Expert Systems With Applications 61 (2016) 282–289

Contents lists available at ScienceDirect

Expert Systems With Applications

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

Using information retrieval for sentiment polarity prediction

Anderson Uilian Kauer, Viviane P. Moreira*

Institute of Informatics, Universidade Federal do Rio Grande do Sul, Porto Alegre, RS, Brazil

ARTICLE INFO

Article history:
Received 18 January 2016
Revised 18 May 2016
Accepted 25 May 2016
Available online 30 May 2016

Keywords:
Sentiment analysis
Opinion mining
Information retrieval
Polarity classification
Twitter

ABSTRACT

Social networks such as Twitter are used by millions of people who express their opinions on a variety of topics. Consequently, these media are constantly being examined by sentiment analysis systems which aim at classifying the posts as positive or negative. Given the variety of topics discussed and the short length of the posts, the standard approach of using the words as features for machine learning algorithms results in sparse vectors. In this work, we propose using features derived from the ranking generated by an Information Retrieval System in response to a query consisting of the post that needs to be classified. Our system can be fully automatic, has only 24 features, and does not depend on expensive resources. Experiments on real datasets have shown that a classifier that relies solely on these features outperforms established baselines and can reach accuracies comparable to the state-of-the-art approaches which are more costly.

1. Introduction

With over 500 million posts a day[1], Twitter[2] has consolidated itself as a major forum for expressing personal opinions on a variety of topics. Because of its popularity, this microblogging service has been the target of numerous studies from a broad range of research areas including Psychology, Sociology, Marketing, and Computer Science. For example, in Mostafa (2013), the analysis of tweets is used to determine the sentiment towards sixteen global brands.

Sentiment analysis, also called Opinion Mining, is dedicated to the computational study of opinions and sentiments expressed in text (Pang & Lee, 2008). This topic has been attracting increasing attention from the research community. Out of the different aspects of opinions that can be studied, the *polarity* of sentiments is the most well investigated. It consists in predicting whether the opinion expressed in the text is *positive* or *negative*.

While most of the research focuses on product reviews, recently, a number of studies on Twitter posts (or simply *tweets*) have emerged. Sentiment Analysis on Twitter can be done at three different levels: (*i*) entity, (*ii*) tweet, or (*iii*) expression. Entity-level analysis deals with discovering the overall opinion about an entity or topic, tweet-level analysis identifies the polarity of individual tweets, and expression level analysis deals with specific phrases within a tweet. Our focus is on the second – tweet-level analysis. The added challenge of analysing tweets (compared to product reviews) is their shorter length – at most 140 characters – which results in very sparse vector representations. In addition, the variety of topics, and the informal vocabulary, characterised by slangs, abbreviations, and misspellings, pose added difficulties to its computational treatment.

Successful approaches for polarity classification on tweets use one or more of the following: resources such as lexicons (which are sometimes manually created) (Fersini, Messina, & Pozzi, 2016; Speriosu, Sudan, Upadhyay, & Baldridge, 2011; Zhang, Ghosh, Dekhil, Hsu, & Liu, 2011), costly preprocessing such as part-of-speech tagging (Fersini et al., 2016; Go, Bhayani, & Huang, 2009; Hu, Tang, Tang, & Liu, 2013; Saif, He, & Alani, 2012), numerous features (Fersini et al., 2016; Go et al., 2009; Saif et al., 2012; Speriosu et al., 2011; Zhang et al., 2011) large amounts of training data (Bakliwal et al., 2012), and elaborated machine learning methods such as classifier ensembles (Coletta, da Silva, Hruschka, & Hruschka, 2014; Martìn-Valdivia, Martìnez-Cámara, Perea-Ortega, & Ureña López, 2013; da Silva, Hruschka, & Hruschka, 2014). In this work, we propose a method called Sentiment Analysis Based on Information Retrieval (SABIR) which uses none of the above. We show that classification accuracy comparable to the state-of-the-art can be achieved with a single classification algorithm using only 24 features. Unlike existing approaches, we do not use the words of the tweets as features. Our features are derived from the ranking generated by an Information Retrieval System in response to a query q which consists of the tweet that we wish to classify. The

* Corresponding author.
E-mail addresses: aukauer@inf.ufrgs.br (A.U. Kauer), viviane@inf.ufrgs.br (V.P. Moreira).

[1] http://www.internetlivestats.com/twitter-statistics/
[2] http://www.twitter.com

http://dx.doi.org/10.1016/j.eswa.2016.05.038
0957-4174/

ranking has the n most similar tweets for which we already know the class in decreasing order of similarity to the unlabelled tweet q. The rationale is to leverage information of the class of the similar posts to classify q.

We have carried out experiments with four datasets of tweets which have been used in similar studies. Since the training data for the classification system can be generated without manual annotation (Barbosa & Feng, 2010; Go et al., 2009), SABIR can be fully automatic. Our results have shown that there is no significant difference between SABIR and the best baseline classifier we implemented using over one thousand features.

2. Related work

The literature on sentiment analysis abounds on methods for classifying the polarity of opinionated texts, such as product reviews (Pang & Lee, 2008). In recent years, in interest on treating tweets has grown and several approaches were proposed. Martínez-Cámara, Martín-Valdivia, Urena-López, and Montejo-Ráez (2014) present a survey devoted exclusively to this topic. The task of identifying the polarity of a tweet is typically modelled as a classification problem. Its solution relies on machine learning algorithms and sentiment lexicons (*i.e.,* a list of opinion words and their polarities).

Building a supervised classifier requires labelled training examples. Since hand-labelled data is very costly to obtain, alternative approaches have relied on automatic but noisy labels. Go et al. (2009) use emoticons to assign the label and show that various machine learning algorithms have accuracies of over 80% when trained over this noisily labelled data. With the same goal, Barbosa and Feng (2010) have used existing websites on sentiment analysis of tweets as a source of training data.

Most of the related work use a bag-of-words (BoW) approach in which the words in the tweets are used as features (*i.e., n*-g, usually unigrams and bigrams) (Davidov, Tsur, & Rappoport, 2010; Fersini et al., 2016; Go et al., 2009; Saif et al., 2012; Vosoughi, Zhou, & roy, 2015), combined to sentiment lexicons (Fersini et al., 2016; Speriosu et al., 2011; Zhang et al., 2011). Part-of-speech (POS) tags have also been widely used (Barbosa & Feng, 2010; Fersini et al., 2016; Go et al., 2009; Hu et al., 2013; Saif et al., 2012). Emoticons and Twitter specific features (such as retweets, hashtags, and follower graph) have also been exploited.

Ghiassi, Skinner, and Zimbra (2013) examined traditional feature selection strategies and proposed a semi-automatic method to identify useful features. They generated a lexicon with 187 features derived for a dataset of tweets on Justin Bieber. The authors mention that whether the features could be used to classify posts about other entities is still to be established.

Carvalho, Prado, and Plastino (2014) applied a genetic algorithm to select the paradigm words that will help determine the polarity of other words. They found an improvement compared to approaches in which the paradigm words are fixed.

More recently, da Silva et al. (2014), following Lin and Kolcz (2012), have analysed the use of feature hashing to solve the problem of the sparsity of the vectors created from tweets when the words are used as features, *i.e.,* in a BoW approach. In this approach, the features are hash integers rather than strings. The authors report that feature hashing was outperformed by the BoW approach in all but one dataset (HCR).

Vosoughi et al. (2015) enriched a standard bigram model with features representing contextual information about the tweet. The added features were: the US state, the hour of the day, the day of the week, the month and the author of the tweet. Their experiments identified a gain of 10% in accuracy with the added features. The limitation of this approach is that such contextual information is not available in the standard datasets used in the literature.

Fersini et al. (2016) applied feature expansion by adding features which explore the presence of adjectives, emoticons, emphatic and onomatopoeic expressions and also expressive lengthening of words to a BoW model. The authors found that adjectives were the most discriminative of those features bringing gains ranging from 0.49 to 4% in accuracy depending on the dataset. In order to derive these added features, a POS-tagger and a sentiment lexicon are necessary.

To avoid the need of labelled training data, a sentiment lexicon with words and their assigned polarities can be used for polarity prediction. While such lexicons are costly to generate, there are a few of them readily available. The limitation with regards to their application on Tweeter data is that tweets tend to have slangs, misspellings, and colloquial expressions which will not typically be included in a lexicon. To overcome this problem, Saif, He, Fernandez, and Alani (2016) proposed an approach to capture the semantic information inferred from co-occurrence patterns to generate a dynamic representation of the words. Their experiments have shown gains in two out of three datasets compared to lexicon-based baselines.

While the most popular machine learning algorithms for sentiment analysis are Naïve Bayes, Multinomial Naïve Bayes, Support Vector Machines, and Maximum Entropy, recent studies (Coletta et al., 2014; Fersini et al., 2016; da Silva et al., 2014) have explored the combination of classifiers. Classifier and cluster ensembles (Coletta et al., 2014; Fersini et al., 2016; da Silva et al., 2014) improved classification quality but bring extra computational costs.

3. Classifying the polarity of tweets

A Twitter post, or *tweet*, expresses the opinions or sentiments of its author about an entity. As mentioned in Section 2, the traditional approach for classifying the polarity of a tweet is to implement a classifier using unigrams as features (*i.e.,* BoW). However, this tends to result in a very sparse set of features because of the large diversity of vocabulary in the tweets. Our hypothesis is that twitter posts that are similar tend to belong to the same class. Thus, information about the class of the n most similar posts may help classify the polarity of a given unlabelled tweet. This notion is analogous to the one in Weren et al. (2014) which was applied to predicting gender and age of social media posts.

Unlike approaches which work by selecting the most representative terms to be used as features, both for sentiment analysis in general (Deng, Luo, & Yu, 2014; Nicholls & Song, 2010; O'Keefe & Koprinska, 2009) and specifically for tweets (Carvalho et al., 2014; Ghiassi et al., 2013; da Silva et al., 2014), we do not employ the words in the tweets as features. Thus, SABIR is not a term selection approach.

SABIR is composed of two steps. The first step consists in obtaining the n most similar posts in relation to the tweet we wish to classify. In the second step, we use information about these n posts as features (or attributes) to train a supervised classifier. An overview of the process is shown in Fig. 1.

More formally, our goal is: given a set of tweets $T = \{t_1, t_2, ..., t_m\}$ for which the class $c_i \in \{positive(+), negative(-)\}$ is known and a set of unlabelled tweets $Q = \{q_1, q_2, ..., q_p\}$ (*i.e.,* for which the class is unknown), we use information about the similarity of each element $q_i \in Q$ in relation to the elements $t_j \in T$ to predict the class of q_i. Information on the similarity of the tweets is taken from an Information Retrieval System as described in Section 3.1.

Our approach can work with one or more sets of labelled instances, as those are required for indexing and for training. In our experiments, we have explored two settings: (*i*) using three sets of data: a large dataset of noisily labelled tweets for indexing; a manually labelled dataset for training, and an unlabelled dataset