Distributed Representation of Paragraphs Implementation

Sravani
201507501

Raksha
201507632

Narendra Babu
201507602

Nagarjuna
201550815

ABSTRACT

The key idea of distributed representation is to describe the data (Text, Image, Speech) with features on multiple, inter dependent layers, each describing the data at different levels of scale. When it comes to text, scale varies from words to documents. Commonly bag of words model is used to represent the text in fixed length feature vectors. Despite their popularity bag of words model has disadvantage of ignoring the word order and semantics. In this paper we are implementing the paragraph-to-vector [1], an unsupervised algorithm that learns fixed length feature representations from variable-length pieces of text (in our paper tweets are used). This algorithm represents tweets by a dense vectors which are further used for sentiment analysis. Its construction gives our algorithm the potential to overcome the weaknesses of bag of words models.

Keywords

Distributed representation; Bag of words; Paragraph to vector; Feature vectors;Sentiment Analysis

1. INTRODUCTION

Text classification is a general and important machine learning problem. Many machine learning algorithms require the input to be represented as a fixed-length feature vectors. So feature engineering has become a crucial step in developing good text classification systems because performance of learning algorithms depends on data representation.

Perhaps the most common fixed-length vector representation for texts is the bag of-words or bag-of-n-grams due to its simplicity, efficiency and often surprising accuracy. However, the bag-of-words (BOW) has many disadvantages. The word order is lost, and thus different sentences can have exactly the same representation, as long as the same words are used. Even though bag-of-n-grams considers the word order in short context, it suffers from data sparsity and high dimensionality.

Bag-of-words and bag-of-n-grams have very little sense about the semantics of the words or more formally the distances between the words. The above problem can be resolved when we can represent the word by its co-occurrence count with neighbouring words (context). Based on this , a new model ”Word2Vec” [2] was designed which generates a similar word vector for the words that occur in a similar context. It has two different neural models: Continuous bag of words(CBOW) model and Skip Gram model. The objective of CBOW is predicting the center word from sum of surrounding word vectors where as the objective of the skip-gram model is predicting the surrounding single words from center word.

The Word2Vec can be extended to represent sentences and paragraphs as vectors [3]. An application of this is discussed in the following sections.

2. PROBLEM

In the past decade, new forms of communication, such as micro blogging and text messaging have emerged and become ubiquitous. While there is no limit to the range of information conveyed by tweets and texts, often these short messages are used to share opinions and sentiments that people have about what is going on in the world around them. Tweets and texts are in short: a sentence or a headline rather than a document. The language used is very informal, with creative spelling and punctuation, misspellings, slang, new words, URLs, and genre-specific terminology and abbreviations, such as, RT for ”re-tweet” and hash tags, which are a type of tagging for twitter messages. In this paper, we propose a supervised learning algorithm to predict the sentiment of tweets. The system takes as input a tweet message. Given a tweet, classify whether it is of positive, negative, or neutral sentiment. For tweets conveying both a positive and negative sentiment whichever is the stronger sentiment should be chosen. Tweets in the positive and negative classes are subjective in nature. However, the neutral class consists of both subjective tweets which do not have any polarity as well as objective tweets.

3. METHOD

The first stage of the algorithm generates a vector representation of each tweet using the distributed bag of words which is a variation of the skip gram neural language model. The objective is to predict the words in a sample window from a paragraph or sentence given its id. The neural network is trained using trigrams and negative sample labels to refer to context of each tweet. The weights are learned using back propagation and stochastic gradient descent. Over 5 iterations , the error reduced from 0.4 to 0.06. The generated vectors are subsequently used to train an SVM and predict the sentiment class of a tweet (positive, negative or neutral). The SVM can be trained by using the parameters predicted from GridSearchCV and cross validation which gives accuracy for various values of C(the optimization parameter).

4. RESULTS

A fixed feature vector length of 10 was produced as the output of the neural network for each tweet. These were subsequently fed into a linear SVC to predict the sentiment class which, for C values in the range 100 to 500 , gave an accuracy ranging from 39

5. REFERENCES

[1] Quoc Le and Tomas Mikolov. Distributed Representations of Sentences and Documents.

[2] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. In Proceedings of Workshop at ICLR, 2013.

[3] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed Representations of Words and Phrases and their Compositionality. In Proceedings of NIPS, 2013.