CS410 Technology Review:

Overview of NLTK toolkits on sentiment analysis

Yijing Yang(yijingy2)

Overview

Analyzing public opinion can offer us useful information. Sentiment analysis on social networks such as Twitter has become a useful tool for learning about users' views and has a wide variety of applications. Deep learning models have been proven in recent years to offer a promising solution to NLP problems.[1]

This paper reviews the Natural Language Toolkit (NLTK) [2] that have employed deep learning to solve sentiment analysis problems. First, the preprocessing steps required to extract features from the dataset. Then, an algorithm is trained for each topic to estimate the sentiment. Finally, the method estimates the sentiment of a sentence based on the best topic related algorithm results.

Introduction

Natural language is a language that is used for everyday communication by humans. Natural languages, unlike artificial languages such as computer languages and mathematical notations, have evolved as they pass from generation to generation, and are hard to define with explicit rules. Natural language processing(NLP) is a model for studying language ability and language application. A computer algorithm is built to implement such a language model, and it is perfected, evaluated, and finally used to design various practical systems.

Sentiment analysis refers to the use of natural language processing, text analysis, computational linguistics, and biometrics to systematically identify, extract, quantify, and study subjective preferences and affective states. In general, sentiment analysis aims to determine a speaker's, writer's, or other subject's attitude toward a topic, or the overall contextual polarity or emotional reaction to a document, interaction, or event.

The Natural Language Toolkit(NLTK) is a popular platform for developing Python programs that interact with human language data. It was developed by Steven Bird and Edward Loper in the Department of Computer and Information Science at the University of Pennsylvania [3]. NLTK is intended to support research and teaching in NLP or closely related areas, including empirical linguistics, cognitive science, artificial intelligence, information retrieval, and machine learning. It offers simple interfaces to over 50 corpora and lexical resources, including WordNet, as well as a suite of text processing libraries for classification, tokenization, and stemming, tagging, parsing, and semantic reasoning.

Data Preprocessing

Several preprocessing steps rely on the NLTK libraries[4]. Furthermore, the algorithm Naive Bayes (NB) also could be implemented by NLTK.

The preprocessing steps aim to begin the feature extraction process and extract bags of words from the dataset. One of the primary goals is to minimize the total number of features extracted. Indeed, feature reduction is critical for improving prediction accuracy in both topic modeling and sentiment analysis. The samples are represented by features, and the more the algorithm is trained on a certain feature, the more accurate the results will be. As a result, if two features are similar, it is easier to merge them as one unique feature. Furthermore, if a feature is irrelevant to the analysis, it can be eliminated from the word bag[5].

- **Tokenize:** Tokenization is almost implicit since the English language is already segmented. Because each word is separated by a space, the token can be generated by dividing the text on each space. The tokenization applied for this project also include other functionalities, such as separated punctuation tokens from the word. NLTK word_tokenize method can be used for tokenizing its samples.

- **Detect POS tags:** Part of speech may have two uses for data analysis. First, it may be used to clarify the meaning of a term. Even if a reader understands that "like" has two different meanings in the lines "I like that" and "I am not like you," the algorithm will treat them as the same when computing the bag of words. The second application of POS tags is to categorize words and treat them differently depending on which type they correspond to[6]. The detection of the samples' POS tags can rely on the NLTK method pos_tag.

- **Lemmatize:** When processing samples, "word" and "words" are recognized as distinct features. As a result, the unigrams can be lemmatized to optimize the features reduction process. This stage mainly allows for the removal of plurals and conjugations. The lemmatization process can be based on the WordNet implementation included with the NLTK distribution.

Sentiment Analysis

Given that the data may be related to several subjects and that a sentiment analysis algorithm has been trained for each of these topics, a technique for using the results of all of these algorithms must be defined. The phrase is first preprocessed to estimate sentiment, and then features from the bag of words are extracted. The topic probability distribution is then calculated. The algorithm trained for this specific subject is used to measure sentiment for any topic whose probability exceeds a threshold. The estimate with the highest probability is retained as the final estimate.

Conclusions

This paper aims to detail the preprocessing steps that have to be applied to extract bags of words from the samples, as well as how NTLK could be used on sentiment analysis. This approach is based on the assumption that the complexity of an analysis may be lowered by focusing the algorithm on a limited range of topics. Because the algorithms must deal with fewer vocabulary, their estimation can be more accurate. The procedure can be improved. The paper concentrated on utilizing the findings of the topic modeling method's default parameter. More research on topic extraction may allow for more distinct subjects, which are likely to result in more accurate estimation.

References

[1] N. C. Dang, M. N. Moreno-García, and F. De la Prieta, "Sentiment Analysis Based on Deep Learning: A Comparative Study," *Electronics*, vol. 9, no. 3, p. 483, Mar. 2020, doi: 10.3390/electronics9030483.

[2] "NLTK :: Natural Language Toolkit." https://www.nltk.org/

[3] J. Yao, "Automated Sentiment Analysis of Text Data with NLTK," *J. Phys.: Conf. Ser.*, vol. 1187, no. 5, p. 052020, Apr. 2019, doi: 10.1088/1742-6596/1187/5/052020.

[4] "Natural Language Processing with Python [Book]." https://www.oreilly.com/library/view/natural-language-processing/9780596803346.

[5] P. Ficamos and Y. Liu, "A Topic based Approach for Sentiment Analysis on Twitter Data," *ijacsa*, vol. 7, no. 12, 2016, doi: 10.14569/IJACSA.2016.071226.

[6] P. Turney, "Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews," in *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, Pennsylvania, USA, Jul. 2002, pp. 417–424. doi: 10.3115/1073083.1073153.