	ham
5	I HAVE A DATE ON SUNDAY WITH WILL!!	35	0.929	0	0	72	ham
6	XXXMobileMovieClub: To use your credit, click the WAP link in the next txt message or click here>> http://wap.xxxmobilemovieclub.com?n=QJKGIGHJJGCBL	149	0.176	1	0	448	spam
7	Oh k...i'm watching here:)	26	0.043	0	0.043	0	ham
8	Eh u remember how 2 spell his name… Yes i did. He v naughty make until i v wet.	81	0.048	0	0	0	ham
9	Fine if that?s the way u feel. That?s the way its gota b	56	0.045	0	0	154	ham
10	England v Macedonia – dont miss the goals/team news. Txt ur national team to 87077 eg ENGLAND to 87077 Try:WALES, SCOTLAND 4txt/ú1.20 POBOXox36504W45WQ 16 +	155	0.242	0	0	187	spam

```
if (SMW < 100) { smsText += " MiniSpm"; }
else { smsText += " MaxiSpm MaxiSpm";}

if (ML < 140) { smsText += " MiniMessage MiniMessage"; }
else { smsText += " MaxiMesssage"; }

if (CWR < Convert.ToDecimal("0.19")){ smsText += " MiniUpperCase"; }
else if (CWR < Convert.ToDecimal("0.6")) { smsText += " MidiUpperCase"; }
else { smsText += " MaxiUpperCase"; }

if (URL > 0) { smsText += " HasURL"; }
else { smsText += " NoURL"; }

if (Emoji > 0) { smsText += " HasEmoji"; }
else { smsText += " NoEmoji"; }
```

Fig. 4 *Code block used to add keywords to the message*

4.1 Dataset

SMS Spam Collection dataset [8] is used in this study. There are two classes in this dataset: spam and ham. Spam is described as unwelcome messages sent with the aim of commercial benefit or simply causing detriment or discomfort to users [38]. Ham has generally desired messages like daily messaging among people. The dataset consists of 5574 lines of short messages consisting of 4827 ham and 747 spam messages. About 70% of the dataset is reserved for testing and 30% for training.

4.2 Preparation of data

4.2.1 Determination of structural properties: Positive contributions of structural properties of messages are observed in content-based classification studies [5]. In this context, it is determined by the features such as message length (ML), capital letters (CWR), emotional expression frequency (Emoji) and URL, which can help in classification. It is observed that spam messages are close to 160 characters long or more, they often contain a URL and have high capitalisation usage rates. In ham messages, the use of emotional expressions emerges as a feature differentiating the usage.

Furthermore, 80 words frequently used in spam messages are selected from the dataset and scored from 1 to 80 according to their frequencies.

Some examples of common words of spam class are given below:

free, the, for, txt, have, from, mobile, com, stop, claim, reply, of, prize, our, only, won, cash, uk, win, send, nokia, new, urgent etc.

The Spam Message Weight (SMW) value of each message is calculated by adding word scores according to the present status of these selected words in the messages. Table 2 shows examples of the values of the structural properties of the messages.

4.2.2 Transforming of structural features into keywords: It is foreseen that the structural features in the proposed solution are to be added to the messages as text and analogies are to be established with these texts and the words in the messages.

The keywords added to messages according to their structural characteristics are:

- ML -> MiniMessage, MaxiMessage
- CWR->MiniUpperCase, MidiUpperCase, MaxiUpperCase
- URL-> HasURL, NoURL
- Emoji-> HasEmoji, NoEmoji
- SMW-> MiniSpam, MaxiSpam

The code snippet in Fig. 4 is used while the features are being translated to text. Some keywords (MaxiSpam, MiniMessage), which are determined as a result of the experiments, are added to the message second time and their weights increased.

4.3 Organising the message according to the model (preparation)

The sensitivity of the features received from the Word2Vec model is increased by adding the features of the message structure (previously defined) to the dataset as a keyword. The SMS features

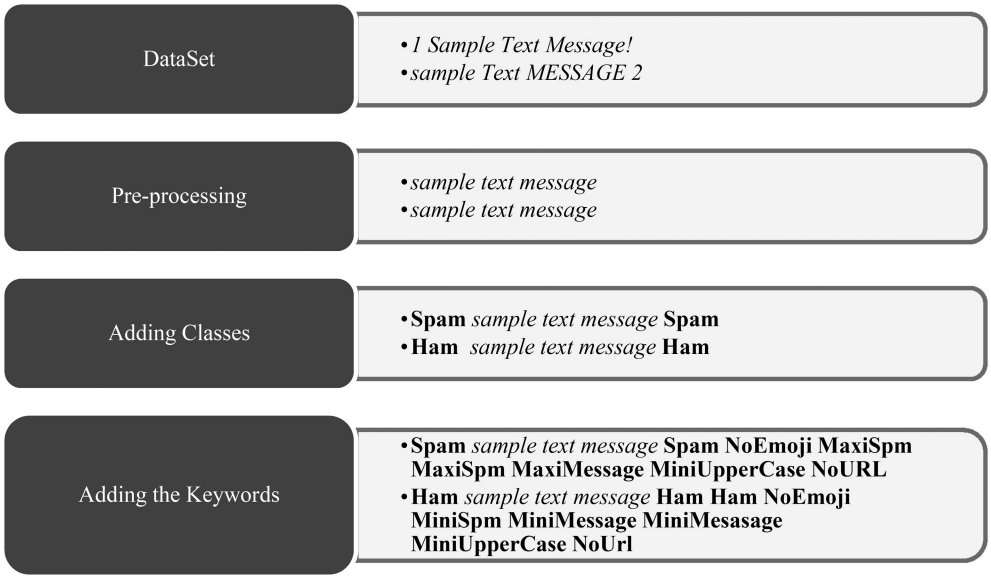

Fig. 5 *Creating a new dataset with keywords*

```
1 import gensim
2 import logging
3 sentences = gensim.models.word2vec.LineSentence("datasetwithKeyWords.txt", max_sentence_length=10000)
4 model = gensim.models.Word2Vec(sentences, size=1600, window=15, min_count=3, workers=5)
5 model.save("ModelMessages.w2v")
```

Fig. 6 *Python code used to create Word2Vec model*

Table 3 Word2Vec model creation parameters

Parameters	Explanations
size	the number of dimensions of feature vectors
window	the highest distance between the actual and estimated word within a sentence
min_count	disregards all words having an overall frequency under this value
workers	number of worker threads to train the model
sg	selecting the training algorithm. If 1, CBOW is employed, otherwise, skip-gram is used
max_vocab_size	restricts the RAM throughout vocabulary construction; if there are more unique words than this value, then it cuts the sparse ones

used in this process are: SMW, Message Length, Capital Letter Frequency and URL Status.

In order to create the Word2Vec model, the dataset is modified with the steps in Fig. 5. First of all, the digits, punctuation marks and symbols are removed from the dataset. All characters are translated into lower case and URLs are removed from the message. Then, the class of the message is added to the beginning and end of the message to create a strong connection with the words. The keywords that specify the structural characteristics of the message are added to the end of the message.

4.4 Creating a model with Word2Vec

An open source Phyton library Gensim [39] is used to create the Word2Vec model from the prepared dataset. The model is formed with the prepared dataset and the code block in Fig. 6. As seen in Fig. 6, Gensim and logging libraries are imported in first and second rows, the command in the third row reads the messages line by line from the file, Word2vec model is created by using the parameters in the fourth row and the model is saved as a file in the fifth row. Explanations of the parameters used are given in Table 3.

4.5 Feature extraction with Word2Vec

Word2Vec model is created after the preliminary processes. Using this model, distance values of the words of each message to spam and ham keywords are calculated by Word2vec. Two new features are created by adding these values separately according to the classes.

Message Sample:

‘URGENT! You have won a 1 week FREE membership in our £100000 Prize Jackpot! Txt the word: CLAIM to No: 81010 T&C www.dbuk.net LCCLTD POBOX 4403LDNW1A7RW18’

After preprocessing:

‘urgent you have won week free membership in our prize jackpot txt the word claim to no t c lccltd pobox ldnw a rw NoEmoji MaxiSpm MaxiSpm MaxiMessage MidiUpperCase HasURL’

Table 4 shows how distances to spam and ham keywords are calculated for a selected sample message. A feature of each word in the message is created by founding and adding distances of each word from keywords. If a word which is not in the model is included in the message, its distance to the keywords is 0 and it is excluded. Since the word ‘jackpot’ is not in the dictionary, the letters such as ‘t’, ‘c’, ‘a’ are not processed because they are single characters.

Table 5 shows ten sample records from the final state of the dataset prepared for use in the classification.

Fig. 7 shows the distribution of w2vSpam and w2vHam features.

4.6 Classification

In the classification phase, keywords are determined according to the structural features of the incoming SMS. This message is then passed through a preliminary process. The message text is converted into lower cases. Numbers, punctuation and URLs are removed from the message, and the specified keywords are added to the message. With the Word2Vec model prepared in the study, two new features are found by adding the distances of the words of the message to spam and ham. These features are classified with

Table 4 Sample message feature extraction

Word	Distance to spam	Distance to ham
urgent	0.975194246506	0.438388839784
you	0.655218728478	0.733712784782
have	0.879464504206	0.637038282437
won	0.971993393431	0.447945280534
week	0.927804236488	0.609127003671
free	0.995738243173	0.294745526108
membership	0.986576426483	0.30541150787
İn	0.659460629677	0.844060527719
our	0.943573380642	0.561459519238
prize	0.992812816078	0.327789914867
Txt	0.975739475582	0.489539918416
the	0.823232509864	0.646893576175
word	0.964968165891	0.529489068039
claim	0.994731596072	0.368184471241
to	0.948806050559	0.442867815647
No	0.749500382947	0.841420898899
pobox	0.907761091268	0.666196932476
ldnw	0.98065959655	0.47467678971
MaxiSpm	0.754575023181	0.628513943787
MaxiSpm	0.754575023181	0.628513943787
MaxiMessage	0.925637560272	0.546333218334
MidiUpperCase	0.888145625486	0.630623345897
HasURL	0.959197094166	0.524208530277
NoEmoji	0.721308231221	0.612132031243
total	21.336674031402	13.229273670938

Table 5 Dataset to be used in classification

Message ID	Feature 1 (distance value to spam label)	Feature2 (distance value to ham label)	Class of message
1	12.416	19.065	ham
2	19.295	15.276	spam
3	22.615	13.213	spam
4	23.213	25.430	ham
5	6.549	11.355	ham
6	18.772	13.564	spam
7	3.964	10.531	ham
8	5.955	11.960	ham
9	5.823	10.573	ham
10	19.265	14.924	spam
…	…	…	…

the help of classification algorithm. The message classified according to the model designed in Fig. 3, is sent to the Spam box without notifying the user as shown in the application developed on the Android operating system in Fig. 8. If the incoming message is classified as ham, the user is warned and the message is moved to the inbox.

5 Experimental results

The performed study is carried out in two stages as the creation of Word2Vec model and classification algorithm. In these two stages, a desktop computer with I7 processor and 8 GB of RAM is used. The Word2Vec model is prepared in Python programming language [40] over Gensim library [39]. Classification procedures are performed using Weka program [41].

Table 6 shows the results reached by random forest, MLP, SVM, logistic regression and Naive Bayes algorithms using two different methods for feature extraction.

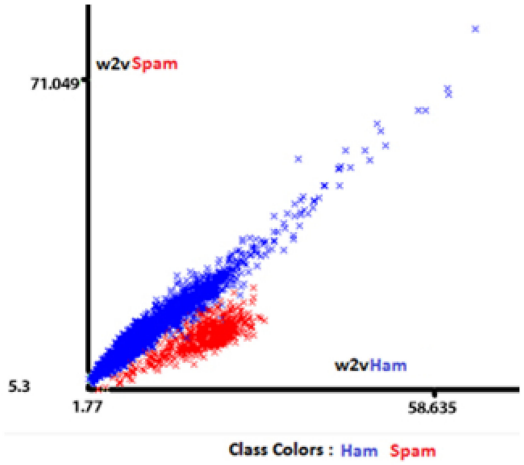

Fig. 7 *Distribution of created features*

5.1 Evaluation metrics

Precision, F-measure, Recall, Roc and Accuracy parameters are examined while evaluating the classification algorithms. Although accuracy (ACC) is the ratio of correct estimations to all estimations, it is not sufficient alone to evaluate the work. For this reason, Precision, Recall, F-measure, ROC and RMSE values are also examined in this study. Equation (1) is used for calculation of accuracy rate [42]. TP value is the number of spam messages classified as spam. FP value is the number of messages classified as spam but they are not spam actually. TN is the number of ham messages classified as ham. FN is the number of spam messages classified as ham. These values are shown in Table 7 (confusion matrix).

$$\text{Acc} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} \times 100 \tag{1}$$

Precision shows the closeness of two or more measurements to each other and it gives the ratio of positive predictions. It is calculated using (2). The Recall is calculated with the proportion of the sum of accurately classified positive instances to the sum of positive instances and shows the precision as shown in (3). F-measure is calculated using (4) and used to evaluate these two criteria together. It is the harmonic mean of Recall and Precision [43]:

$$\text{Precision}(p) = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2}$$

$$\text{Recall}(r) = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

$$\text{F} - \text{measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} \times \text{Recall}} \tag{4}$$

5.2 Results and discussion

In Table 6, only Word2Vec part shows the results reached by the features found only by calculating distances of words to the types (spam, ham) without considering the structural features of the message (message length, URL status etc.). According to Table 6, MLP has the best correct accuracy rate with 99.64% in only Word2Vec part.

In Table 6, structural features + Word2Vec part shows the results of the classification algorithms which are compared according to the distance values of the words (to the types) in the message after the structural properties of the message are added to the message as a keyword. According to Table 7, random forest has the best accuracy rate with 99.64%.

Although the correct accuracy rates of the MLP and RF methods in Table 6 seem to be equal, when the confusion matrices in Table 7 are examined, it is seen that the RF algorithm in Table 7 Word2Vec + structural features part does not mark any ham

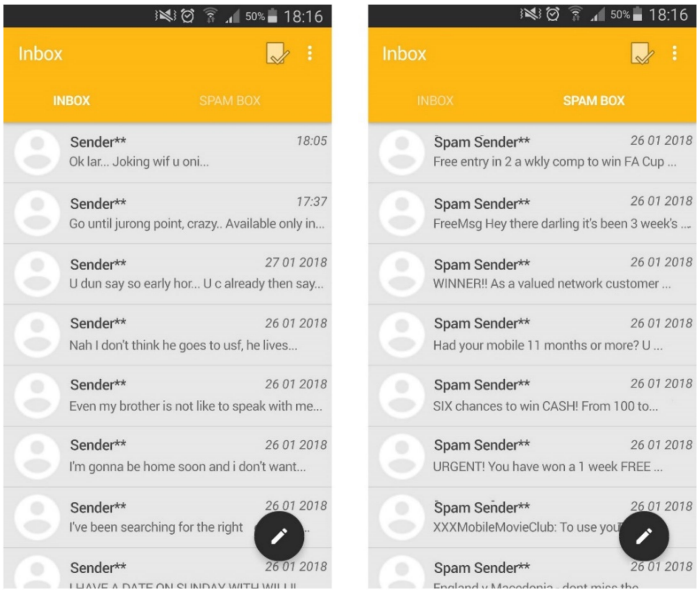

Fig. 8 *Screenshot of the application*

Table 6 Comparison results of classification algorithms

	SPAM			HAM			TOTAL		
	Precision	Recall	F-Mea.	Precision	Recall	F-Mea.	ROC(AUC)	RMSE	ACC, %
Only Word2Vec									
Random forest	0.970	0.966	0.968	0.994	0.995	0.995	0.999	0.0795	99.1029
MLP	0.983	0.991	0.987	0.999	0.997	0.998	1.000	0.0584	99.6411
SVM	1.000	0.931	0.964	0.989	1.000	0.994	0.966	0.0978	99.0431
Logistic regression	0.983	0.987	0.985	0.998	0.997	0.998	1.000	0.0563	99.5813
Naive Bayes	0.510	0.534	0.522	0.924	0.917	0.921	0.880	0.3004	86.4234
Structural features + Word2Vec									
Random forest	1.000	0.974	0.987	0.996	1.000	0.998	0.997	0.0651	99.6411
MLP	0.950	0.983	0.966	0.997	0.992	0.994	1.000	0.0731	99.0431
SVM	0.991	0.974	0.983	0.996	0.999	0.997	0.986	0.0692	99.5215
Logistic regression	0.991	0.970	0.980	0.995	0.999	0.997	1.000	0.0629	99.4617
Naive Bayes	1.000	0.664	0.798	0.949	1.000	0.974	0.958	0.2008	95.3349

Table 7 Confusion matrices

	Random forest		MLP	
Classified as	Spam	Ham	Spam	Ham
Confusion matrix of only Word2Vec				
Spam	224	8	230	2
Ham	7	1433	4	1436
Confusion matrix of Word2Vec + structural features				
Spam	226	6	228	4
Ham	0	1440	12	1428

messages as spam. Considering the real-life experiences in a message filtering application, the FP value, which shows the measure of ham messages that are incorrectly classified, is an important measure and a condition to be considered. In this study, Word2Vec which has low FP value and random forest method are proposed by using the structural features together.

In classification problems, the accuracy rate is an important criterion for giving the success rate of classification. According to Table 6, MLP and random forest obtain the highest ACC value with 99.6411 being the accuracy.

When the message filtering problem is addressed, the precision value becomes an important criterion. That is, marking ham messages as spam is the last desired state. Table 6 also shows that SVM, random forest and navie Bayes methods have the highest precision values of 1.000. MLP classifies ham message as more spam because it has a precision of 0.983.

When these values are taken into consideration, random forest is determined to be the most appropriate method for this problem and it is suggested. Since the random forest uses more than one tree, it has lower variance than other methods. This means that the chance of a classifier which is not working well due to the relationship between train and test data is greatly reduced.

Table 8 shows the previous studies carried out with SMS Spam collection v1 dataset. It is observed that the correct classification percentages of the studies shown vary between 92 and 98.9%. With using the features extracted via the help of deep learning-based tool named Word2Vec used in this study, more successful correct classification percentage is achieved with the random forest method getting 99.64% as accuracy compared to other studies.

To check the improvement obtained in this study and to confirm the difference from other approaches, the statistical significance is measured by using N-way analysis of variance test. The formal test is

H_0: Proposed approach in this study has the same accuracy rate as other approaches.

H_1: They have significantly different accuracy rates.

At a significance level of 0.05, the achieved p-value reported across the different classification methods is $p < 0.01$ and the null hypothesis is rejected. It indicates that there is a statistically significant difference between the accuracy rate by proposed approach in this study and accuracy rates achieved by the other approaches.

Table 8 Previous studies using SMS Spam Collection v1

Authors	Dataset	Recommended method	Accuracy, %
Almeida *et al.* [8]	SMS Spam Col. v1	SVM	97.5
Bozan *et al.* [13]	SMS Spam Col. v1	SVM	98.61
Ho *et al.* [2]	SMS Spam Col. v1	Graph-based KNN	98.9
Fernandes *et al.* [18]	SMS Spam Col. v1	OPF with complete	92.23
Akbari and Sajedi [19]	SMS Spam Col. v1	GentleBoost	98.30
Suleiman and Al-Naymat [20]	SMS Spam Col. v1	Random forest	97.7
Nagwani [21]	SMS Spam Col. v1	SVM	96.68
Proposed model in this study	SMS Spam Col. v1	Word2Vec + RandomForest	99.64

6 Conclusion

In this study, a content-based classification solution is proposed to prevent spam SMS which is an important problem nowadays. Unlike previous studies, the proposed model in this study uses semantic relationships between words in the SMS message. Words in the message are transformed into vectors and the distances between them are calculated and an analogy is established between the words by using Word2Vec.

Keywords are determined according to the structural properties of the messages in the dataset and then new dataset is created by adding these keywords to the dataset. The Word2Vec model is constructed from the newly created dataset formed with Word2Vec library and the features are extracted for each message with the help of this model. Using the generated features, common classification successes of learning algorithms in previous studies are examined.

As a result, the random forest is selected as the best method with the accuracy rate of 99.6411%. Considering the previous studies performed with the same dataset, it is observed that the method proposed in this study achieved more successful and statistically significant results in the correct classification percentage.

The proposed approach can be integrated into existing SMS applications to detect spam SMS. This approach may also be used in similar communication applications such as E-mail, WhatsApp, Messenger, Viber etc. to prevent spam messages. As future work, tuning the input parameters of the classification algorithms would be convenient, also it would be advantageous to carry out the cross-validation of the developed models.

7 References

[1] Castiglione, A., De Prisco, R., De Santis, A.: 'Do you trust your phone?'. E-Commerce and Web Technologies, Linz, Austria, September 2009, pp. 50–61

[2] Ho, T., Kang, H., Kim, S.: 'Graph-based KNN algorithm for spam SMS detection', *J. Univers. Comput. Sci.*, 2013, **19**, (16), pp. 2404–2419

[3] Church, K., Oliveira, R.D.: 'What's up with Whatsapp?: comparing mobile instant messaging behaviors with traditional SMS'. 15th Int. Conf. Human-Computer Interaction with Mobile Devices and Services, Mobile HCI, Munich, Germany, 2013

[4] Delany, S.J., Buckley, M., Greene, D.: 'SMS spam filtering: methods and data', *Expert Syst. Appl.*, 2012, **39**, (10), pp. 9899–9908

[5] Karasoy, O., Balli, S.: 'Developing mobile application for content base spam SMS filtering and comparison of classification algorithms'. Int. Artificial Intelligence and Data Processing Symp., Malatya, Turkey, September, 2016, pp. 47–53

[6] Junaid, M.B., Farooq, M.: 'Using evolutionary learning classifiers to do mobile spam (SMS) filtering'. Proc. of Genetic and Evolutionary Computation Conf., Dublin, Ireland, July 2011, pp. 1795–1802

[7] 'SMS spam collection'. Available at http://archive.ics.uci.edu/ml/datasets/SMS+Spam+Collection, accessed January 2018

[8] Almeida, T.A., Hidalgo, J.M.., Yamakami, A.: 'Contributions to the study of SMS spam filtering: new collection and results'. Proc. 11th ACM Symp. Document engineering, New York, USA, September 2011, pp. 259–262

[9] 'SMS spam corpus v. 0.1'. Available at http://www.esp.uem.es/jmgomez/smsspamcorpus, accessed January 2018

[10] 'NUS SMS corpus'. Available at http://www.comp.nus.edu.sg/entrepreneurship/innovation/osr/corpus, accessed January 2018

[11] 'DIT SMS spam dataset'. Available at http://www.dit.ie/computing/research/resources/smsdata, accessed January 2018

[12] 'Turkish SMS'. Available at http://ceng.anadolu.edu.tr/par, accessed January 2018

[13] Bozan, Y.., Çoban, Ö., Özyer, G.T., *et al.*: 'SMS spam filtering based on text classification and expert system'. 23nd Signal Processing and Communications Applications Conf. (SIU), Malatya, Turkey, May 2015, pp. 2345–2348

[14] He, H., Watson, T., Maple, C., *et al.*: 'A new semantic attribute deep learning with a linguistic attribute hierarchy for spam detection'. Int. Joint Conf. Neural Networks (IJCNN), Anchorage, AK, USA, May 2017, pp. 3862–3869

[15] Arifin, D.D., Shaufiah, , Bijaksana, M.A.: 'Enhancing spam detection on mobile phone short message service (SMS) performance using FP-growth and Naive Bayes classifier'. IEEE Asia Pacific Conf. Wireless and Mobile (APWiMob), Bandung, Indonesia, September 2016, pp. 80–84

[16] Waheeb, W., Ghazali, R., Deris, M.M.: 'Content-based SMS spam filtering based on the scaled conjugate gradient backpropagation algorithm'. 12th Int. Conf. Fuzzy Systems and Knowledge Discovery (FSKD), Zhangjiajie, China, August 2015, pp. 675–680

[17] Ma, J., Zhang, Y., Liu, J., *et al.*: 'Intelligent SMS spam filtering using topic model'. Int. Conf. Intelligent Networking and Collaborative Systems (INCoS), Ostrawva, Czech Republic, September 2016, pp. 380–383

[18] Fernandes, D., Costa, K.A.P., Almeida, T.A., *et al.*: 'SMS spam filtering through optimum-path forest-based classifiers'. IEEE 14th Int. Conf. Machine Learning and Applications (ICMLA), Miami, FL, USA, December 2015, pp. 133–137

[19] Akbari, F., Sajedi, H.: 'SMS spam detection using selected text features and boosting classifiers'. 7th Conf. Information and Knowledge Technology (IKT), Urmia, Iran, May 2015, pp. 1–5

[20] Suleiman, D., Al-Naymat, G.: 'SMS spam detection using H2O framework', *Procedia Comput. Sci.*, 2017, **113**, pp. 154–161

[21] Nagwani, N.K.: 'A Bi-level text classification approach for SMS spam filtering and identifying priority messages', *Int. Arab J. Inf. Technol.*, 2017, **14**, (4), pp. 473–480

[22] Uysal, A.K., Gunal, S., Ergin, S., *et al.*: 'The impact of feature extraction and selection on SMS spam filtering', *Elektron. Elektrotech.*, 2013, **19**, (5), pp. 67–72

[23] Uysal, A.K., Gunal, S., Ergin, S., *et al.*: 'A novel framework for SMS spam filtering'. Int. Symp. Innovations in Intelligent Systems and Applications, Trabzon, Turkey, July 2012, pp. 1–4

[24] Bılgıç, A., Kurban, O.C., Yildirim, T.: 'Face recognition classifier based on dimension reduction in deep learning properties'. 25th Signal Processing and Communications Applications Conf. (SIU), Antalya, Turkey, May 2017, pp. 1–4

[25] Enríquez, F., Troyano, J.A., Lopez-Solaz, T.: 'An approach to the use of word embeddings in an opinion classification task', *Expert Syst. Appl.*, 2016, **66**, pp. 1–6

[26] Zhang, D., Xu, H., Su, Z., *et al.*: 'Chinese comments sentiment classification based on word2vec and SVM', *Expert Syst. Appl.*, 2015, **42**, (4), pp. 1857–1863

[27] 'Word2vec Tutorial'. Available at https://rare-technologies.com/word2vec-tutorial, accessed January 2018

[28] 'NLP with gensim (word2vec)'. Available at http://www.samyzaf.com/ML/nlp/nlp.html, accessed January 2018

[29] 'Introduction to Word2Vec'. Available at https://deeplearning4j.org/word2vec, accessed January 2018

[30] Wensen, L., Zewen, C., Jun, W., *et al.*: 'Short text classification based on wikipedia and Word2vec'. 2nd IEEE Int. Conf. Computer and Communications (ICCC), Chengdu, China, October 2016, pp. 1195–1200

[31] Mikolov, T., Sutskever, I., Chen, K., *et al.*: 'Distributed representations of words and phrases and their compositionality', *Proc. Adv. Neural Inf. Process. Syst.*, 2013, **26**, pp. 3111–3119

[32] Mathew, K., Issac, B.: 'Intelligent spam classification for mobile text message'. Proc. 2011 Int. Conf. Computer Science and Network Technology, Harbin, China, December 2011, pp. 101–105

[33] Breiman, L.: 'Random forests', *Machine Learning*, **43**, (1), (Springer, Berlin, Heidelberg, 2001), pp. 5–32

[34] Wang, Z., Qu, Z.: 'Research on Web text classification algorithm based on improved CNN and SVM'. 2017 IEEE 17th Int. Conf. Communication Technology (ICCT), Chengdu, China, 2017, pp. 1958–1961

[35] Silalahi, M., Hardiyati, R., Nadhiroh, I.M., *et al.*: 'A text classification on the downstreaming potential of biomedicine publications in Indonesia'. Int. Conf. Information and Communications Technology (ICOIACT), Yogyakarta, Indonesia, 2018, pp. 515–519

[36] Sethi, P., Bhandari, V., Kohli, B.: 'SMS spam detection and comparison of various machine learning algorithms'. Int. Conf. Computing and Communication Technologies for Smart Nation (IC3TSN), Gurgaon, 2017, pp. 28–31

[37] Liu, Y., Liu, S., Wang, Y., *et al.*: 'A stochastic computational multi-layer perceptron with backward propagation', *IEEE Trans. Comput.*, 2018, **67**, (9), pp. 1273–1286

[38] Saab, S.A., Mitri, N., Awad, M.: 'Ham or spam? A comparative study for some content-based classification algorithms for email filtering'. 17th IEEE Mediterranean Electrotechnical Conf., Beirut, 2014, pp. 339–343

[39] 'GENSIM'. Available at https://radimrehurek.com/gensim/models/word2vec.html, accessed January 2018

[40] 'Phyton'. Available at https://www.python.org/, accessed January 2018

[41] 'Weka'. Available at https://www.cs.waikato.ac.nz/~ml/weka/, accessed January 2018
[42] Balli, S., Sağbas, E.A.: '*The usage of statistical learning methods on wearable devices and a case study: activity recognition on smartwatches, advances in statistical methodologies and their application to real problems*' (InTech, Rijeka, Croatia, 2017)
[43] Witten, I.H., Frank, E., Hall, M.A.: '*Data mining: practical machine learning tools and techniques*' (Elsevier, Burlington, 2011, 3rd edn.)