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

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.