Information Processing and Management 50 (2014) 104–112

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/infoproman

The impact of preprocessing on text classification

Alper Kursat Uysal *, Serkan Gunal

Department of Computer Engineering, Anadolu University, Eskisehir, Turkiye

ARTICLE INFO

Article history:
Received 27 February 2013
Received in revised form 20 August 2013
Accepted 28 August 2013
Available online 16 September 2013

Keywords:
Pattern recognition
Text categorization
Text classification
Text preprocessing

ABSTRACT

Preprocessing is one of the key components in a typical text classification framework. This paper aims to extensively examine the impact of preprocessing on text classification in terms of various aspects such as classification accuracy, text domain, text language, and dimension reduction. For this purpose, all possible combinations of widely used preprocessing tasks are comparatively evaluated on two different domains, namely e-mail and news, and in two different languages, namely Turkish and English. In this way, contribution of the preprocessing tasks to classification success at various feature dimensions, possible interactions among these tasks, and also dependency of these tasks to the respective languages and domains are comprehensively assessed. Experimental analysis on benchmark datasets reveals that choosing appropriate combinations of preprocessing tasks, rather than enabling or disabling them all, may provide significant improvement on classification accuracy depending on the domain and language studied on.

1. Introduction

Text classification is one of the challenging research topics due to the necessity to organize and categorize growing number of electronic documents worldwide. So far, text classification has been successfully applied to various domains such as topic detection (Ghiassi, Olschimke, Moon, & Arnaudo, 2012), spam e-mail filtering (Gunal, Ergin, Gulmezoglu, & Gerek, 2006), SMS spam filtering (Uysal, Gunal, Ergin, & Gunal, 2012), author identification (Cheng, Chandramouli, & Subbalakshmi, 2011), web page classification (Ozel, 2011) and sentiment analysis (Maks & Vossen, 2012).

A conventional text classification framework consists of preprocessing, feature extraction, feature selection, and classification stages. The preprocessing stage usually contains the tasks such as tokenization, stop-word removal, lowercase conversion, and stemming. The feature extraction stage generally utilizes the vector space model (Salton, Wong, & Yang, 1975) that makes use of the bag-of-words approach (Joachims, 1997). The feature selection stage, most of the time, employs the filter methods such as document frequency (Yang & Pedersen, 1997), mutual information (Liu, Sun, Liu, & Zhang, 2009), information gain (Lee & Lee, 2006), chi-square (Chen & Chen, 2011), Gini index (Shang et al., 2007), and distinguishing feature selector (Uysal & Gunal, 2012). Finally, the classification stage uses well-known and successful pattern classification algorithms, e.g., support vector machines, decision trees, artificial neural networks, and naïve Bayesian classifier (Theodoridis & Koutroumbas, 2008).

While it is verified that the feature extraction (Gunal et al., 2006), feature selection (Feng, Guo, Jing, & Hao, 2012), and classification method (Tan, Wang, & Wu, 2011) have substantial impact on the success of text classification process, the preprocessing step may also influence this success noticeably. Common behaviour in text classification studies is to apply alphabetic tokenization, stop-word removal, lowercase conversion and stemming, without deeply examining their contributions

* Corresponding author. Tel.: +90 5359777370.
E-mail addresses: akuysal@anadolu.edu.tr (A.K. Uysal), serkangunal@anadolu.edu.tr (S. Gunal).

0306-4573/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.ipm.2013.08.006

Table 1
Comparison of the characteristics of this study with previous ones.

Study	TK	SR	LC	ST	Multiple language	Multiple collection	Multi-class vs. binary-class collection	Balanced vs. Imbalanced collection	Feature selection
Song et al. (2005)		√		√		√		√	√
Toman et al. (2006))		√		√	√	√			
Méndez et al. (2006)	√	√		√					√
Pomikálek & Rehurek (2007)	√	√		√		√			√
Duwairi et al. (2009)				√					
Gonçalves et al. (2010)				√					
Torunoglu et al. (2011)		√		√	√	√			
Toraman et al. (2011)		√		√		√			√
The proposed work	√	√	√	√	√	√	√	√	√

to classification accuracy. Few researchers have analysed the influence of preprocessing tasks on text classification at some depth. For instance, effectiveness of stop-word removal and stemming are investigated for English news datasets in (Song, Liu, & Yang, 2005). It is concluded that the impacts of stop-word removal and stemming are small. However, it is suggested to apply stop-word removal and stemming in order to reduce the dimensionality of feature space and promote the efficiency of the text classification system. The effects of lemmatization, stemming and stop-word removal are examined on English and Czech datasets in (Toman, Tesar, & Jezek, 2006). It is stated that stop-word removal improved the classification accuracy in most cases. On the other hand, the influence of word normalization (stemming or lemmatization) on text categorization is negative rather than positive. It is suggested that applying stop-word removal and omitting word normalization can be the best choice for text classification. The use of stop-word removal, stemming and different tokenization schemes on spam e-mail filtering are analysed in (Méndez, Iglesias, Fdez-Riverola, Díaz, & Corchado, 2006). It is reported that performance of SVM is surprisingly good when stemming and stop-word removal are not used. However, some stop-words are rare in spam messages and they should not be removed from feature list in spite of being semantically void. Besides, selection of the right tokenization schema may contribute to the performance of spam filtering. Furthermore, the influence of preprocessing tasks including tokenization, stop-word removal, and stemming are studied on trimmed versions of Reuters 21578, Newsgroups and Springer in (Pomikálek & Rehurek, 2007). It is concluded that selection of stemmer and removal of stop-words has very little impact on the overall classification results. Besides, the effect of stemming on Arabic documents is analysed in (Duwairi, Al-Refai, & Khasawneh, 2009). In this study, two stemming approaches were used to investigate the effects of stemming. It is reported that one of the stemming approaches improves the accuracy of the classifier. In (Gonçalves, Gonçalves, Camacho, & Oliveira, 2010), stemming and pruning are applied in combination for the classification of MEDLINE documents, whereas the other preprocessing parameters such as tokenization, lowercase conversion and stop-word removal are directly applied without comparison in all experiments. It is stated that stemming and pruning contributes to the improvement of the classification accuracy. The impact of stemming and stop-word removal on Turkish texts are evaluated in (Torunoglu, Cakirman, Ganiz, Akyokus, & Gurbuz, 2011) using self-compiled newspaper articles from the internet. It is concluded that stemming and stop-word removal has very little impact on classification accuracy. They claim that the effect of stop-word removal and stemming is visible when the training set size is small. The influence of stemming on Turkish news articles is studied in (Toraman, Can, & Kocberber, 2011) as well. They conducted some experiments with five predefined experimental settings and some of these settings include preprocessing steps. It was observed that preprocessing increased accuracies in most cases.

This paper investigates the impact of widely used preprocessing tasks including tokenization, stop-word removal, lowercase conversion, and stemming in a different manner than those of the abovementioned studies, such that all possible combinations of those preprocessing tasks are considered comparatively in two different languages, namely Turkish and English, and on two different text domains, namely news and e-mails. In this way, contribution of the regarding preprocessing tasks to the classification success at various feature dimensions, possible interactions among these tasks, and also the dependency of these tasks to the language and domain studied on are extensively assessed. In order to clarify the differences of this work from the previous ones, the investigated preprocessing tasks and experimental settings are comparatively presented in Table 1. Tokenization, stop-word removal, lowercase conversion and stemming are abbreviated as TK, SR, LC and ST, respectively. The experimental settings include multiple language, multiple collection, multi-class vs. binary-class collection, balanced vs. imbalanced collection, and feature selection. All these items are briefly described in the following sections.

The remainder of the paper is organized as follows: Section 2 briefly explains the preprocessing methods used in the study. Section 3 describes the experimental settings including combinations of the preprocessing methods, the datasets, the feature selection method, the classification algorithm, and the success measure utilized. Details of the experimental analysis and the related results are provided in Section 4. Finally, some concluding remarks are given in Section 5.

2. Preprocessing methods

Four common preprocessing steps of text classification including tokenization, stop-word removal, lowercase conversion, and stemming are considered within the scope of this paper.

In text processing, tokenization is the procedure of splitting a text into words, phrases, or other meaningful parts, namely tokens. In other words, tokenization is a form of text segmentation. Typically, the segmentation is carried out considering only alphabetic or alphanumeric characters that are delimited by non-alphanumeric characters (e.g., punctuations, whitespace).

Stop-words are the words that are commonly encountered in texts without dependency to a particular topic (e.g., conjunctions, prepositions, articles, etc.). Therefore, the stop-words are usually assumed to be irrelevant in text classification studies, and removed prior to the classification. Stop-words are specific to the language being studied as in the case of stemming.

Another widely used preprocessing step for text classification is lowercase conversion. Since uppercase or lowercase forms of words are assumed to have no difference, all uppercase characters are usually converted to their lowercase forms prior to the classification.

The aim of stemming is to obtain stem, or root, forms of derived words. Since derived words are semantically similar to their root forms, word occurrences are usually computed after applying the stemming on a given text. Stemming algorithms are indeed specific to the language being studied. Though there are different approaches (Zemberek, 2013), the fixed-prefix algorithm (Can et al., 2008) is computationally simple but very effective stemming tool for Turkish language. On the other hand, the stemming algorithm introduced in (Porter, 1980) is commonly employed by researchers for English.

3. Experimental settings

Combinations of the preprocessing methods, the datasets, the feature selection method, the classification algorithm, and the success measure employed within this work are briefly explained in the following subsections.

3.1. Combinations of the preprocessing methods

In this study, all possible combinations of the preprocessing methods are considered as below so that possible interactions between the preprocessing tasks can be revealed. Tokenization is either alphanumeric or alphabetic. Stop-word removal is either ON or OFF; that is, stop-words are either eliminated or kept within text. Lowercase conversion is either ON or OFF; that is, terms are either converted to lowercase or kept in their original forms. Stemming is either ON or OFF; that is, terms are either reduced to their root forms or kept in their inflected forms. Thus, 16 different combinations are obtained as listed in Table 2.

Table 2
Combinations of preprocessing methods.

No.	TK Alphanumeric (0)/ Alphabetic (1)	SR OFF (0)/ON (1)	LC OFF (0)/ON (1)	ST OFF (0)/ON (1)
1	0	0	0	0
2	0	0	0	1
...	...	...	...	...
16	1	1	1	1

Table 3
Turkish/English news dataset.

Class no	Class label	# Training samples	# Testing samples
1	Spor/earn	2877	1087
2	Yazar/acq	1650	719
3	Yaşam/money-fx	538	179
4	Ekonomi/grain	433	149
5	Siyaset/crude	389	189
6	Magazin/trade	369	117
7	Dünya/interest	347	131
8	Astro/ship	197	89
9	Tv/wheat	212	71
10	Sanat/corn	121/181	38/56

3.2. Datasets

The preprocessing methods are evaluated on two different domains, namely e-mail and news, and in two different languages, namely Turkish and English. While Turkish is one of the widely used agglutinative languages worldwide, English is a good example of non-agglutinative languages. For fair evaluation, the numbers of documents within the same domains are kept nearly identical. The e-mail datasets contain 300 training and 100 testing samples for each class, namely spam and legitimate. Class distributions of news datasets are summarized in Table 3. Class distributions of the news datasets are the same except the tenth class.

The e-mail datasets are balanced; on the contrary, the news datasets are imbalanced. One should also note that the email datasets are used to evaluate the preprocessing tasks on binary classification problem, whereas the news datasets are used for the evaluation on multi-class classification.

The Turkish e-mail dataset consisting of spam and legitimate e-mails was previously constituted in (Ergin, Gunal, Yigit, & Aydin, 2012). The English email dataset, on the other hand, is shaped using a subset of well-known Enron dataset (Metsis, Androutsopoulos, & Paliouras, 2006). In the meantime, for news classification, a subset of Milliyet news collection (Can et al., 2008), and top-10 classes of well-known Reuters-21578 (ModApte split) (Asuncion & Newman, 2007) are employed for Turkish and English, respectively.

3.3. Feature selection method

Even though there are filter, wrapper, and embedded approaches for feature selection (Gunal & Edizkan, 2008), researchers prefer the filter methods in text classification problems due to classifier independency and relatively low computation time of the filters (Forman, 2003). In this sense, one of the widely used methods, namely chi-square (CHI2) (Uysal & Gunal, 2012), is employed in this paper to select informative features.

3.4. Classification algorithm and success measure

SVM, which is one of the state-of-the-art pattern classifiers (Kumar & Gopal, 2010), is used as the classification algorithm. Also, the success measure is selected as well-known Micro-F1 score (Uysal & Gunal, 2012) for this study.

4. Experimental work

During the experiments, all possible combinations of the four preprocessing tasks were considered as mentioned before. Various feature sizes including 10, 20, 50, 100, 200, 500, 1000, and 2000 were investigated in the study so that the impact of preprocessing can be comparatively observed within a wide range of feature dimensions. These features were determined using the chi-square based feature selection method as stated previously. The results of the experimental analysis on the four datasets are illustrated in Figs. 1–4. These figures include the plots of the maximum and minimum Micro-F1 scores, and the corresponding combinations of the preprocessing tasks at different feature sizes. In this way, the best and worst cases indicating the impact of preprocessing are highlighted. Considering the maximum Micro-F1 scores, the figures also

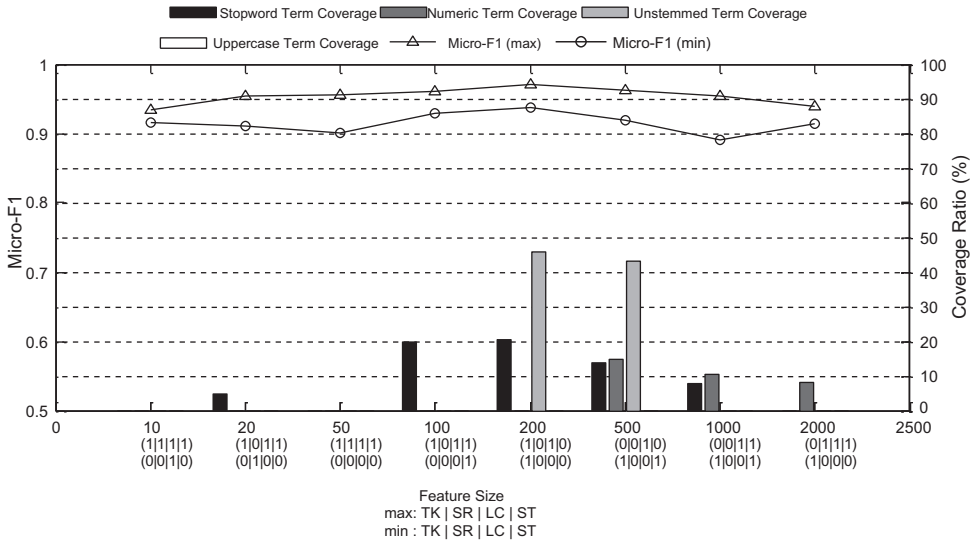

Fig. 1. Experimental results for Turkish email dataset.

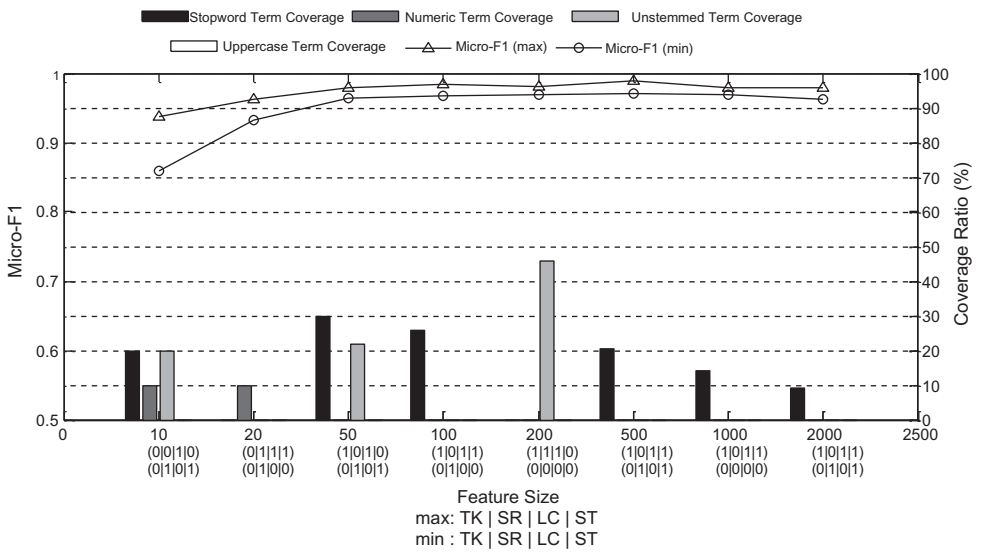

Fig. 2. Experimental results for English email dataset.

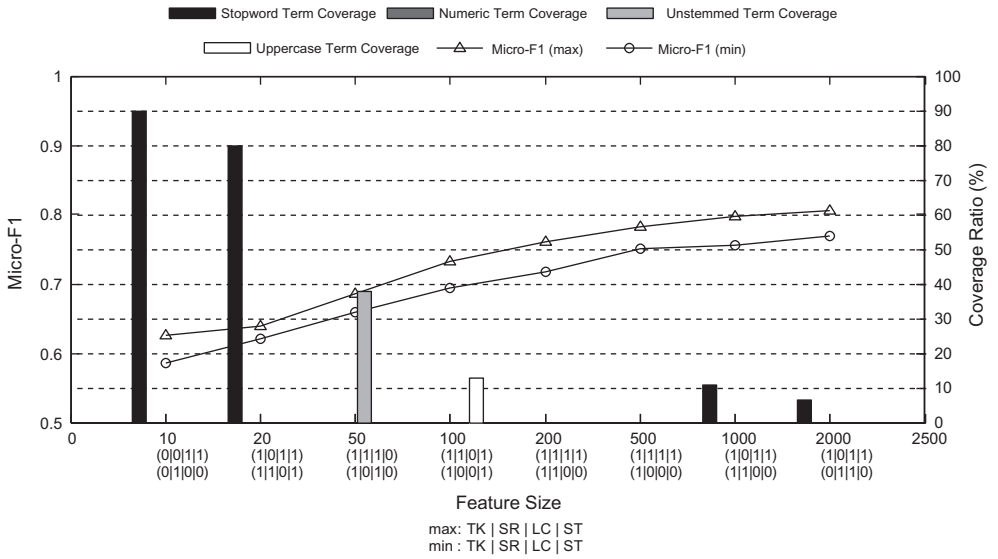

Fig. 3. Experimental results for Turkish news dataset.

provide bar charts for coverage ratios of the terms arisen from the regarding preprocessing tasks to the selected terms at different feature sizes.

In order to clarify the interpretation of those figures, some specific examples are provided as follows:

- In Turkish email dataset, the maximum Micro-F1 score is 0.9713. This score is attained when the feature size is 200 and the preprocessing combination is (TK: 1 | SR: 0 | LC: 1 | ST: 0); that is, tokenization is alphabetic, stop-word removal is OFF, lowercase conversion is ON, and stemming is OFF. In this case, stop-word term coverage and unstemmed term coverage ratios are around 20% and 45%, respectively. In other words, 20% of 200 selected terms are the stop-words whereas 45% of these 200 selected terms are unstemmed terms as well.
- Additionally, the minimum Micro-F1 score is 0.8913 for Turkish email dataset. This score is obtained when the feature size is 1000 and the preprocessing combination is (TK: 1 | SR: 0 | LC: 0 | ST: 1); that is, tokenization is alphabetic, stop-word removal is OFF, lowercase conversion is OFF, and stemming is ON. It should be reminded that, in case of minimum Micro-F1 scores, coverage ratios were not taken into consideration; therefore, they were not displayed in the figures.

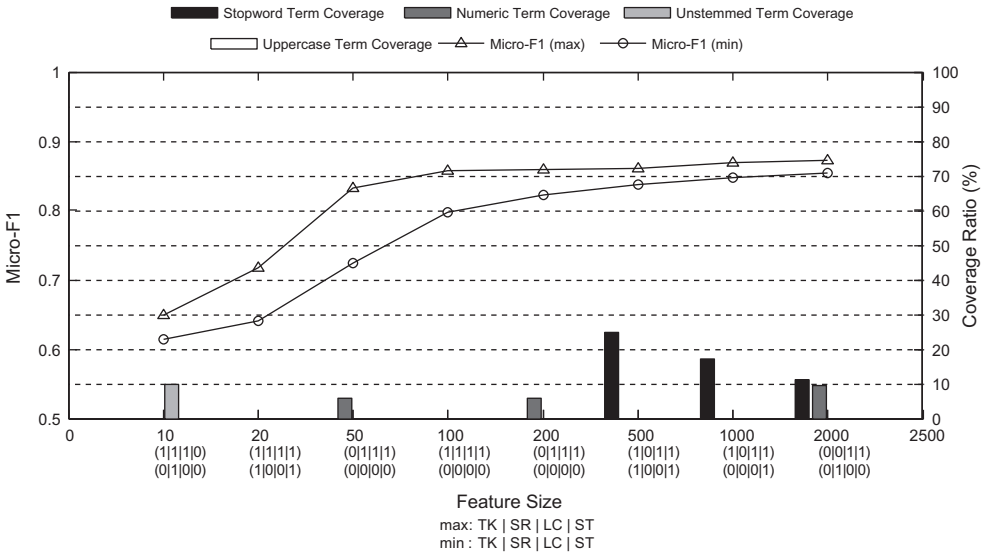

Fig. 4. Experimental results for English news dataset.

Based on all the information provided in these figures, the impact of preprocessing were analysed according to several aspects including accuracy, domain, language, and feature size.

4.1. Accuracy analysis

In this part, Micro-F1 scores attained by all 16 combinations of the preprocessing tasks were measured to assess the impact of preprocessing in terms of accuracy. The maximum Micro-F1 scores among all feature sizes, and the corresponding preprocessing combinations are listed in Table 4 for each dataset.

Considering all four datasets, the difference between the highest and lowest Micro-F1 scores at each feature size for all preprocessing combinations ranged from 0.0113 to 0.1084. More specifically, the difference was between 0.0175 and 0.0625 in Turkish email dataset, between 0.0113 and 0.0787 in English email dataset, between 0.0195 and 0.044 in Turkish news dataset, and between 0.0179 and 0.1084 in English news dataset. The amount of differences in accuracies confirms that the appropriate preprocessing combinations depending on the domain and language may improve the accuracy considerably. In the meantime, inappropriate preprocessing combinations may degrade the accuracy significantly as well.

The impact of preprocessing was also statistically analysed using two-tailed paired t-test over the highest and lowest Micro-F1 scores at each feature size. *P*-values were obtained as (0.000138, 0.016818, 0.000007, and 0.003908) for Turkish e-mail, English e-mail, Turkish news, and English news datasets, respectively. The result for English e-mail dataset was statistically significant with a significance level of 0.05 whereas the remaining three datasets obtained a significance level of 0.01.

Besides, the findings of the proposed study in terms of contribution to the accuracy were compared against the previous works mentioned in the introduction section. The comparison is presented in Table 5, where "+" and "−" signs respectively represent positive and negative impacts, and "N/A" indicate that the corresponding analysis is not available. One should note that the impacts of the preprocessing tasks for the proposed work are based on the combinations listed in Table 4. According to the proposed work, applying alphabetic tokenization, lowercase conversion and stemming, while not applying stop-word removal has positive impact overall. Although the previous works does not analyse combinations of all four preprocessing tasks, it may be stated that the impact of stemming is consistent with most of the previous works, whereas the impact of stop-word removal seems to be opposite. The lowercase conversion was proven to be helpful; however, it cannot be compared to the previous works. Finally, the contribution of appropriate tokenization to the accuracy was positive, which is consistent with the single previous work that is the only one analysing tokenization.

Table 4
Maximum Micro-F1 scores and the corresponding preprocessing tasks.

Dataset	Feature size	Max. Micro-F1	Preprocessing tasks
Turkish e-mail	200	0.9713	TK: 1 | SR: 0 | LC: 1 | ST: 0
English e-mail	500	0.9888	TK: 1 | SR: 0 | LC: 1 | ST: 1
Turkish news	2000	0.8061	TK: 1 | SR: 0 | LC: 1 | ST: 1
English news	2000	0.8719	TK: 0 | SR: 0 | LC: 1 | ST: 1

Table 5
The impacts of the preprocessing methods in various studies.

Study	TK	SR	LC	ST
Song et al. (2005)	N/A	+	N/A	+
Toman et al. (2006)	N/A	+	N/A	–
Méndez et al. (2006)	+	–	N/A	–
Pomikálek & Rehurek (2007)	N/A	+	N/A	+
Duwairi et al. (2009)	N/A	N/A	N/A	+
Gonçalves et al. (2010)	N/A	N/A	N/A	+
Torunoglu et al. (2011)	N/A	+	N/A	+
Toraman et al. (2011)	N/A	+	N/A	+
The proposed work	+ (often)	–	+	+ (often)

4.2. Domain and language analysis

In this part, the impact of preprocessing was evaluated for every domain and language considering the maximum Micro-F1 scores at each case.

Tokenization type in e-mail domain for both languages was alphabetic; however, news domain involved alphabetic tokenization in Turkish, and alphanumeric tokenization in English. To confirm the impact of alphanumeric tokenization, numeric term coverage within the selected feature set of English news dataset was also computed. The coverage ratio was around 10%; in other words, 10% of the selected terms contained numeric characters. When the selected terms were further investigated, it was revealed that specifically the business related news contain significant numbers of numeric terms such as "1st", "2nd" and "3rd". In case of alphabetic tokenization, these terms are regarded as "st", "nd", and "rd" which will cause losing their discriminative powers. On the other hand, there would be no such problem in case of alphanumeric tokenization.

Stop-word removal is not applied in any of the domains and languages. In order to verify the impact of stop-words on classification success, stop-word coverage within the selected feature sets of each dataset was also computed. The coverage ratios were found as 20.50%, 20.60%, 6.50%, and 11.25% for Turkish e-mail, English e-mail, Turkish news, and English news datasets, respectively. It should be noted that e-mails are usually short whereas news are much longer. Even under these circumstances, one can see from those ratios that the presence of the stop-words within the selected terms is very obvious in each domain and language. This finding is really remarkable bearing in mind that most of the text classification studies in the literature remove stop-words directly by assuming them irrelevant.

Lowercase conversion is active in both domains and languages. In other words, all characters should be converted to lowercase without dependency to domain or language. Since lowercase conversion helps grouping the terms which contain the same information with either upper or lower case characters, less amount of features with more discrimination are attained.

Stemming is required in news domain for both languages; on the contrary, it is not applied in Turkish e-mail domain whereas it is necessary for English e-mails. Again, to validate the impact of not applying the stemming in Turkish e-mail domain, unstemmed term coverage within the selected feature set was computed and found to be around 45%. In other words, almost half of the selected terms consist of unstemmed terms. By means of its agglutinative property, deriving large number of words from the same root by appending suffixes is possible for Turkish language. Especially in Turkish e-mails, these derived forms may exist frequently and they can become more discriminative than their root forms.

One can conclude that stop-words should not be removed and characters should be converted to lowercase without dependency to domain or language. However, tokenization type and stemming status may change depending on the domain and language.

Reminding that the email datasets are binary and balanced whereas the news datasets are multi-class and imbalanced, all the statements above may be generalized for different class distributions (balanced vs. imbalanced) and different numbers of classes (binary vs. multi-class) as well.

4.3. Feature size analysis

In this part, the impact of preprocessing was evaluated in terms of dimension reduction. For this purpose, the preprocessing tasks providing the highest Micro-F1 scores at minimum feature size for each dataset were taken into consideration as listed in Table 6.

Table 6
The preprocessing tasks providing the highest accuracy at minimum feature sizes.

Dataset	Min. feature size	Max. Micro-F1	Preprocessing tasks
Turkish e-mail	10	0.9337	TK: 1 | SR: 1 | LC: 1 | ST: 1
English e-mail	10	0.9375	TK: 0 | SR: 0 | LC: 1 | ST: 0
Turkish news	10	0.6490	TK: 1 | SR: 0 | LC: 1 | ST: 1
English news	10	0.6265	TK: 1 | SR: 1 | LC: 1 | ST: 0

Table 7
The preprocessing tasks: maximum accuracy vs. minimum feature size.

Dataset	Preprocessing tasks (Max. accuracy)	Preprocessing tasks (Min. feature size)
Turkish e-mail	TK: 1 | SR: 0 | LC: 1 | ST: 0	TK: 1 | SR: 1 | LC: 1 | ST: 1
English e-mail	TK: 1 | SR: 0 | LC: 1 | ST: 1	TK: 0 | SR: 0 | LC: 1 | ST: 0
Turkish news	TK: 1 | SR: 0 | LC: 1 | ST: 1	TK: 1 | SR: 0 | LC: 1 | ST: 1
English news	TK: 0 | SR: 0 | LC: 1 | ST: 1	TK: 1 | SR: 1 | LC: 1 | ST: 0

In e-mail domain, as a common behaviour in both languages, lowercase conversion was applied. Status of the remaining preprocessing tasks, however, varied depending on the language.

In news domain, alphabetic tokenization and lowercase conversion were common preprocessing tasks in both languages whereas status of stop-word removal and stemming were opposite for each language. While stop-word removal was applied in English news, stemming was required for Turkish news.

For Turkish language, stop-word removal was applied only on e-mail domain while alphabetic tokenization, lowercase conversion, and stemming were commonly applied in both domains. Stop-word coverage ratio within the selected feature set of Turkish news dataset was computed as 90%. Hence, it is obvious that the stop-words have a dominant impact among all terms at minimum feature dimension of Turkish news domain only.

For English language, lowercase conversion was applied, but stemming was not active in both domains. Unstemmed term coverage ratios within the selected feature sets were 20% and 10% for e-mail and news datasets in English, respectively. On the other hand, the status of tokenization type and stop-word removal were opposite for each language. While alphabetic tokenization was applied in English news domain, stop-words were kept in English e-mail domain. Stop-word coverage ratio within the selected feature set of English e-mail dataset was computed as 20%, and numeric term coverage ratio was found as 10% in English news dataset as well.

4.4. Maximum accuracy vs. minimum feature size

In this section, the preprocessing tasks, which provided the maximum Micro-F1 scores, were compared to the ones providing the highest Micro-F1 scores at minimum feature size for each domain and language. The comparison is listed in Table 7.

It is obvious from the table that lowercase conversion is the only preprocessing task that is common in all cases. In other words, lowercase conversion should be applied to achieve either maximum accuracy or minimum feature size with the highest accuracy for all domains and languages. Another common behaviour was related to Turkish language such that alphabetic tokenization should be applied in Turkish, regardless of the domain, to achieve either maximum accuracy or minimum feature size. No other common behaviour was observed related to the remaining preprocessing tasks for any domain or language.

5. Conclusions

In this paper, the influence of widely used preprocessing tasks on text classification was thoroughly examined in two different domains and languages. The examination was carried out using all possible combinations of the preprocessing tasks by considering various aspects such as accuracy, domain, language, and dimension reduction. Extensive experimental analysis revealed that appropriate combinations of preprocessing tasks depending on the domain and language may provide a significant improvement on classification accuracy whereas inappropriate combinations may degrade the accuracy as well. Consequently, preprocessing step in text classification is as important as feature extraction, feature selection, and classification steps.

Although there are particular preprocessing tasks such as lowercase conversion that improve classification success in terms of accuracy and dimension reduction regardless of domain and language, there is no unique combination of preprocessing tasks providing successful classification results for every domain and language studied on. Therefore, for a text classification problem on any domain and in any language, researchers should carefully analyse all possible combinations of the tasks rather than completely/individually enabling or disabling them. Otherwise, classification results may significantly differ. Another interesting finding of the study was the importance of stop-words while most of the text classification studies in the literature assume the stop-words irrelevant.

Since all four datasets studied on the paper have distinct characteristics in terms of domain, language, class distribution, and number of classes, the outcome of this paper may be generalized for the other text collections as well.

References

Asuncion, A., & Newman, D. J. (2007). UCI machine learning repository. In C. A. Irvine (Ed.), University of California, Department of Information and Computer Science.

Can, F., Kocberber, S., Balcik, E., Kaynak, C., Ocalan, H. C., & Vursavas, O. M. (2008). Information retrieval on Turkish texts. *Journal of the American Society for Information Science and Technology, 59*, 407–421.

Chen, Y.-T., & Chen, M. C. (2011). Using chi-square statistics to measure similarities for text categorization. *Expert Systems with Applications, 38*, 3085–3090.
Cheng, N., Chandramouli, R., & Subbalakshmi, K. P. (2011). Author gender identification from text. *Digital Investigation, 8*, 78–88.
Duwairi, R., Al-Refai, M. N., & Khasawneh, N. (2009). Feature reduction techniques for Arabic text categorization. *Journal of the American Society for Information Science and Technology, 60*, 2347–2352.
Ergin, S., Gunal, E. S., Yigit, H., & Aydin, R. (2012). Turkish anti-spam filtering using binary and probabilistic models. *AWERProcedia Information Technology and Computer Science, 1*, 1007–1012.
Feng, G., Guo, J., Jing, B. Y., & Hao, L. (2012). A Bayesian feature selection paradigm for text classification. *Information Processing & Management, 48*, 283–302.
Forman, G. (2003). An extensive empirical study of feature selection metrics for text classification. *Journal of Machine Learning Research, 3*, 1289–1305.
Ghiassi, M., Olschimke, M., Moon, B., & Arnaudo, P. (2012). Automated text classification using a dynamic artificial neural network model. *Expert Systems with Applications, 39*, 10967–10976.
Gonçalves, C. A., Gonçalves, C. T., Camacho, R., & Oliveira, E. C. (2010). The impact of pre-processing on the classification of MEDLINE documents. In *Proceedings of the 10th international workshop on pattern recognition in information systems* (pp. 53–61).
Gunal, S., & Edizkan, R. (2008). Subspace based feature selection for pattern recognition. *Information Sciences, 178*, 3716–3726.
Gunal, S., Ergin, S., Gulmezoglu, M. B., & Gerek, O. N. (2006). On feature extraction for spam e-mail detection. *Lecture Notes in Computer Science, 4105*, 635–642.
Joachims, T. (1997). A probabilistic analysis of the Rocchio algorithm with tfidf for text categorization. In *14th international conference on machine learning* (pp. 143–151). Morgan Kaufmann Publishers Inc.
Kumar, M. A., & Gopal, M. (2010). A comparison study on multiple binary-class SVM methods for unilabel text categorization. *Pattern Recognition Letters, 31*, 1437–1444.
Lee, C., & Lee, G. G. (2006). Information gain and divergence-based feature selection for machine learning-based text categorization. *Information Processing & Management, 42*, 155–165.
Liu, H., Sun, J., Liu, L., & Zhang, H. (2009). Feature selection with dynamic mutual information. *Pattern Recognition, 42*, 1330–1339.
Maks, I., & Vossen, P. (2012). A lexicon model for deep sentiment analysis and opinion mining applications. *Decision Support Systems, 53*, 680–688.
Méndez, J. R., Iglesias, E. L., Fdez-Riverola, F., Díaz, F., & Corchado, J. M. (2006). *Tokenising, stemming and stopword removal on anti-spam filtering domain. Proceedings of the 11th spanish association conference on current topics in artificial intelligence*. Springer-Verlag: Santiago de Compostela, Spain.
Metsis, V., Androutsopoulos, I., & Paliouras, G. (2006). Spam filtering with naive Bayes – Which naive Bayes? In 3rd conference on email and anti-spam (Vol. 17, pp. 28–69).
Ozel, S. A. (2011). A web page classification system based on a genetic algorithm using tagged-terms as features. *Expert Systems with Applications, 38*, 3407–3415.
Pomikálek, J., & Rehurek, R. (2007). The Influence of preprocessing parameters on text categorization. *International Journal of Applied Science, Engineering and Technology, 4*, 430–434.
Porter, M. F. (1980). An algorithm for suffix stripping. *Program, 14*, 130–137.
Salton, G., Wong, A., & Yang, C. S. (1975). A vector space model for automatic indexing. *Communications of the ACM, 18*, 613–620.
Shang, W., Huang, H., Zhu, H., Lin, Y., Qu, Y., & Wang, Z. (2007). A novel feature selection algorithm for text categorization. *Expert Systems with Applications, 33*, 1–5.
Song, F. X., Liu, S. H., & Yang, J. Y. (2005). A comparative study on text representation schemes in text categorization. *Pattern Analysis and Applications, 8*, 199–209.
Tan, S., Wang, Y., & Wu, G. (2011). Adapting centroid classifier for document categorization. *Expert Systems with Applications, 38*, 10264–10273.
Theodoridis, S., & Koutroumbas, K. (2008). *Pattern recognition* (4th ed.). Academic Press.
Toman, M., Tesar, R., & Jezek, K. (2006). Influence of word normalization on text classification. In Proceedings of the 1st international conference on multidisciplinary information sciences & technologies (Vol. 2, pp. 354–358). Merida, Spain.
Toraman, C., Can, F., & Kocberber, S. (2011). Developing a text categorization template for Turkish news portals. In International symposium on innovations in intelligent systems and applications (INISTA) (pp. 379–383).
Torunoglu, D., Cakirman, E., Ganiz, M. C., Akyokus, S., & Gurbuz, M. Z. (2011). Analysis of preprocessing methods on classification of Turkish texts. In International Symposium on Innovations in Intelligent Systems and Applications (INISTA) (pp. 112–117).
Uysal, A. K., Gunal, S., Ergin, S., & Gunal, E. S. (2012). A novel framework for sms spam filtering. In Proceedings of the IEEE international symposium on innovations in intelligent systems and applications. Trabzon, Turkiye.
Uysal, A. K., & Gunal, S. (2012). A novel probabilistic feature selection method for text classification. *Knowledge-Based Systems, 36*, 226–235.
Yang, Y., & Pedersen, J. O. (1997). A comparative study on feature selection in text categorization. In *14th international conference on machine learning* (pp. 412–420). Morgan Kaufmann Publishers Inc.
Zemberek. <http://code.google.com/p/zemberek/> (Accessed January 2013).