Learning Sentiment-Specific Word Representations from Tweets

(Project Number - 3)

A Project Report

Submitted by

Group - 9

Abhinaba Sarkar	201405616
Shashank S	201405599
Veerendra Bidare	201405571
Lokesh Walase	201405597

Under the guidance of

Dr. Vikram Pudi

Under the mentorship of

Ganesh Jawahar

For the course

Data Warehousing and Data Mining

IIIT, Hyderabad

November, 2015

***Abstract*—The project demos a method that learns word em bedding for Twitter sentiment classification using neural network. It is implementation of [1].**

Most existing algorithms for learning continuous word representations typically only model the syntactic context of words but ignore the sentiment of text. This is problematic for sentiment analysis as they usually map words with similar syntactic context but opposite sentiment polarity, such as good and bad, to neighboring word vectors. We address this issue by learning sentiment specific word embedding (SSWE), which encodes sentiment information in the continuous representation of words. Specifically, we develop three neural networks to effectively incorporate the supervision from sentiment polarity of text (e.g. sentences or tweets) in their loss function s. To obtain large scale training corpora, we learn the sentiment specific word em bedding from massive distant supervised tweets collected by positive and negative emoticons. The project shows that (1) the SSWE feature performs comparably with hand crafted features in the top performed system; (2) the performance is further improved by concatenating SSWE with existing feature set.

***Keywords*—SSWE, Neural Network, Sentiment Polarity**

I. Introduction

Twitter sentiment classification has attracted in creasing research interest in recent years (Jiang et al., 2011; Hu et al., 2013). The objective is to classify the sentiment polarity of a tweet as positive,negative or neutral.

Feature engineering is important but labor intensive. It is therefore desirable to discover explanatory factors from the data and make the learning algorithms less dependent on extensive fea ture engineering (Bengio, 2013). For the task of sentiment classification, an effective feature learning method is to compose the representation of a sentence (or document) from the representation s of the words or phrases it contains (Socher et al., 2013b; Yessenalina and Cardie, 2011). Accordingly, it is a crucial step to learn the word representation (or word embedding), which is a dense, low dimensional and real valued vector for a word. Although existing word embedding learn ing algorithms (Collobert et al., 2011; Mikolov et al., 2013) are intuitive choices, they are not effective enough if directly used for sentiment classification. The most serious problem is that traditional methods typically model the syntactic con text of words but ignore the sentiment information of text. As a result, words with opposite polarity, such as good and bad, are mapped into close vectors. It is meaningful for some tasks such as pos tagging (Zheng et al., 2013) as the two words have similar usages and grammatical roles, but it becomes a disaster for sentiment analysis as they have the opposite sentiment polarity.

In the proposed sentiment specific word embedding (SSWE) for sentiment analysis, we encode the sentiment information in to the continuous representation of words, so that it is able to separate good and bad to opposite ends of the spectrum.To this end, we extend the existing word embedding learning algorithm (Collobert et al., 2011) and develop three neural net works to effectively incorporate the supervision from sentiment polarity of text (e.g. sentences or tweets) in their loss functions. We learn the sentiment specific word embedding from tweet s, leveraging massive tweets with emoticons as distant supervised corpora without any manual an notations. These automatically collected tweet s contain noises so they cannot be directly used as gold training data to build sentiment classifier s, but they are effective enough to provide weakly supervised signals for training the sentiment specific word embedding.

II. Problem Statement

To incorporate the sentiment information of sentences to learn continuous representations for words and phrases. We extend the existing word embedding learning algorithm (Collobert et al., 2011) and develop three neural networks to learn SSWE.

III. Proposed Solution

A. Data Description

Twitter is a social networking and microblogging service that allows users to post real time messages, called tweets. Tweets are short messages, restricted to 140 characters in length. Due to the nature of this microblogging service (quick and short messages), people use acronyms, make spelling mistakes, use emoticons and other characters that express special meanings. Following is a brief terminology associated with tweets. Emoticons: These are facial expressions pictorially represented using punctuation and letters; they express the users mood. Target: Users of Twitter use the @ symbol to refer to other users on the microblog. Referring to other users in this manner automatically alerts them. Hashtags: Users usually use hashtags to mark topics. This is primarily done to increase the visibility of their tweets.

We acquired 23058819 annotated Twitter data (tweets). 4899561 - negative, 18159258 - positive

B. Unified Model ($SSWE_u$)

The C&W model learns word embedding by modeling syntactic contexts of words but ignoring sentiment information. We develop a unified model $SSWE_u$ in this part, which captures the sentiment information of sentences as well as the syntactic contexts of words. This model is described in the Figure 1.

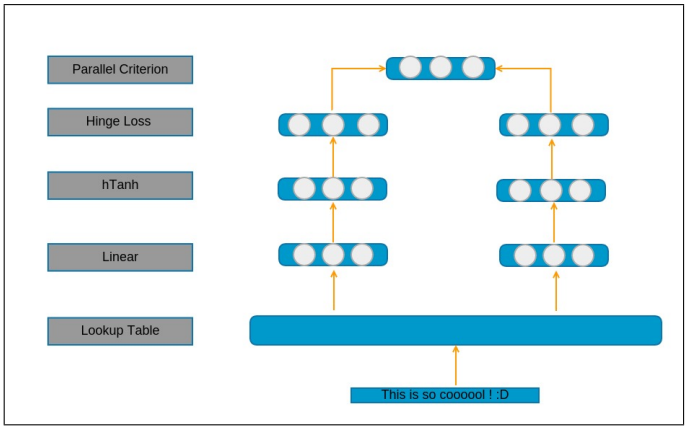

Fig. 1. *Diagrammatic Representation of neural network $SSWE_u$ for learning sentiment-specific word embedding*

Given an original (or corrupted) ngram and the sentiment polarity of a sentence as the in- put, $SSWE_u$ predicts a two-dimensional vector for each input unigram. The loss function

of SSWE$_u$ is the linear combination of t- wo hinge losses, as seen here -

$$loss_u(t, t^r) = \alpha \cdot loss_{cw}(t, t^r) + (1 - \alpha) \cdot loss_{cw}(t, t^r)$$

where loss$_{cw}(t, t^r)$ is the hinge loss for syntactic part as and loss$_u(t, t^r)$ is the hinge loss for sentiment part. The hyper-parameter α weighs the two parts. Empirically it was set to 0.5

C. Training Model

We train sentiment-specific word embedding from massive distant-supervised tweets collected with positive and negative emoticons. we introduce few new resources for pre-processing twitter data. Like an emoticon dictionary, which is prepared by labeling several emoticons listed on Wikipedia with their emotional state. For example, :) is labeled as positive whereas :=(is labeled as negative. We assign each emoticon a label from the following set of labels: epositive, enegative. (epositive denotes a positive emoticon and enegative, a negative one).

We pre-process all the tweets as follows: a) replace all the emoticons with a their sentiment polarity by looking up the emoticon dictionary, b) remove all the URLs, c) remove targets (e.g. @John), d) replace all negations (e.g. not, no, never, nt, cannot) by tag not, e) remove the hash from the hashtags and keep the content following the # symbol. f) remove the stopwords.

We used the ARK tokenizer to tokenize the data. The tokenizer takes the above preprocessed file as input and does the following. a) Replaces multiple spaces and tabs with single space b) Generates a new file containing tweets as tokens separated by space.

We train SSWE$_u$ by taking the derivative of the loss through back-propagation. We empirically set the window size as 2, the embedding length as 50. We learn embedding for unigrams with neural network and same parameter setting. The contexts of unigram are the surrounding unigrams.

IV. Experimentation and Results

We conducted experiments on the Twitter sentiment classification benchmark dataset in SemEval 2013. The training set consisted of 7,072 tweets and test set consisted of 1,316 tweets in all with the sentiment-wise distribution as shown in table 1. The tweets with neutral sentiment were removed, since the embedding file that we used for training our model did not have tweets with neutral sentiment.

	Positive	**Negative**	**Neutral**
Training	2642	994	3436
Testing	461	265	590

We used Support Vector Machines(SVM) for classification, using the standard library libsvm. The results are tabulated in the below confusion matrix :

	Positive	**Negative**
Positive	412	49
Negative	184	81

The overall accuracy obtained was **67.9**%. F-Score obtained for positive samples was considerably good at 81.81, whereas the F-Score obtained for negative samples was 41.44. The Overall F-Score being **61.63**. The F-Score obtained with respect to negative samples was observed to be quite low.

V. Conclusion

In this project, we propose learning continuous word representations as features for Twitter sentiment classification under a supervised learning framework.

We show that the word embedding learned by traditional neural networks are not effective enough for Twitter sentiment classification. These methods typically only model the context information of words so that they cannot distinguish words with similar context but opposite sentiment polarity (e.g. good and bad).

We learn sentiment-specific word embedding (SSWE) by integrating the sentiment information into the loss functions of three neural networks. We train SS WE with massive distant-supervised tweets select ed by positive and negative emoticons.

Our unified model combining syntactic context of words and sentiment information of sentences yields the best performance.

Acknowledgment

We would like to express profound thanks to our mentor - Ganesh Jawahar, who consistently attended to our doubts & guided us generously for the accomplishment of this project.

References

[1] Duyu Tang et al., "Learning Sentiment-Specific Word Embedding for Twitter Sentiment Classification." *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics.* June, 2014