2 STEP SENTIMENT POLARITY CLASSIFICATION ON IMDB MOVIE REVIEW DATASET

Francesco Ferrini [224010]
University of Trento
Via Sommarive 9, 38123, Povo, Trento, TN
francesco.ferrini@studenti.unitn.it

1. Abstract

In this project are shown several natural language processing techniques useful for performing sentiment analysis on IMDB movie reviews dataset. The pipeline is made by two steps: the first one is the elimination of objective sentences in each review, while the second one is the classification into positive and negative reviews. For the classification task I used the Bag of Words (BOW) and Term Frequency - Inverse Document Frequency (TF-IDF) followed by different machine learning approaches including Logistic regression, Multinomial Naive Bayes, Linear Support Vector machines, random forest and Multi Layer Perceptron.
It's showed that adding the subjectivity step, it is possible to improve the accuracy of sentiment polarity classification.

2. Introduction

Sentiment analysis is a very used task in Natural Language Processing where, given some text, the objective is to determine the polarity of that text. The polarity of a movie review coming from the dataset, in our case, can be positive (+) or negative (-).

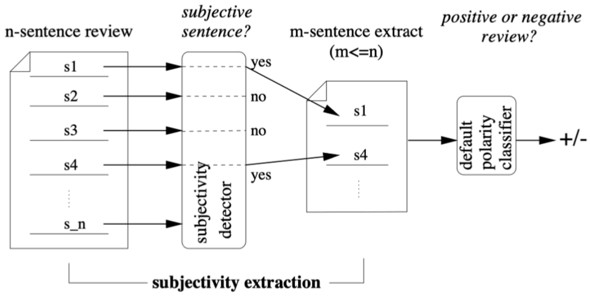

Figure 1

In the proposed approach it's added a task that is the elimination of the objective sentences in each movie review.

The method analyzed in this paper is described in Figure 1 [1].

3. Dataset

It's used a dataset from NLTK that is called movie review dataset. It's a collection of 2000 reviews (1000 positive and 1000 negative) coming from IMDB. Each review is composed by many sentences and some of them could not be useful for the classification and so have to be removed.

4. Framework

A. Pre-processing

To convert a cleaned sequence of words to numerical feature vectors are used the following methods:

- Bag of Words (BOW)
  Bag of words is probably the simplest way to numerically represent some text. A bag of words is a representation of text that describes the occurrence of words within a document. It's a method to just keep track of word counts and disregard the grammatical details and the word order.

- Term Frequency — Inverse Document Frequency (TF-IDF)
  Another possibility is to quantify a word in documents, generally computing a weight to each word which signifies the importance of the word in the document and corpus. This method is a widely used technique in Information Retrieval and Text Mining.

Once the feature vectors are finally extracted several binary classifiers are used to learn the sentiments.

B. Subjectivity detection approach

While sentiment classification and subjectivity detection [2] are closely related to each other, it has been reported that separating subjective and objective instances from text is more difficult than sentiment classification, and the improvement of subjectivity detection can benefit the latter as well.
For this purpose are tried and compared 2 different approaches:

- TextBlob
  This is a Python (2 and 3) library for processing textual data. It provides a simple API for diving into common natural language processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, and more.
  It can also return the subjectivity of a sentence in a range between 0 and 1 where 1 means that the sentence taken into consideration is very subjective.

- LSTM
  In this approach [3] it's used another dataset (called subjectivity dataset). It contains 5000 subjective and 5000 objective processed sentences.
  For this method it's also used a context independent embedding model called Word2Vec that is a representation for a word regardless of where the word occurs in a sentence or the different meanings it might has. For this I used pre-trained vectors trained on a subset of the Google News Dataset.
  Figure 2 describes the architecture of the network and the output is a value that explains how much a sentence is subjective.

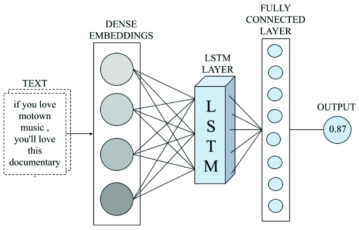

Figure 2

C. LSTM implementation details

For the implementation of this model, used to carry out the second subjectivity approach, it's used as embedding method Word2Vec. It returns only one numeric representation for a word regardless of where the word occurs in a sentence or the different meanings it might has.
The pertained word embeddings are used as an embedding layer. This is used as connection between the input and the LSTM layer which is used for learning long distance dependencies between word sequences in short text. The output of the LSTM is connected to a fully connected layer with softmax classifier that outputs the probability of a sentence to be subjective.
After training, LSTM has an accuracy of 0.85.

D. Classifiers Algorithm

The classier algorithms used for the classification into positive/negative review are:

- Logistic Regression
- Naive Bayes
- SVM
- Random Forest
- MLP

5. Results

In the tables are shown the results before the removal of objective sentences and after the removal with the 2 different approaches. Since are tried different values in the subjectivity range task, are reported only the best results. The accuracy reported is F1-Score.

A. Results without subjectivity task

	BOW	TF-IDF
Logistic Regression	0,842	0,822
Naive Bayes	0,814	0,808
SVM	0,833	0,86
Random Forest	0,803	0,786
MLP	0,858	0,852

B. Results using TextBlob for subjectivity detection

	BOW	TF-IDF
Logistic Regression	0,838	0,838
Naive Bayes	0,825	0,831
SVM	0,822	0,855
Random Forest	0,774	0,784
MLP	0,843	0,853

C. Results using LSTM model for subjectivity detection

	BOW	TF-IDF
Logistic Regression	0,832	**0,844**
Naive Bayes	**0,834**	0,818
SVM	0,827	**0,86**
Random Forest	**0,812**	0,806
MLP	0,861	**0,863**

6. Comments

In the results section are highlighted the best F1 score found for the classification task. The LSTM approach gives better results than TextBlob and also with respect of the no usage of any subjectivity detector. This demonstrate that in some way the objective sentences are not useful for the classification task and that removing them can improve the accuracy of a binary polarity classification.

6. References

[1] Bo Pang and Lillian Lee, “A Sentimental Education: Sentiment Analysis Using Subjectivity Summarization Based on Minimum Cuts”, Proceedings of the 42nd ACL, pp. 271—278, 2004

[2] Bing Liu, “Sentiment Analysis and Subjectivity”, Handbook of Natural Language Processing, Second Edition, (editors: N. Indurkhya and F. J. Damerau), 2010

[3] Ritika Nandi, Geetha Maiya, Priya Kamath, Shashank Shekhar, “An Empirical Evaluation of Word Embedding Models for Subjectivity Analysis Tasks”, 2021 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT)