Journal of Engineering and Applied Sciences 13 (6): 1331-1337, 2018
ISSN: 1816-949X

Junk Messages Filtering Using Fuzzy-Rough Set with Latent Semantic Analysis

[1]Raghad M. Hatim and [2]Ghaidaa A. Al-Sultany
[1]Department of Software, [2]Department of Information Network,
Babylon University, Babel, Iraq

Abstract: Data mining has been effectively used in SMS classification which attracted the attention of researches in the last decades. The main objective of this study is to improve classification on an SMS junk using fuzzy-rough set approach with latent semantic analysis as feature selection which aims to reduce the error factor according to coverage factor reach to zero. The results are compared with the traditional fuzzy-rough set quick rules algorithm, decision tree and Naive Bays algorithms. The proposed method has satisfied better results for the classification rate according to the precision, F-measure, recall performance measures.

Key words: SMS, fuzzy-rough set, latent semantic analysis, decades, precision, recall

INTRODUCTION

The Short Message Services (SMS) is one of the important text communication mode due to the rapid increase of the number of mobile phone's users around the world. In addition, it is an effective communication service for its low cost and simplicity, hence, it has become the target for spammers to target the subscribers (Bodic, 2005).

Junk SMS is an unsolicited bulk SMS that is delivered to mobile phone for credit opportunities of banks, promotion and discount announcements of stores and new tariffs of communications services providers (Wang *et al*., 2010). Mobile junk messages reach mobile phones' users without their consent. There are different kinds of these messages such as advertisements and phishing messages. The spread of junk messages caused many problems such as wasting time and resources, full SMS inbox, reduction advertising effectiveness and sometimes lead to mobile failure if those messages were harm and might include a destructive link leading to a web page containing malicious programs (Wang *et al*., 2010; Zhu and Tan, 2011; Wen-Liang *et al*., 2009). Therefore, junk SMS have become a wide area for researches to develop and improve many applications for detection the spam messages. Topic modeling can be used as features selection techniques where it was proved as an effective manner against the noisiness and high data dimensionality such text data. Moreover, it can extract discriminative features without effecting on the significant data as it treats with the semantic relation among the features.

Latent semantic analysis as one of the topic modeling techniques has employed in this research for features selection. It has approved more appropriation for the SMS junk messages filtering compared with the present SMS junk filtering techniques. Furthermore, it can reduce the sparsity difficulty in the short text messages classification and giving more attention to the types of symbols, idioms and observation that are popular with the text messages.

Literature review: Content based filtering solutions have been proved to be effective against email which is larger in size in comparison with SMS message. Abbreviations and acronyms are used more frequently in SMS messages that increase the level of ambiguity. Thus, it is difficult to adopt the email filters without any modification and it should employ effective mechanisms that appropriate with the nature of SMS message (Delany *et al*., 2012; Almeida *et al*., 2013).

The research, Zhang and Wang (2009) have been employed Bayesian classification and junk filtering mechanism that is relied on the black list, white list and keyword, this filtering of junk SMS turns to Study based on the text content of SMS messages. Uysal *et al*. (2012) proposed a novel approach for SMS junk classification which applying two different features selection approaches depend on the information gain and Chi-square metrics to detect Discriminative features representing SMS messages. After that these features subsets are then employed in Bayesian-based classifiers, this approach accomplish outcomes with the highest overall precision is obtained as 90.17% by the binary classification. The researchers Sakshi Agarwal, Sanmeet Kaur, Sunita Garhwal have been applied two classifiers Support Vector (SVM) and Multinomial Naive Bays (MNB) and both classifiers gave encouraged outcomes when implemented on the Indian junk SMS corpus but the time that required for SMS junk filtering with the support vector machine algorithm was less than the time required

Corresponding Author: Raghad M. Hatim, Department of Software, Babylon University, Babel, Iraq

with the multinomial Naive Bays (Agarwal *et al*., 2015). Valles and Rosso (2011) have evaluated the accomplishment executed by plagiarism discovery tools while using as filters for SMS junk messages. They have done experiments on the SMS junk set (Almeida *et al*., 2011) and evaluated the outputs with the ones executed by the well-known CLUTO approach. Their major conclusion is that plagiarism discovery tools have discovered a perfect number of semi-duplicate SMS junk messages and exceeded the CLUTO tool. The research Akbari and Sajedi (2015) suggest an algorithm gentle boost algorithm for spam SMS detection because they have unbalance data and found that gentle boost performed that better than other. Boosting classifier and lead to the batter performance the reason for that this might be a batter method is that it works batter for unbalanced and binary classification. Biggio *et al*. (2011) added a cost function to a Naive Bays filter which assigned a high cost to false positives with Gini index feature selection. The Gini index has a disadvantage that it selects large number of features and Naive Bays algorithm does not take into account the dependency among the words.

MATERIALS AND METHODS

Fuzzy rough set mathematical tool for classification was implemented as shown in Fig. 1. The system performance has been enhanced with the latent semantic analysis that is applied as the attributes selection method. There were four processes in the proposed method.

Data representation and preprocessing: The English SMS Spam Corpus v.1 (Al-Hasan and El-Alfy, 2015; Almeida *et al*., 2011) was used in this research. It consists of a subgroup of 421 messages. This subgroup has 372 messages with 'ham-clean' type and 49 messages with 'spam' type. The average of the characters count per message is 4.44 characters and the average number of words per message is 15.725. The preprocessing process consists of three stages.

Tokenization: Splitting each message into semantically coherent parts (such as words, symbol, etc.).

Vector space model: Transformation each message into a feature-weight vector where the features are already defined tokens and their weight can be occurrences, frequencies, TF-IDF.

Stop words removal: Stop words or excluded words are words which are excluded before the automatic language processing of data (texts). These words are repeated in

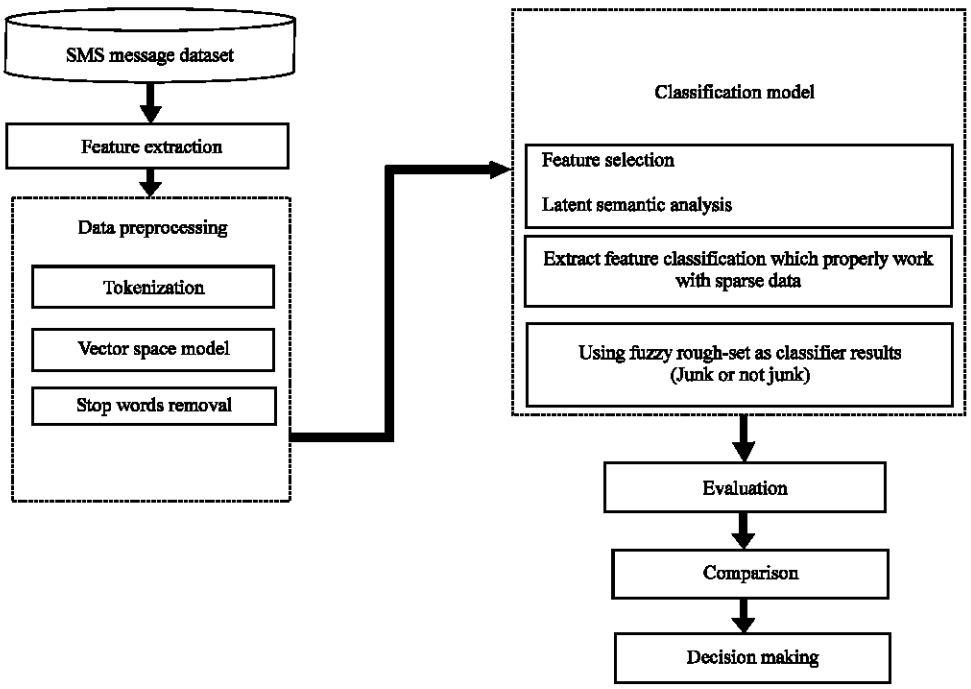

Fig. 1: Architectural design for junk filtering

texts such as (in, from, to, ...) and it is advisable to ignore them and not to index them to improve the search and thus improve the performance of the classifier.

Feature selection: Features selection, attributes selection or variable selection methods are the task of choosing a subgroup of related features (predictors, variables) for process in classifier build with an evaluation metrics to evaluate the diverse feature subsets. Feature selection methods are employed for several purposes such as simplify of classifiers to be easy for Explanation by researchers and users, little training periods, to avert the curse of dimensionality, improved generalization by limiting of overfitting. The main idea from utilizing a feature selection method is that the data includes several features which are either irrelevant or redundant and then can be excluded without a great losing of information. Irrelevant or redundant features are two different concepts where one related feature can be redundant in the existence of other related feature that it is robustly linked. The clearest method is to check every subgroup of features to find the set that possesses the lowest error rate. This is an exhaustive search of the space and is computationally intractable for all but the shorter of feature collections.

Latent Semantic Analysis (LSA): LSA technique is a new method in SMS messages classification. Commonly, LSA factorized relations between a word and topics included in an unstructured set of SMS messages. It is named latent semantic analysis because of its capability to connect semantically relevant words that are underlying in a message. LSA provides a collection of topics that is shorter in size than the original collection, relevant to messages and words (L'Huillier *et al*., 2010; Landauer *et al*., 1998). It applied SVD (Singular Value Decomposing) to determine pattern between the words and topics included in the message and identify the relations between messages. The way generally indicated to as concept searches. It has capacity to elicit the meaningful content of a set of messages through creating links between those words which happen in like contexts. LSA is generally applied in text clustering targets and page retrieval systems. LSA overcomes two of the more suspicious keyword queries: several terms which have identical concepts and terms which possess more than one concept (Landuaer *et al*., 2011). The Latent Semantic Analysis (LSA) algorithm is implemented for features selection as it has the ability for analyzing the relationship among messages. It provides a set of topics related to the messages content (Galvez and Gravano, 2017). In latent semantic analysis, the words that are similar in the meanings would be gathered in similar groups of texts. The latent semantic analysis consists of two steps.

Table 1: Describes part of results after perform preprocessing phase on 421 SMS message

No	Right	Bawling	Eyes	Failing	Feel	Like
M1	0.693147	0.693147	0.693147	0.693147	1.098612	1.098612
M2	0	0	0	0	0	0
M3	0	0	0	0	0	0.693147
M4	0	0	0	0	0	0.693147
M5	0	0	0	0	0	0.693147
M6	0	0	0	0	0	0
M7	0	0	0	0	0	0
M8	0	0	0	0	0	0

Word-message matrix: Construct matrix containing word weight in each message whereas rows refer to words and columns to messages and weighting value in this matrix refer to word frequency in each message. Let, L be a matrix where element describes the frequency of word in message (for example, the frequency) (Table 1):

$$W_i^T = \begin{bmatrix} & Mj & \\ x_{1,1} & \cdots & x_{1,n} \\ \vdots & \ddots & \vdots \\ x_{m,1} & \cdots & x_{m,n} \end{bmatrix}$$

Each row in this matrix is a vector corresponding to a word giving its relationship to each message:

$$W_i^T = \left[x_{i,1}, \ldots, x_{i,n} \right]$$

$$W_i^T = 0.693147,\ 0.693147,\ 0.693147,\ 0.693147,\ 1.098612,\ 1.098612$$

While the columns in this matrix refer to a vector corresponding to a message giving its relationship to each word:

$$M_j = \begin{bmatrix} x_{1,j} \\ \vdots \\ x_{m,j} \end{bmatrix},$$

$$M_j = 1.098612,\ 0,\ 0.693147\ 0.693147,\ 0.693147,\ 0,\ 0,\ 0$$

Low-ranking based Singular Value Decomposition (SVD): After preprocessing messages and build frequency (weights) array, the first step is applying SVD which is the heart in latent semantic analysis. The singular value decomposition of an array X is the analysis of X for the simple three arrays $X = U\Sigma V^T$, so, the columns of V and U are orthonormal and the array Σ is diagonal with real positive inputs. The SVD is applied in several fields such as statistics and signal processing, general classification, clustering, collaborative filtering and so on. The data array X is near to an array of low rank and it is helpful to get a low rank array that is a perfect approximation to the data array. Also, singular value decomposition knows for all arrays (square and

rectangular). In contrast the most generally employed spectral decomposition in linear algebra. The columns of U are named the left singular vectors and also compose an orthogonal collection. A modest results of the orthogonality is that for a square and invertible array X, the columns of V in the singular value decomposition, named the right singular vectors of X, usually compose an orthogonal collection with no hypothesizes on X, we can be describes the main steps of SVD as:

- Find its transpose X^T to compute X^TX to compute U matrix which is representing words-topics and XX^T to compute V^T matrix which is representing messages-topics
- Determine the eigenvalues of X^TX and sort these in descending order, in the absolute sense. Square roots these to obtain the singular values of X
- Construct diagonal matrix Σ by placing singular values in descending order along its diagonal
- Use the ordered eigenvalues from step 2 and compute the eigenvectors of X^TX. Place these eigenvectors along the columns of V and compute its transpose, V^T

Using SVD to perform low-ranking in latent semantic analysis for two main reasons:

- The frequency matrix which represented word-message matrix is very large for computing resources and noisiness
- Also, this matrix is sparse that is word-message matrix only the words actually in each message whereas must be interested for all words related in each message that much larger set may be synonym (Markovsky, 2012). After applying SVD on words-messages matrix W_i^T will be obtain into three matrixes:

$$U = \begin{bmatrix} \begin{bmatrix} \\ u_1 \\ \\ \end{bmatrix} \cdots \begin{bmatrix} \\ u_1 \\ \\ \end{bmatrix} \end{bmatrix}, \Sigma = \begin{bmatrix} \sigma_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_1 \end{bmatrix}, V^T = \begin{bmatrix} [& v1 &] \\ & \vdots & \\ [& vl &] \end{bmatrix}$$

$$W_i^T = U \cdot \Sigma \cdot V^T$$

As shown in Table 2 which is explain part of V^T Matrix that result from applying LSA on messages-topics (frequency matrix).

Table 2: Messages-topics matrix

No.	Topic 1	Topic 2	Topic 3	Topic 4
Message 1	0.062046,1	-0.08711,2	0.016807,3	-0.014183,4
Message 2	0.117934,1	-0.087369,2	-0.022652,3	0.094949,4
Message 3	0.00812,1	-0.005211,2	-0.003991,3	-0.004517,4
Message 4	0.004795,1	-0.000594,2	-0.006292,3	0.00392,4
Message 5	0.059072,1	-0.00616,2	0.049599,3	-0.018109,4

These matrices will be the inputs to classification model in which U matrix will be used as the features and V matrix will contain the data and the Algrothim 1 explains feature extraction and feature selection algorithm.

Algorithm 1; Features extraction and features selection:

Input: Latent semantic analysis output
Output: Classification SMS message to junk or not junk
Begin
Phase 1:-preprocessing SMS messages
For each SMS message
Step 1:- tokenization to extract separately words(features)
Step 2:-stop words removal.
Step 3:-convert each feature to BOW based on log(1+ frequency W_iM_j)
End for
Phase 2:-Inputs words-message X matrix to latent semantic analysis to dimensional reduction by applying low ranking based on Singular Value Decomposition (SVD)
SVD:

1. Find its transpose X^T to compute X^TX to compute U matrix which is representing words-topics and XX^T to compute V^T matrix which is representing messages-topics
2. Determine the eigenvalues of X^TX and sort these in descending order, in the absolute sense. Square roots these to obtain the singular values of X
3. Construct diagonal matrix Σ by placing singular values in descending order along its diagonal
4. Use the ordered eigenvalues from step 2 and compute the eigenvectors of X^TX. Place these eigenvectors along the columns of V and compute its transpose, V^T

End

Quick rules based fuzzy rough set classification model: In this research, a fuzzy-rough set (Hu *et al*., 2012) for classification SMS junk was applied as shown in Algrothim 1. It has approved its effectiveness in many areas with the features selection and classification in terms of the following points:

- Flexibility in approximations
- Fuzzy rough set deal with discrete or real-valued noisy data (or a mixture of both)
- This technique can be applied to data with continuous or nominal decision attributes and as such can be applied to regression as well as classification datasets
- Dependency of words

Quick rules based fuzzy rough set has been suggested in this work. It relies on the principle of rules induction for features selection. Quick reduct technique is an efficient method for computing reduct. This is commonly applied in multiple soft computing applications using rough sets. Quick-reduct algorithm tries to compute

a reduct without exhaustively creating all subsets. It will take its inputs from algorithm 1 after preprocessing SMS messages and after that it begins with an empty collection and gathers in turn, one at a time, those features that are the biggest increase in the rough set dependency metric till providing the greatest possible value for the dataset.

Lower and upper approximation: Lower and upper approximations are employed in fuzzy rough set to identify to what extent the group of objects can be labeled into a given class weakly or strongly. Where the lower approximation is the complete collection of elements which can be positively (unambiguously) categorized as belong to one class and upper approximation is the collection of elements which cannot be positively (ambiguously) classified as belong to more than one class. As shown in following Eq. 1 and 2:

$$(R_{BA})(y) = INF_{x\in X}\, \beta\, ((R_B(x, y), A(x)) \quad (1)$$

$$(R_B^A)(y) = sup_{x\in X}\, £((R_B(x, y), A(x)) \quad (2)$$

where, R(x, y) is the similarity value for each topic in a message, we using A(x) is the class value for each message. £ is the T-norm equation that using to compute upper approximation and β is the implication equation that using to compute lower approximations.

Finding positive region: In fuzzy rough set the positive region is a fuzzy collection in X that includes every element y to the extent that all elements with approximately equal values for the features in B, have equal class values d. The positive region can be computing by summation the lower approximation as shown in the following Eq. 3:

$$POS_B(y) = (U_{x\in X}\, R_d^x) \quad (3)$$

After that using positive region to calculating dependency function for each feature as following in Eq. 4:

$$\text{Dependency function} = \frac{\text{POS B(y)}}{\text{No. of message}} \quad (4)$$

Quick rules step can be summarized in Algorithm 2.

Algorithm 2; Quick rules based on fuzzy rough sets:

Input: Latent semantic analysis outputs (SMS messages-topics)
Output: Classification SMS messages to junk or not junk
Begin
Step 1:-
C is conditional features
$Dependency_{best} = 0$, $Dependency_{prev} = 0$, R = {}
Do
$Dependency_{prev} = Dependency_{best}$
$Dependency_{high=0}$, $Dependency_{low=1}$
reducts={}
∀X∈C
1. Compute similarity matrix using the law
R(x, y) = abs(a(x)-a(y))/abs(a_max-a_min)
a_max is the highest weight for word W_i in message M_j
2. Compute lower approximation by equation
T(x, y) = min(x, y)
3. Compute upper approximation by equation
I(x, y) = max(1-x, y)
4. compute positive region by summation lower approximation
$POS_B(y) = \sum (RB\ X\ k)(y)$
5. calculate dependency value for each reduct
Dependency (x) = POS B (y)/no. of message
Reducts = reducts U(x, dependency(x))
If (dependency(x)>$Dependency_{high}$)
$Dependency_{high}$ = dependency(x)
Else
if (dependency(x)<$Dependency_{low}$)
$Dependency_{low}$ = dependency(x)
Reducts = (reduct, $Dependency_{high}$, $Dependency_{low}$)
R = R U select features(reducts)
$Dependency_{best}$ = dependency(x)
Until $Dependency_{best} = Dependency_{prev}$
Return R
End

After applying main steps of quick rules algorithm and obtaining the dependency value for each feature, set of rules for these features are generated. It starts with empty set of rules and each time additional rules are added for increasing the accuracy and accordingly would increase the classification performance. This process will be stopped in two cases:

- Reached to maximum evolution function (or to degree α)
- Or when rules which its accuracy will not optimize classifier precision as shown in Fig. 2

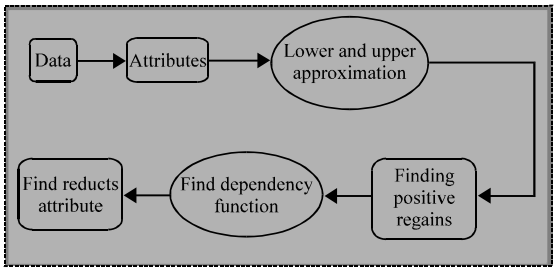

Fig. 2: The fuzzy rough set structure

RESULTS AND DISCUSSION

In our research using precision, F-measure, recall as measures to evaluate classifier performance (Uluyagmur *et al*., 2012):

- Precision is a ratio of the relevant suggested items to total number of items suggested
- Recall is a ratio of the relevant suggested items to total number of relevant items available
- F-measure is combining the precision and recall

The proposed model has been evaluated using the 421 messages from data set with 70% for training and 30% for testing. There are many algorithm have been applied on SMS messages classification such as decision tree and Naive Bays as shown in Table 3 with 70% for training and 30% for testing.

Quick rules has 0.921 precision. It has been confirmed that the quick rules method alone does not always create a minimal reduct as dependency function is not a good heuristic. It produces a converged minimal reduct which is still helpful in highly minimizing dataset dimensionality. An understanding of quick reduct means that, for dimensionality evaluations of the dependency function may be executed for the worst-case dataset. In the decision tree J48 algorithm the matrix of words-messages has represented as tree where each node acts as word. The tree depth is enlarged when the number of features increased and hence, excesses time complexity of model. in addition, the decision tree may add new features which do not improve classifier's performance, this phenomenon is usually called as overfitting. Naive Bays technique has many disadvantages when applying on SMS message such as very strong assumption where it depended on the data distribution, i.e., any two features are independent given the output class. It was observed that applying the latent semantic analysis as features extraction has effected positively in increasing the classification accuracy for the inputted SMS messages with the focus on semantic relation among words of messages. Figure 3 and 4 explain the latent semantic analysis effectiveness for increasing the quick rules accuracy. Figure 5 shows the quick rules performance on 421 SMS messages without LSA, Fig. 4 explains quick rules performance when have been employed LSA and Fig. 5 illustrated the performance of all methods that applied on 421 messages.

Table 3: Performance Naive Bayes, decision tree and quick rules without LSA and with LSA techniques

Algorithms	Precision	Recall	F-measure	Class
Quick rules	0.921	0.990	0.967	Not junk
	0.857	0.500	0.632	Junk
Decision tree	0.939	0.995	0.966	Not junk
	0.926	0.510	0.658	Junk
Naïve bayes	0.946	0.950	0.958	Not junk
	0.800	0.898	0.846	Junk
Quick rules with latent	0.951	1.000	0.978	Not junk
semantic analysis	1.000	0.615	0.762	Junk

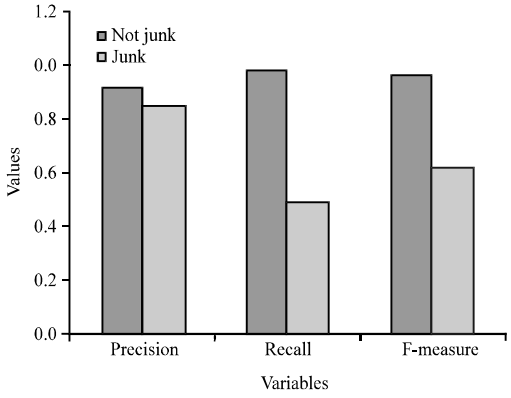

Fig. 3: The evaluation of fuzzy rough quick rules without LSA

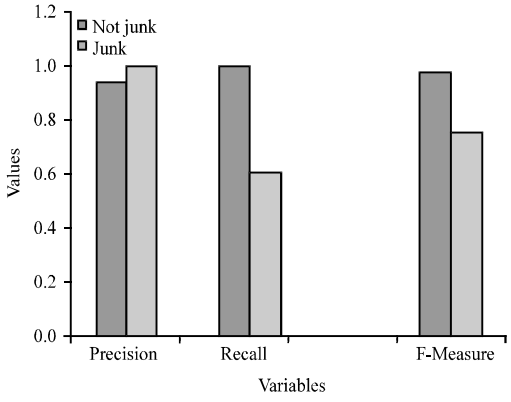

Fig. 4: Quick rules with LSA

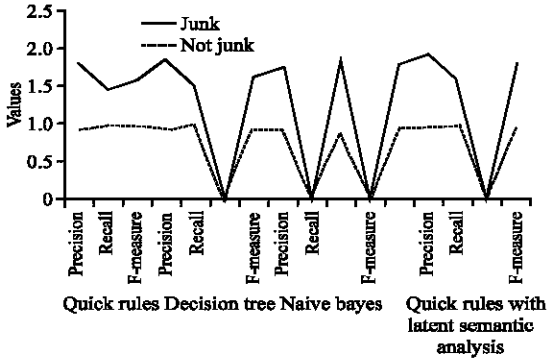

Fig. 5: Decision tree J48, Naive Bayes, quick rule without LSA, quick rule with LSA algorithms performance comparison

CONCLUSION

This model has provided a review for employing fuzzy-rough set approach in SMS junk filtering. The main obstacles were the nature and the length of the messages text, however, utilizing the latent semantic analysis has shown significant enhancement with the performance of the quick rules fuzzy rough set technique. The results were encouraging in comparison with decision tree J84, Naive Bays and random forest. Also, the fuzzy-rough set have proven high accuracy with the text messages classification.

REFERENCES

Agarwal, S., S. Kaur and S. Garhwal, 2015. SMS spam detection for Indian messages. Proceedings of the 1st International Conference on Next Generation Computing Technologies (NGCT), September 4-5, 2015, IEEE, Dehradun, India, ISBN:978-1-4673-6809-4, pp: 634-638.

Akbari, F. and H. Sajedi, 2015. SMS spam detection using selected text features and boosting classifiers. Proceedings of the 2015 7th International Conference on Information and Knowledge Technology (IKT), May 26-28, 2015, IEEE, Urmia, Iran, ISBN:978-1-4673-7485-9, pp: 1-5.

Al-Hasan, A.A. and E.S.M. El-Alfy, 2015. Dendritic cell algorithm for mobile phone spam filtering. Procedia Comput. Sci., 52: 244-251.

Almeida, T., J.M.G. Hidalgo and T.P. Silva, 2013. Towards SMS spam filtering: Results under a new dataset. Intl. J. Inf. Secur. Sci., 2: 1-18.

Almeida, T.A., J.M.G. Hidalgo and A. Yamakami, 2011. Contributions to the study of SMS spam filtering: New collection and results. Proceedings of the 11th ACM Symposium on Document Engineering, September 19-22, 2011, ACM, Mountain View, California, USA., ISBN:978-1-4503-0863-2, pp: 259-262.

Biggio, B., G. Fumera, I. Pillai and F. Roli, 2011. A survey and experimental evaluation of image spam filtering techniques. Pattern Recognit. Lett., 32: 1436-1446.

Bodic, L.G., 2005. Mobile Messaging Technologies and Services: SMS, EMS and MMS. John Wiley & Sons, Hoboken, New Jersey, Pages: 431.

Delany, S.J., M. Buckley and D. Greene, 2012. SMS spam filtering: Methods and data. Expert Syst. Appl., 39: 9899-9908.

Galvez, R.H. and A. Gravano, 2017. Assessing the usefulness of online message board mining in automatic stock prediction systems. J. Comput. Sci., 19: 43-56.

Hu, Q., L. Zhang, S. An, D. Zhang and D. Yu, 2012. On robust fuzzy rough set models. Fuzzy Syst. IEEE. Trans., 20: 636-651.

L'Huillier, G., A. Hevia, R. Weber and S. Rios, 2010. Latent semantic analysis and keyword extraction for phishing classification. Proceedings of the 2010 IEEE International Conference on Intelligence and Security Informatics (ISI), May 23-26, 2010, IEEE, Vancouver, British Columbia, Canada, ISBN:978-1-4244-6444-9, pp: 129-131.

Landauer, T.K., P.W. Foltz and D. Laham, 1998. An introduction to latent semantic analysis. Discourse Process., 25: 259-284.

Landuaer, T.K., D.S. McNamara, S. Dennis and W. Kintsch, 2011. Handbook of Latent Semantic Analysis. Routledge, Abingdon, UK., ISBN: 978-0-8058-5418-3, Pages: 531.

Markovsky, I., 2012. Low-Rank Approximation: Algorithms, Implementation, Applications. Springer, Berlin, Germany, ISBN 978-1-4471-2226-5, Pages: 253.

Uluyagmur, M., Z. Cataltepe and E. Tayfur, 2012. Content-based movie recommendation using different feature sets. Proceedings of the World Congress on Engineering and Computer Science Vol. 1, October 24-26, 2012, WCECS, San Francisco, USA., ISBN:978-988-19251-6-9, pp: 17-24.

Uysal, A.K., S. Gunal, S. Ergin and E.S. Gunal, 2012. A novel framework for SMS spam filtering. Proceedings of the 2012 International Symposium on Innovations in Intelligent Systems and Applications (INISTA), July 2-4, 2012, IEEE, Trabzon, Turkey, ISBN:978-1-4673-1446-6, pp: 1-4.

Valles, E. and P. Rosso, 2011. Detection of near-duplicate user generated contents: The SMS spam collection. Proceedings of the 3rd International Workshop on Search and Mining User-Generated Contents, October 24-28, 2011, Glasgow, Scotland, UK, pp: 27-34.

Wang, C., Y. Zhang, X. Chen, Z. Liu and L. Shi *et al*., 2010. A behavior-based SMS antispam system. IBM. J. Res. Dev., 54: 3-16.

Wen-Liang, H., L. Yong, Z. Zhi-Qiang and S. Zhong-Ming, 2009. Complex network based SMS filtering algorithm. ACTA. Autom. Sin., 35: 990-996.

Zhang, H.Y. and W. Wang, 2009. Application of Bayesian method to spam SMS filtering. Proceedings of the 2009 International Conference on Information Engineering and Computer Science ICIECS, December 19-20, 2009, IEEE, Wuhan, China, ISBN:978-1-4244-4994-1, pp: 1-3.

Zhu, Y. and Y. Tan, 2011. A local-concentration-based feature extraction approach for spam filtering. IEEE. Trans. Inf. Forensics Secur., 6: 486-497.