Neurocomputing 187 (2016) 126–132

Contents lists available at ScienceDirect

Neurocomputing

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

Sentiment analysis via integrating distributed representations of variable-length word sequence

Zhijian Cui [a,b], Xiaodong Shi [a,b,*], Yidong Chen [a,b]

[a] Fujian Key Lab of the Brain-like Intelligent Systems, Xiamen University, Xiamen, 361005 Fujian, China
[b] Department of Cognitive Science, School of Information Science and Technology, Xiamen University, Xiamen, 361005 Fujian, China

ARTICLE INFO

Article history:
Received 25 May 2015
Received in revised form
20 July 2015
Accepted 21 July 2015
Available online 2 December 2015

Keywords:
Sentiment analysis
Feature weight
Word representation
Natural language processing

ABSTRACT

Sentiment analysis aims to identify the overall emotional polarity of a given text. It is a nontrivial task to perform sentiment analysis as sentiment information is crucial in many natural language processing applications. Previous n-gram features is derived from a bag-of-n-gram model which is insensitive to the order of the n-gram. To address this problem, we integrates distributed semantic features of word sequence, with fixed-size independent of the length of the word sequence. We also learn distributed semantic features of part-of-speech (POS) sequence as additional syntax-related clues to sentiment analysis. Our semantic features are able to capture both local contexts and global contexts automatically without involving comprehensive task-specific feature engineering. We validate the effectiveness of the method on our constructed sentiment dataset. Experiment results show that our method are able to improve the quality of sentiment analysis when comparing with several competitive baselines.

1. Introduction

Sentiment analysis is one of the branches of natural language processing (NLP), and it mainly aims to identify the overall emotional polarity of a given text. Generally, the polarity consists of positive, negative and neural. Most of the existing methods in sentiment analysis can be divided into two categories: (1) sentiment knowledge based methods, and (2) classified features based methods.

The sentiment knowledge based methods normally identifies the emotional polarity of a given text based upon manually defined sentiment lexicons. Intuitively, constructing sentiment lexicons are important but involving intensive labor investment. Also, sentiment lexicons are often language-dependent which constrain their applications in other languages.

The classified features based methods treat the sentiment analysis as a classified task and thus identifies the emotional polarity by means of a variety of sentiment features. Conventional methods practically follow the work [22] to construct a sentiment classifier based on annotated polarity training instances. To some extent, these methods are effective; however, they require excessive task-specific feature engineering. It is necessary to develop a better learning algorithm to model the features which require less linguistic intuition and features selection process. Hence, recent studies [27,34,11,13] focus on projecting the words, phrases and even the whole sentences into distributed semantic representations. The projection is language independent and able to obtain dense and real-valued representations without resorting to manually defined features. Based on these features, a classifier is further used to identify the emotional polarity.

We believe previous methods not being able to explore the features comprehensively as most of them are unable to capture the global contexts within the sentences. In some conditions, we should resort to a wider scope to identify the overall emotional polarity. In our method, both local and global contexts are integrated as features in sentiment analysis. The local contexts are captured via n-gram features. Because different words contribute different impacts on the sentiment analysis, we use a feature weight matrix to re-weight the phrase embedding. The global contexts are captured via word sequence features and POS sequence features. Word sequence features mainly inherit the semantic of words within the whole instances. POS sequence features are treated as additional syntax-related clues to sentiment analysis. More precisely, our framework for sentiment analysis is shown in Fig. 1.

Our method is capable of capturing the semantic features of the n-gram and the whole sequence during context modeling. In Fig. 1, we have input word sequence "⟨B⟩ Oil price is likely to rebound ⟨B⟩" where label "⟨B⟩" represents the word sequence boundary. First, we concatenate the word embedding within the sliding window (represented with orange) of length n into a longer vector [5]. Here, word embedding (black dots in the purple frame) is a

* Corresponding author at: Fujian Key Lab of the Brain-like Intelligent Systems, Xiamen University, Xiamen, 361005 Fujian, China. Tel.: +86 18959288068.
E-mail address: mandel@xmu.edu.cn (X. Shi).

http://dx.doi.org/10.1016/j.neucom.2015.07.129
0925-2312/

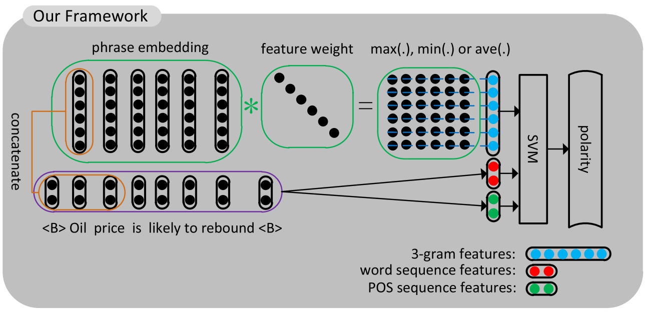

Fig. 1. An illustration of our method in sentiment analysis. The input is the word sequence "⟨B⟩ Oil price is likely to rebound ⟨B⟩". The output is the SVM based classified features where the blue dots, red dots and green dots represent the 3-gram features, word sequence features, POS sequence features, respectively. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

distributed word representation in the form of a dense, low-dimensional and real-valued vector. In our example, n is set to 3, such that we obtain the first 3-gram "⟨B⟩ Oil price", the second 3-gram "Oil price is", etc. All these longer vectors are formed as the phrase embedding. By doing so, we transform a word sequence into a matrix, of which size is $(2*e+1)*n$, where e is the size of word embedding and n is the length of the word sequence excluding the word boundary.

Generally, each word in the training instance contributes different impacts on the sentiment analysis. For example, "rebound" tends to indicate stronger emotions than "price". Hence, inspired by previous method [2], we re-weight the representations of the target instances based on a feature weight matrix.

We believe that the n-gram features somehow neglect the order information of the whole sequence because of the restriction of the size of the window. Hence, we follow previous implement PV-DM [13] to obtain distributed semantic features of word sequence, with fixed-size independent of the sequence length. In addition, we also learn distributed POS sequence features as additional syntax-related clues to train a better sentiment classifier. Doing so, our sentiment classifier is capable of providing a better performance when identifying the emotional polarity of the test instances.

Concisely, our main contributions lie in two aspects:

- Rather than resorting to manually designed features and external sentiment resources, our method is able to learn the classified features without involving excessive task-specific feature engineering.
- Instead of merely considering the n-gram features, we are able to integrate word sequence features, capable of inheriting the semantic of words across the whole sequence during context modeling. In addition, we learn distributed POS sequence features, providing additional syntax-related clues in sentiment analysis.

The remainder of this paper is organized as follows. Section 2 summarizes and compares related works. Section 3 presents our method on how to learn the n-gram features with respect to the word sequence. Then we elaborate our method on how to learn distributed semantic features of word and POS sequence. Section 4 describes our experiments and shows results with discussions. Finally, Section 5 concludes and outlines future directions.

2. Related work

In this section, we present a brief review with respect to the related work into two aspects: (1) sentiment analysis, and (2) learning distributed representations.

2.1. Sentiment analysis

Sentiment analysis, broadly speaking, is the technique that allows to identify the emotional polarity within the text. Generally, the methods employed in sentiment analysis can be divided into two categories: (1) sentiment knowledge based methods, and (2) classified features based methods.

The sentiment knowledge based methods [22,7,29,32] mainly resort to pre-defined sentiment lexicons where each sentiment word is manually annotated with a emotional polarity. With these sentiment words and a set of sentiment computation paradigms, then the sentiment of a sentence or document is induced directly. Despite their wider applications to sentiment analysis, sentiment lexicons is domain-dependent and often require excessive human investment.

The classified features based methods [22,21,6,12] consider the sentiment analysis as a task of text classification based upon their emotional polarity. The main idea is to learn the representative features for a sentiment classifier during training stage. At the test stage, the learned classifier is applied directly to predict the emotional polarity of the test sentence or document. Overly, most of existing methods involve excessive feature engineering. These features are regarded as the gold standard to discriminate different emotional polarities. Hence, if they are not selected or defined properly, then the accuracy of the system will degrade sharply. A representative work is found in [20], which develops a sentiment classifier via a great number of hand-crafted features, and this classifier is able to achieve a promising performance in SemEval 2013 Twitter Sentiment Classification Task.

More recently, [30,31] learns the distributed representation for sentiment analysis, which jointly incorporates the syntactic context and sentiment polarity. The distributed representations of sentiment-specific words are proved to be informative in sentiment analysis. In fact, many researchers attempt to model words, phrases and even the whole sentences with distributed semantic representations. Early works on using neural networks to learn phrase representations can be found in [8], which uses a recurrent neural network to learn dense real-value representations of the phrases. More recently, RAE based methods have been developed in many NLP tasks [24,27,25,14,28].

In addition, the cross-domain sentiment classification [3,10] and cross-lingual sentiment classification [33,15,17] are also received tremendous attentions.

Unlike conventional methods, our method learns the distributional features without involving excessive task-specific feature engineering. Also, rather than merely considering the local features, we are able to inherit the semantic of words across the whole sequence. We believe the POS features are useful in sentiment analysis and thus learn the POS sequence features as additional syntax-related clues in sentiment analysis.

2.2. Learning distributed representation

Conventional methods such as [22] use a one-hot vector to represent each word. The length of such vector is the size of the vocabulary, and only one dimension in the vector are set to 1 while the rest are all set to be 0. However, the one-hot vector contains severe data sparsity issue and somehow unable to capture complex linguistics knowledge.

According to the deep learning framework [1], distributed semantic representations of words are learned in an unsupervised fashion, capable of providing a simple and generalized features to enhance existing supervised NLP tasks. Ref. [10] takes advantage of Stacked Denoising Autoencoders for sentiment classification. Socher et al. develops Recursive Neural Network [24,26], Recursive Vector-Matrix [25] and Recursive Neural Tensor Network [28] to