Iran Journal of Computer Science (2018) 1:99–107
https://doi.org/10.1007/s42044-018-0007-4

ORIGINAL ARTICLE

An evaluation of Reber stemmer with longest match stemmer technique in Kurdish Sorani text classification

Ari M. Saeed[1] · Tarik A. Rashid[2,4] · Arazo M. Mustafa[3] · Rawan A. Al-Rashid Agha[2] · Ahmed S. Shamsaldin[2] · Nawzad K. Al-Salihi[2]

Received: 16 November 2017 / Accepted: 12 January 2018 / Published online: 26 January 2018

Abstract
Stemming is one of the most significant preprocessing. stages in text categorization that most of the academic investigators aim to improve and optimize the accuracy of the classification task. High dimensionality of feature space is one of the challenges in text classification that can be decreased by many techniques. In stemming, high dimensionality of feature space is decreased by grouping those words that they have same grammatical forms and then getting their root. This work is dedicated to build an approach for Kurdish language classification using Reber Stemmer. Thus, an innovative approach is investigated to get the stem of words in Kurdish language by removing longest suffix and prefixes of words. This approach has a strong capability and meets the requirements in responding to the process of deleting as many of the required affixes as possible to get the stem of words in Kurdish language. The advantage of this stemmer is that it ignores the ordering list of affixes that receives correct stem for more than one words that have the same format. The stemming technique is implemented on KDC-4007 dataset that consists of eight classes. Support Vector Machine (SVM) and Decision Tree (DT or C 4.5) are used for the classification. This stemmer has been successfully compared with the Longest-Match stemmer technique. According to results, the *F*-measure of Reber stemmer and Longest-Match method in SVM is higher than DT. Reber stemmer in SVM for classes (religion, sport, health and education) obtained higher *F*-measure, while the rest of classes are lower in Longest-Match. Reber stemmer in DT for classes (religion, sport and art) had higher *F*-measure for Reber stemmer while in Longest match the rest of classes showed lower *F*-measure.

Keywords Kurdish text classification · Stemming · Support vector machine · Decision tree

1 Introduction

Text classification is a substantial task to retrieve valuable information from massive amount of data. The importance of this domain has appeared and become a big challenge when the Internet has been distributed rapidly in the world. Documents on the Internet that are classified depend on the contents of a text which can be grouped for one or more predefined labels. The idea of text classification is utilized in many various applications such as recommender systems, language guessing and spam filtering [1,2]. In text classification, high dimensionality of features space decreases the efficiency of classifier. Hence, searching and indexing of documents need more time as the recall gets decreased without compromising of precession. Nevertheless, a new challenge is raised to progress the effectiveness of classifier by optimizing and inventing new techniques. Stemming is one of the techniques in data-preprocessing that researchers encounter which aids to increase in the accuracy of classification. In other words, the prime factor of using stemmers is reducing of words by removing the affixes of words to get the stem (root). Furthermore, it can be said that stemmers are used to decrease high dimensionality of features and they have a profound impact to minimize time taken to construct models. This issue has been addressed and numerous stemmer's

✉ Tarik A. Rashid
tarik.ahmed@ukh.edu.krd

[1] Department of Computer Science, College of Science, University of Halabja, Halabja, Iraq

[2] Department of Computer Science and Engineering, University of Kurdistan Hewler, Erbil, Iraq

[3] School of Computer Science, College of Science, University of Sulaymaniyah, Sulaymaniyah, Iraq

[4] Software and Informatics Engineering, Salahaddin University-Erbil, Erbil, Iraq

algorithms implemented on various languages. Hence, it is pointless to say that languages are different according to their morphology and richness, for example: Kurdish language is one of the Northwestern Iranian languages that spoken by Kurdish people in Western Asia and it is complex and rich in dialects as well. There are two main dialects in Kurdish language: Kurmanji and Sorani. In this study, a new method is proposed and evaluated to remove the affixes of Sorani text [3,4].

Generally, in Kurdish language text classification is divided for three various tasks: preprocessing, classification and evaluation. Preprocessing involves some stages to clean the data and remove useless information from the dataset that might increase the performance of classification. In this paper, there is an attempt to evaluate the Reber stemmer with Longest-Match stemmer in Kurdish text and compare ****F-measure of each classifier [5].

The other sections of this paper are prepared as follows: Sect. 2 is primarily about reviewing the previous works in document classification. Kurdish Language morphology is presented in Sect. 3. In Sect. 4 root/stemming technique approach is presented. Classification technique is explained in Sect. 5. Implementation and experiments are detailed in Sect. 6. Finally, the key points are concluded.

2 Related work

In [4], the researchers evaluated four distinct types of affixing removal stemming algorithm for each Porter, Lovins, S-Removal and Paice by using Hamming distance measure. The similarity and strength of each algorithm are depicted after implementing a list of 49,659 English frequently-used words that derived from Momby corpus and UNIX spelling dictionary. The number of measurement is six and the strongest stemmer is the one that has a highest value for each one. According to the experiments the strongest one is Paice, Lovins, Porter and S-Removal; accordingly, Paice had the highest value in recall and index compression while its precision was the lowest.

In another approach [6], four different accuracy measurements are utilized to compare three different algorithms (Shereen Khoja Stemmer, Tim Buckwalter Morphological analyzer and Tri-literal Root Extraction Algorithm) with gold standard. The methods of each stemmer to remove affixes are different, for example, Khoja extracted the word to get the stem by removing the longest affixes whilst Buckwalter used all prefixes to compile only one lexicon and Tri-literal used weighting of word depending on their position. The accuracy rate was a crucial factor in natural language processing. In the best state, the registered accuracy algorithm was 75%. According to the results, the accuracy of the Khoja stemmer had the highest ranked place, then tri-literal algorithm and then followed by Buckwalter morphological analyzer [6].

Salavati used rule-based stemmer to build Jedar as a first stemmer for Sorani and Kurmanji dialects. Gras algorithm was implemented as statistical technique to remove nested suffix. The dataset in this experiment was Pewans that consists of 25K Kurmanji and 115K Sorani documents. According to results, the performance of Information Retrieval (IR) was increased by utilizing rule-based stemmer if the length of stem was 3 for each Sorani and Kurmanji. Additionally, the performance of Jedar stemmer was enhanced dramatically by 35% for improving the IR [3].

Moghadam and Keyvanpour [7] compared in their study different types of Farsi/Persian stemmers to evaluate the strongest and weakest one. In Farsi/Persian language stemmers are grouped for three commonly types: statistical stemmer, lookup table stemmer and structural stemmers. The performance in this experiment was evaluated by using precision and recall. The effectiveness of each stemmer depends on the size of dataset by the form which for some stemmers the precision was high and the recall was low.

In another work [8], the use of the SVM for text categorization was explained. It revised the individual properties of leanring and why SVM was appropriate for such a work. The theoretical analysis concluded that SVMs recognized specific text properties that were (1) high dimensional feature spaces, (2) few irrelevant features and (3) sparse instance vectors. The experiments consisted of comparing the SVMs performance via using polynomial and RBF Kernels with four standard learning methods usually utilized for the purpose of text classification. The experimental results supported the theoretical findings indicating that the SVMs prospered in improvement over the currently work performing approaches.

Zeng et al. [9] explained the limits of key factors for SVMs in huge sample tasks. The sequential minimal optimization (SMO) is regarded as the key method to resolve the SVMs which can reduce the difficulty degree through decompositions strategies. In this work, a procedure regarding the parallel computing concept based on symmetric multiprocessor (SMP) machine was improved. This new method had great benefit in terms of speediness when applied to problems with large training sets and high dimensional spaces without reducing generalization performance of SVMs.

A comparison between methods of multiclass support vector machines was explained [10]. They suggested decompostion implementations for two methods and after that they compared their performance based on binary classifications "one against all" and "one against one". Their results indicated that "one against one" and directed acyclic graph methods were more suitable for practical use.

A stemmer technique was used for Farsi/Persian language [11], like most languages, the stemmer was based on mor-

phology and the implementation was done for suffix stemmer and prefix stemmer. The first step in this work was to find the terminal substring of the word which was already in the list of Farsi/Persian morphological prefix, and then, removing the suffix and in cases of multiple suffixes were founded the designed stemmer would select the longest suffix that would leave the word with less characters. In other words, the suffix alone could be one character or more and when it was added to the other suffix. It would make it as one long suffix and this one would be removed. The algorithm developed to consider some exceptions in the database like the words which were structurally similar to other words. Also the algorithm could find the verbal suffixes but it would not remove them, and the suffix "stan" would not be removed as it is usually used for countries and regions. As well as the algorithm was limited and the stem should consist of three letters or more after removing the suffixes and prefixes. In case if it was less than three letters, then the algorithm would remove the suffix considering that the result would not be less than three and a part of the suffix would remain with the stem.

Furthermore, they used BNF machine to implement this algorithm [11]. The prefix would be removed in two states which were to detect and remove, while the suffix would be removed in 15 states. The suffix stemmer reversed the word before removing any prefix or suffix in each step. As well as checking the type of the word and also checking the prefixes and suffixes which were removed in prior steps. The algorithm mainly can be divided into two parts:

1. First step: this step both suffix or prefix would be detected and removed from the word. While the results would be given back to the suffix stemmer or prefix stemmer as a new word.
2. Last step: the above process was repeated till it could not find any suffix or prefix or the word contained less than three letters. In this situation, the word was returned without any removal.

3 Kurdish language morphology

The Kurdish language is an Indo-European language that is spoken by 35 million people [12]. The homeland of this language is Kurdistan that is located among Iraq, Iran, Turkey and Syria. The Kurdish language has different dialects and the populations of each dialect various, for example, Central Kurdish (Sorani) and Gorani dialects are spoken by Kurds of Iran and Iraq. While Northern Kurdish (Kurmanji) dialect is spoken in Iraq, Iran, Turkey, Syria, Armenia, and Lebanon. Zazaki dialect is spoken in Turkey. The diversity of dialects and its spread on different countries have become diversity of scripting. Hence, it can be said that there is not any consensus on dialects and scripts of the Kurdish language. Sorani and Kurmanji populations are 75%. Therefore, this article is focused on Sorani dialects with a small overview on Kurmanji and the below points are the characteristics of this dialect [13]:

Table 1 shows Kurdish character changing position

Characters	Example	Meaning
In the end (ی)	ئازادی	Freedom
Between (ـیـ)	سەیران	Picnic
In the beginning (یـ)	یاری	Play
In the end (ە)	یەکێکە	Someone
Between (ـھـ)	میهرەبان	Magnificent
In the beginning (ھـ)	هەمیشە	Forever
In the end (ك)	کاغەزێك	A paper
Between (ک)	چونکە	Because
In the beginning (ک)	کردووە	Did
In the end (ێ)	دڵەڕاوکێ	Hesitation
Between (ـێـ)	بەهێز	Strong

1. Due to some geopolitical and geographical situation four different scripts are used (Yekgirtû (unified), Cyrillic, Persian/Arabic and Latin); for example, Sorani and Gorani (Hawrami) use Persian/Arabic scripts because of the population of Kurds are in Iraq and Iran while mostly of Kurmanji and Zazaki use Latin scripts since most of Kurmanji are in Turkey and some of them live in Iraq, Syria, Soviet and Armenia use Cyrillic scripts.
2. In Sorani, the character of 'غ' and 'خ' are different while in Latin only 'x' is used according to Bedrxan and Lescot. Moreover, in Sorani and Yekgirtû (unified) two characters are used instead of one, for example: 'وو' in Persian/Arabic is used instead of 'û' in Latin and "sh" in Yekgirtû (unified) is used instead of 'ş' in Latin.
3. Shape of same character is changed in Sorani according to their position in words for example:
   (a) Sorani and Gorani (Hawrami) script are written from right to left while Kurmanji and Zazaki are written from left to right.
   (b) No capital letters in Sorani.
   (c) The position of negations is different, for example, in Sorani it is said: "le nêzîk **niye**" in Kurmanji: "**ne** li nêzîkê" (the negations are shown in bold).
   (d) No difference in gender in Sorani while in Kurmanji there is difference [14] (Table 1).

4 Root/stemming techniques

Stemming is the process of acquiring the root of words which depends on some linguistic rules. In text classification, mor-

Table 2 List of prefixes and suffixes

Prefixes	Suffixes
"دەر","هەڵ","ڕا","بە","لە","جێ","پێ","گۆ","دە", "ب","م","مان","تان","ێ","یان","سەر",	"ایە","دا","ش","ێ" , "گا","یان","تان", "مان","مکان","و" , "یەک" , "مکە" ,"مک","بە","وە", "ەوە" , "تن" , "ین" , "ان" ,"دن" , "ە" ,"ن","م" ,"ێت", , "تر","بوون" , "کار", "هات"

Table 3 Example of removing prefixes

String	Prefixes	First Iteration	Prefixes	Second Iteration	Prefixes	Third Iteration	Shortest String
بەگۆیدەنگەکانیان	ب	مگۆیدەنگەکانیان		مگۆیدەنگەکانیان		مگۆیدەنگەکانیان	دەنگەکانیان
	بە	گۆیدەنگەکانیان	گۆ	یدەنگەکانیان	ی	دەنگەکانیان	

phological variation of a word is one of the few challenges. The main aim of stemming is reducing the number of words and grouping them under one stem. Furthermore, the advantage of stemmer is less time to process and less memory space is needed for storage. There have been many algorithms and techniques that have been implemented on different languages that depend on two major norms which are Iteration and Longest-Match:

(a) Iteration stemmer is used to remove the suffixes of words when it is matched with suffix that is already ordered. In this method only one suffix is removed that starts from the end of the word to the beginning with taking into consideration of the ordering list of suffixes. For example: the word "خواردنەکەمان" ends with the nested suffixes of ("مان", "مکە" and "ن") but according to this algorithm the root of "خواردنەکەمان" may be one of ("خواردنەکەما", "خواردنەکەم" or "خواردنەکە") by removing one of these suffixes ('ن', "ان" or "مان") that depends on the order of suffixes. When the first suffix is matched that one is removed and so on.

(b) Longest-match: is used to remove the suffixes of word by selecting the longest suffixes which is matched with a list of suffixes with taking into account the list of suffixes order. The main idea of this method is removing longest suffix and getting the smallest stem (root). For example; the word "خواردنەکەمان" ends with the suffixes of ("ن", "ان" and "مان"). According to this algorithm the root of "خواردنەکەمان" is "خواردنەکە" if the order of suffix "مان" is in the beginning or else the root may be one of the ("خواردنەکەما", "خواردنەکەم" or "خواردنەکە") based on the suffix order. However, the word of "خواردنەکە" is not stem since Kurdish Language is followed by more than one suffix and starts by more than one prefixes. As a result, in this method the order of suffixes gives the right stem only for one suffix that is not suitable for Kurdish language [6,13].

In this paper, a new method is proposed called "Reber" to remove the longest suffixes and prefixes for more than one affixes without considering the order of affixes. Light is shed on the complexity of inflection and derivation in Kurdish language which investigated in this experiment. Table 2 illustrates the suffixes and prefixes that are used in this work.

One of the main points that distinguishes the Kurdish language with other languages is starting the word with more than one prefix and followed by more than one suffix. The following flowchart consists of two steps to remove affixes by using a program in Java programming language:

(1) The first step is used to remove three prefixes of a word. This is done by using one "for loop" that iterates three times and each iteration is used to remove one prefix. Then three array lists are used for this purpose as showed in Flowchart 1, after that the results (strings) are compared in the array list to select the shortest one as showed in the example in Table 3.

As shown in Table 3: in the beginning of the word "بەگۆیدەنگەکانیان" there are two choices of prefixes ('ب' and "بە") both of them are removed then "مگۆیدەنگەکانیان" and "گۆیدەنگەکانیان" are made accordingly after that, there is no prefixes with "مگۆیدەنگەکانیان" since there is no change. While "گۆیدەنگەکانیان" starts with "گۆ" that removed and "یدەنگەکانیان" is remaining then 'ی' is removed "دەنگەکانیان" is

Table 4 Explanation of removing suffixes from a Kurdish word

String	Suffixes	First Iteration	Suffixes	Second Iteration	Suffixes	Third Iteration	Suffixes	Fourth Iteration	Stem = Shortest String
دەنگەکانیان	ن	دەنگەکانیا		دەنگەکانیا		دەنگەکانیا		دەنگەکانیا	دەنگ
					ن	دەنگەکا		دەنگەکا	
	ان	دەنگەکانی	ی	دەنگەکان	ان	دەنگەک	مک	دەنگ	
					مکان	دەنگ		دەنگ	
	یان	دەنگەکان	ن	دەنگەکا		دەنگەکا		دەنگەکا	
			ان	دەنگەک	مک	دەنگ		دەنگ	
			مکان	دەنگ		دەنگ		دەنگ	

remaining then comparing "دەنگەکانیان" with "دەنگەکانیـان" and shortest string is selected that is "دەنگەکانیان".

2. The second step is used to remove suffixes by getting the string from the first step: four suffixes are removed by using one "for loop" that consists of four iterations and each iteration is used to remove one suffix as shown in Table 4 and Fig 1:

In Table 4, the word "دەنگەکانیـان" ends with 'ن', "ان" and "یـان" all of them are removed. The results are "دەنگەکانیـا", "دەنگەکـانی" and "دەنگەکـان" in first iteration. In second iteration "دەنگەکانیـا" is not changed since it doesn't end with suffix, "دەنگەکـانی" is converted to "دەنگەکـان" after removing 'ی' then the "دەنگەکـان" is converted to "دەنگەکـا", "دەنـگەک" and "دەنـگ" then the word "دەنگەکـان" ends with 'ن', "ان" and "مکـان", respectively. The results are "دەنگەکـا", "دەنـگەک" and "دەنـگ". The words "دەنگەکانیـا", "دەنگەکـا" and "دەنـگ" are not changed since no suffixes are ended. The word "دەنـگەک" is changed to "دەنـگ" after removing "مک". In last iteration, all strings are compared and then the smallest one is selected that is "دەنگ".

5 Classification techniques

In text classification, to evaluate a new technique, different algorithms need to be used to measure the efficiency and accuracy of each method. There are number of algorithms that are implemented in text classification such as SVM, Naive Bayes (NB) and DT. These three are among the top 10 data mining algorithms that were identified by [15]. A brief explanation on SVM and DT is written below:

(a) SVM: is one of the strongest supervised classification algorithms in machine learning. Linear and non-linear are two types of SVM. Hyperplane is a border that can be used to separate the positive and negative classes in linear type, while in the non-linear there is not any straight line, but curved line is utilized for separating classes. The distance between closest vectors to hyperplane are called Margin. One of the purposes of using SVM is maximizing of Margin. One against one and one against all are two different techniques that is used for classifying multiclass classification in SVM [15].
(b) DT: is a simple and hierarchical model algorithm that uses binary classification tree to split the vector space for two different paths. Each path is assigned to specific class and the set of variables are input and the output is discarded till it reaches the correct outcome of the class. DT is applied in many areas of classification and prediction. One of the advantages of DT is making complicated model based on a complex relationship between set of inputs while translating model is easy. In addition, it is difficult to work with noisy data and multiple variables are not allowed [14,16].

6 Implementation and experiments

In this research work, the proposed method is valued by using a set of Kurdish text documents that was collected from various websites. The number of documents are 4007 that

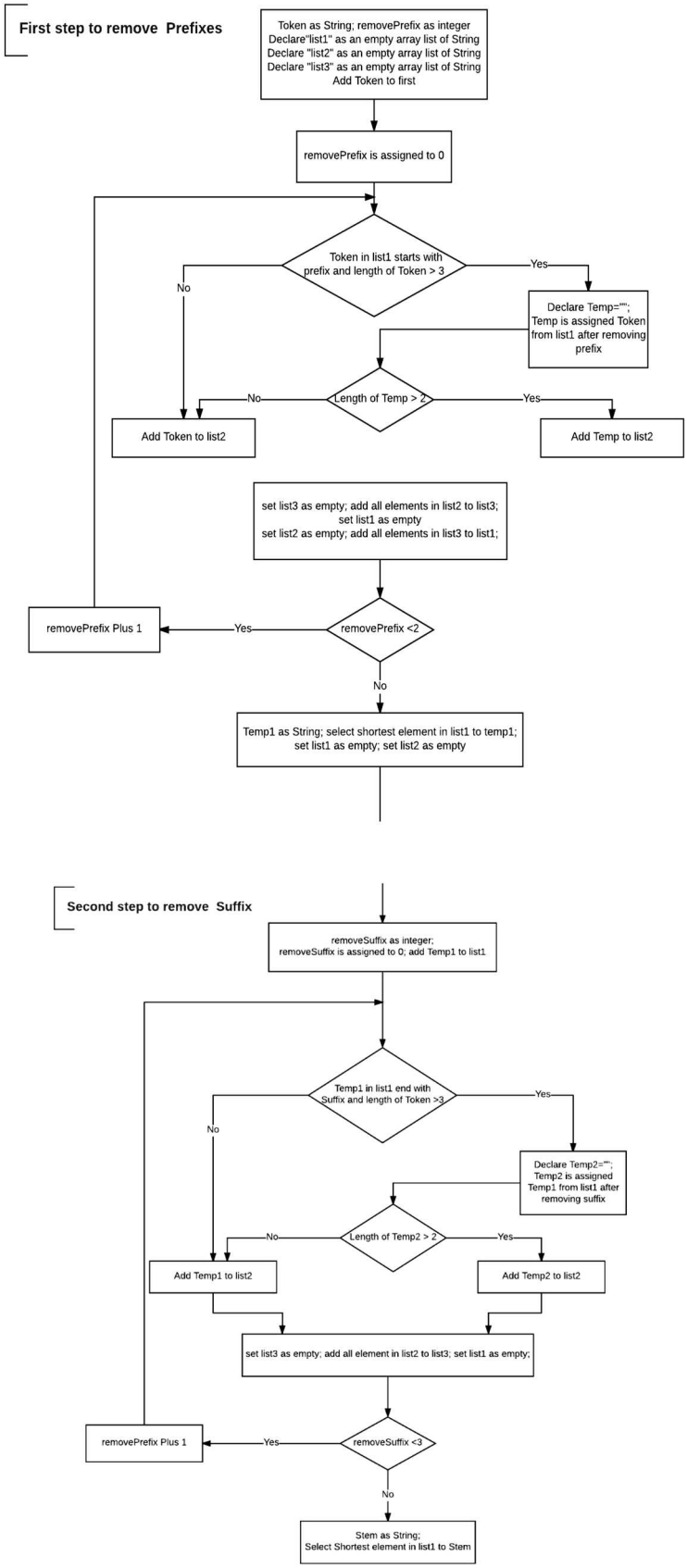


Fig. 1 The steps of proposed stemmer "Reber"

Table 5 Confusion matrix for measuring performance in text categorization

Prediction	Actual	
	Positive	Negative
Positive	TP	FP
Negative	FN	TN

partitioned for training and testing. Two imbalance data sets are made (70% as training and 30% as testing). The technique of tenfolds cross validation is used for training and testing. According to this technique, ninefolds are used as training subsets and onefold as a testing subset. The experiments are repeated ten times. Each time a different subset of testing is entered. The testing is implemented after a new model is made by training. The result is assessed by getting the average for each class (Economy, Health, Education, Sport, Style, Health, Religion, Socials). The actual results are presented by using confusion matrix for correctly and incorrectly predicted samples as shown in Table 5.

The above table shows that there are four possible outcomes while testing an instance: TP, FP, FN and TN. These are true positive, false positive, false negative and true negative, respectively. Furthermore, the positive means that the classification of the document is belonging to the category while the negative means it was not belonging. On the other hand, true means that the classification was correct with false being not correct. The performance is valued by using the formulas of precision, recall and *F*-measure. The above confusion matrix can be obtained from the below formulas [5,16–19]:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{1}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{2}$$

$$\text{F-measure} = \frac{2 \times \text{precision} \times \text{recall}}{\text{Precision} + \text{recall}} \tag{3}$$

The performance of proposed technique is weighed by using our KDC-4007 dataset which can be found in the following link: https://archive.ics.uci.edu/ml/datasets/KDC-4007+dataset+Collection and the results are compared after implementing SVM and C4.5. According to the results, using Eq. (3) the *F*-measure value shows to be different for each class. Hence, the values of the four classes (Religion, Art, Social and Style) are increasing while for the rest are decreased as shown in Table 6.

As shown in Tables 6 and 7 the *F*-measure for classes of Religion, Art, Social and Style is increased gradually while the values of Sport, Education, Health and Economy are decreased slowly in SVM. In DT the values are different

Table 6 Experiments results of *F*-measure on SVM

Classes	*F*-measure of natural sentences	*F*-measure of longest-match method	*F*-measure of Reber method
Religion	0.84	0.866	0.868
Sport	0.922	0.943	0.942
Health	0.877	0.903	0.9
Education	0.897	0.928	0.921
Art	0.905	0.944	0.949
Social	0.9	0.918	0.926
Style	0.901	0.93	0.933
Economy	0.947	0.966	0.964

Table 7 Experiments results of *F*-measure on C4.5

Classes	*F*-measure of natural sentences	*F*-measure of longest-match method	*F*-measure of Reber method
Religion	0.61	0.706	0.707
Sport	0.666	0.834	0.824
Health	0.573	0.76	0.751
Education	0.688	0.758	0.759
Art	0.675	0.795	0.773
Social	0.7	0.78	0.775
Style	0.681	0.815	0.824
Economy	0.759	0.869	0.87

using Eq. (3); the *F*-measure of Religion, Education, Style and Economy slightly raised and the classes of Sport, Art, Health, Economy and Social dropped steadily. It is clear that the success of this technique depends on the classes of the dataset.

Both results of the *F*-measure on SVM and C4.5 are also reflected onto a chart comparing between the Reber method and traditional methods (See Figs. 2, 3).

Another crucial point in implementing the Reber stemmer is decreasing the number of feature space that is obtained in this technique as exhibited by Fig. 4. In Fig. 4 there are three experiments that show the number of features after removing punctuation and none Kurdish letter. The number of features in data set before stemming (natural sentences) is 24,977 features but these numbers of features is decreased to 14,230 features after implementing Longest-match, while in this proposed method the number of features is 13,958. As a result, the difference between Longest-match method and proposed method is 815 features. Hence, it can be said that the proposed method is better than Longest-match method for Kurdish text classification. Additionally, another advantage is that the proposed stemmer needs less time to build model than Longest-match method stemmer. As shown in

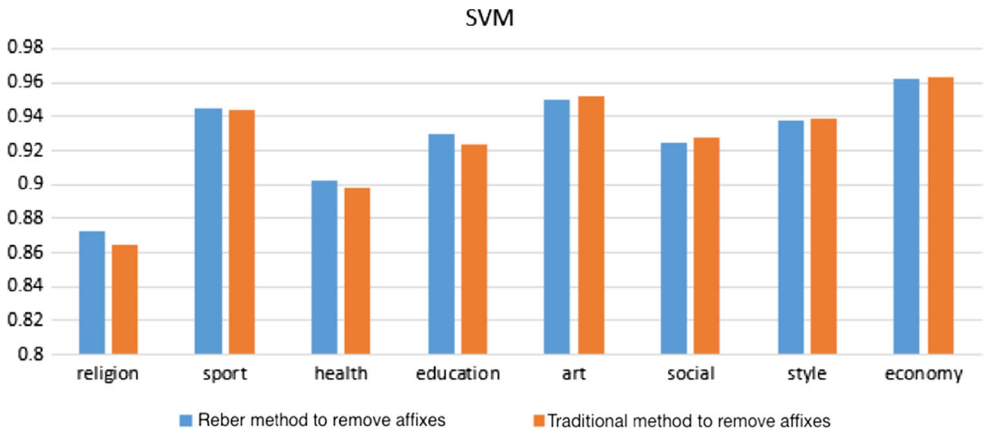


Fig. 2 Experiments Results of *F*-measure on SVM

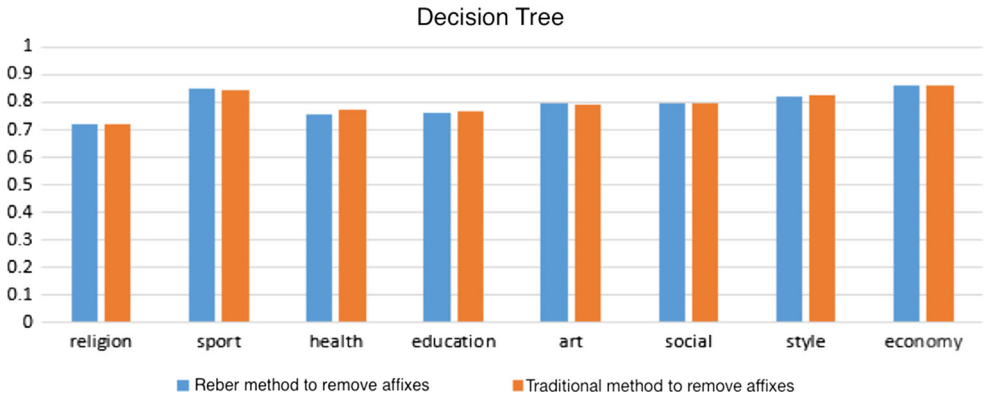


Fig. 3 Experiments results of *F*-measure on DT

Fig. 4 Number of features in dataset

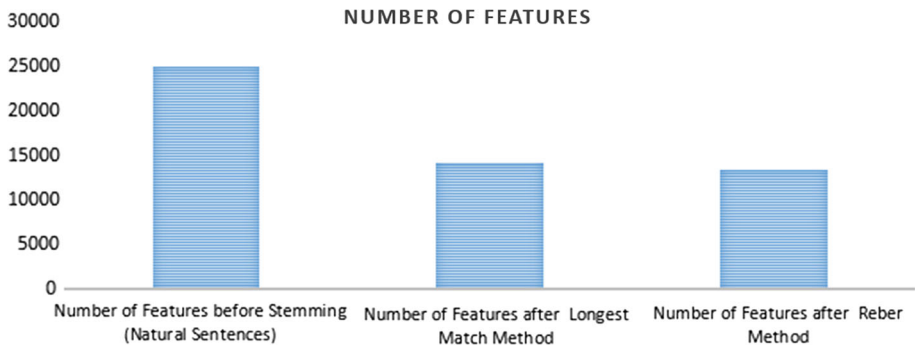
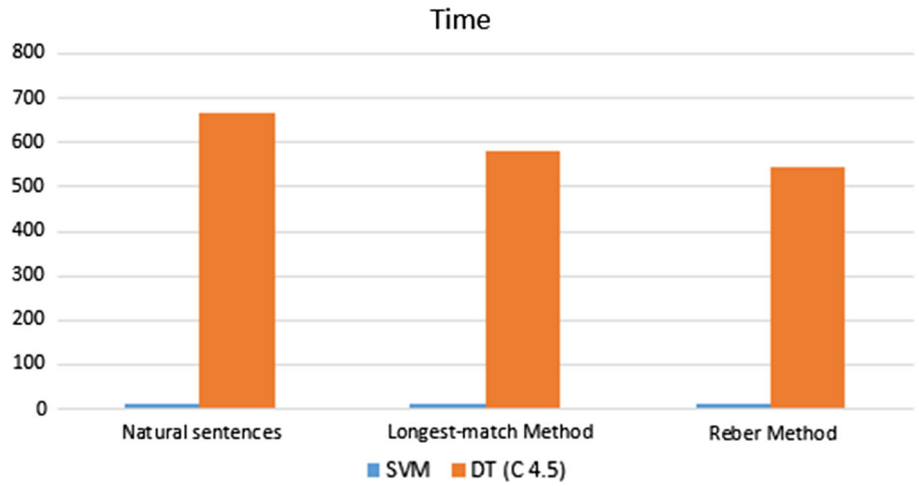


Fig. 5 the time to build model in natural sentences is 12.58 s but for Longest-Match it needs less time which is 10.11 s and in the proposed method is 9.97 s for SVM. In C4.5 the time taken to build model is 668.35 s in natural sentences while in Longest-match it decreased to 580.77 s and in the Reber method is 543.39 s. As a result, the proposed method is successful and requires less time in SVM and C 4.5.

7 Conclusion

The main aim of this paper was to investigate and evaluate the Reber stemmer for enhancing and increasing the accuracy of Kurdish text classification. According to the experiments, the number of features in data set before stemming was 24,977. This number was decreased to 14,230 after using Longest-match, though in the proposed method the number of features was 13,958. So, the difference was 815 features. Therefore, it

Fig. 5 Time to build model



is concluded that the proposed method is better than Longest-match method for Kurdish text classification. In addition, the proposed stemmer required lesser time than Longest-match method stemmer for constructing the model.

Acknowledgements The study is funded by the University of Kurdistan Hewlêr (UKH). The authors would like to thank the UKH for providing facilities and equipment for this research work.

References

1. Rashid, T.A., Mustafa, A.M., Saeed, A.: A robust categorization system for Kurdish Sorani text documents. Inf. Technol. J. **16**(1), 27–34 (2017)
2. Sawalha, M., Atwell, E.: Comparative evaluation of Arabic language morphological analysers and stemmers. In: 22nd International Conference on Computational Linguistics (2008)
3. Salavati, S., Sheykh Esmaili, K., Akhlaghian, F.: Stemming for Kurdish Information retrieval. Inf. Retr. Technol. **8281**, 272–283 (2013)
4. Shah, F., Patel, V.: A review on feature selection and feature extraction for text classification. In: IEEE Conference Publication. http://Ieeexplore.ieee.org; http://ieeexplore.ieee.org/abstract/document/7566545/. Accessed 08 Oct 2017 (2016)
5. Alahmadi, A., Joorabchi, A., Mahdi, A.: A new text representation scheme combining bag-of-words and bag-of-concepts approaches for automatic text classification. In: IEEE Conference Publication. http://Ieeexplore.ieee.org; http://ieeexplore.ieee.org/abstract/document/6705759/. Accessed 06 Oct 2017 (2013)
6. Yuan, P., Chen, Y., Jin, H., Huang, L.: MSVM-*k*NN: Combining SVM and *k*-NN for Multi-Class Text Classification. http://www.citeulike.org/user/alad/article/5393470. Accessed: 04 Oct 2017 (2008)
7. Moghadam, F., Keyvanpour, M.: Comparative study of various Persian Stemmers in the field of information retrieval. J. Inf. Process. Syst. **11**(3), 450–464 (2015)
8. Joachims, T.: Text categorization with support vector machines: learning with many relevant features. In: Nédellec C., Rouveirol C. (eds.) Machine learning: ECML-98. ECML 1998. Lecture Notes in Computer Science (Lecture Notes in Artificial Intelligence), vol. 1398. Springer, Berlin, Heidelberg (1998)
9. Zeng, Z., Yu, H., Xu, H., Xie, Y., Gao, J.: Fast training support vector machines using parallel sequential minimal optimization. In: IEEE Conference Publication. http://Ieeexplore.ieee.org; http://ieeexplore.ieee.org/abstract/document/4731075/?reload=true. Accessed 05 Nov 2017 (2008)
10. Hsu, C., Lin, C.: A Comparison of Methods for Multiclass Support Vector Machines (2002). http://ieeexplore.ieee.org/abstract/document/991427/. Accessed: 06 Nov 2017
11. Estahbanati, S., Javidan, R., Dezfooli, M.: A new method for stemming in Persian language considering exceptions. In: 5th SASTech. Khavaran Higher-Education Institute, Mashhad (2017) **(paper reference number: 15)**
12. Gunter, M.: The Kurds Ascending. Palgrave MacMillan, New York (ISBN 978-0-230-60370-7) (2008)
13. Hassani, H., Medjedovic, D.: Automatic Kurdish Dialects Identification. http://Airccj.org; http://airccj.org/CSCP/vol6/csit65007.pdf. Accessed: 03 Oct 2017 (2016)
14. Sharma, D., Jain, S.: Evaluation of stemming and stop word techniques on text classification problem. Int. J. Sci. Res. Comput. Sci. Eng. **3**(2), 1–4 (2015)
15. Bui, D.T., Pradhan, B., Lofman, O., Revhaug, I.: Landslide susceptibility assessment in Vietnam using support vector machines, decision tree, and naïve bayes models. Math Probl Eng. **2012**, 26 (2012). Art ID 974638. https://doi.org/10.1155/2012/974638
16. Mamoun, R., Ahmed, M.: Basic sciences and engineering studies (SGCAC), 2016 Conference, 20-23 Feb. In: IEEE xplore. Khartoum, Sudan (2016). https://doi.org/10.1109/SGCAC.2016.7458011
17. Sharma, N., Sharma, A., Thenkanidiyoor, V., Dileep, A.: Text classification using combined sparse representation classifiers and support vector machines. In: IEEE Conference Publication. http://Ieeexplore.ieee.org; http://ieeexplore.ieee.org/abstract/document/7743280/?section=abstract. Accessed 02 Oct 2017 (2016)
18. Rahman, A., Qamar, U.: A Bayesian classifiers based combination model for automatic text classification. In: IEEE Conference Publication. http://Ieeexplore.ieee.org; http://ieeexplore.ieee.org/document/7883016/. Accessed 01 Oct 2017 (2016)
19. Frakes, W., Fox, C.: Strength and similarity of affix removal stemming algorithms. ACM SIGIR Forum **37**(1), 26–30 (2003)