CS 410 Technology Review - Review on recent text sentiment classification techniques for social media comments

Chuanyue Shen (NetID: cs11)

Introduction

Social media now is a necessary part of our daily life with digitalization. The posts, comments, and responses on social media, which are generated heavily everyday and presented as unstructured, have more or less impact in a positive or negative way. Understanding and tagging unstructured text data offers businesses numerous benefits such as sorting data at scale, real-time analysis, and consistent criteria [1].

Sentiment analysis is the process of analyzing a piece of text and determining the opinions or attitude towards the text, which recently becomes one of the most popular applications of Natural Language Processing with use of machine learning. Sentiment can be positive, neutral, or negative. The most frequency quantification of sentiment is by polarity, i.e. a positive or negative value. The overall sentiment is inferred from the sign of the polarity score [2]. Type of sentiment analysis includes [1]

- fine-grained sentiment analysis, which expands the polarity to five categories, i.e. very positive, positive, neutral, negative, and very negative;
- emotion detection, which aims at detecting emotions, such as happiness, frustration, anger, sadness, etc;
- aspect-based sentiment analysis, which identifies the particular features that are sensed;
- and multilingual sentiment analysis, which is the most challenging and requires several preprocessing and resources.

Due to the nature of categorizing text into different groups, sentiment analysis can be treated as a classification problem. With the development of machine learning, researchers take advantage of machine learning techniques to form an automatic sentiment analysis system. This review will be focusing on recent machine learning techniques for text sentiment classification of Reddit comments and Tweets.

Twitter

Verbal aggression can be often seen on social media comments, which is one of common issues for developing a healthy internet. J. Chen, S. Yan and K. Wong [3] proposed an efficient CNN-based deep learning model for verbal aggression detection on Twitter comments. In the deep learning model, the authors set 128 rectangular-shaped filters of 10 × 2 on the convolutional layer, used mean pooling layer, and two-layered multilayer

perceptron following the pooling layer. The authors adopted a word embedding method that treats word embedding on-the-fly [3] [4]. They also applied two-dimensional (2D) TF-IDF matrix as document-level features. Their study compared the developed CNN model to other popular methods, such as support vector machine (SVM), logistic regression with stochastic gradient decent, and long short-term memory (LSTM). Holdout validation method was used for performance benchmarking. In the experiment, they used 60% of original data for training, and 40% of data for testing. Performance metrics are accuracy and area under the curve (AUC) of the receiver operating characteristics (ROC) [3].

The results show that the CNN model with 2D TF-IDF outperforms other models, achieving 0.92 accuracy, 0.98 micro-AUC, and 0.97 macro-AUC, and the second better model is LSTM with 2D TF-IDF.

Reddit

Sarcasm is a type of sentiment where the intended meaning is opposite to the way it is communicated. People can easily recognize sarcastic languages in communication based on the context, common-sense knowledge, and the way that words are uttered. However, without knowing these cues, it is difficult to unveil the languages that are sarcastic wits.

K. Clarkson and M. Reza [5] explored several machine learning models for classifying sarcastic comments on Reddit. Instead of traditional word representation method "bag-of-n-grams", they make use of word embedding for the comment representation. In their work, they used a corpus of self-labelled Reddit comments, created by Kaggle user Dan-Ofer. The authors represent Reddit comments as high-dimensional vectors using genism library's [7] document-to-vector (doc2vec) with distributed bag-of-words (DBOW). The doc2vec is built on word2vec with an additional consideration of the same word appearing elsewhere within the same document. With that, doc2vec was expected to capture the semantic meaning of words given the comments they reside in. These vector representations were then input into classification models to predict sarcastic comments. Multiple classification models were used and compared in the paper, including logistic regression with L1 penalty, logistic regression with L2 penalty, K-nearest-neighbors (KNN), and SVM using an RBF change of basis. The model training was done on 750,000 comments from three subreddits. The average validation error of 5-fold cross validation was used as the model evaluation metric.

The comparison of results finds that logistic regression and SVM have lower average validation errors than KNN in overall, and the doc2vec representation with the approaches mentioned above was no better than bag-of-n-grams representation with Naive Bays. The results of the study suggest that the used doc2vec word embedding representation may not be adequate for classifying Reddit sarcasm comments.

Conclusion

The popular text sentiment classification methods include logistic regression, naive bayes, support vector machine, k-nearest-neighbors, long short-term memory, convolutional neural network, etc. The most attractive method among these is CNN that has the most potential to learn deep and achieve desired performance. In addition, applying TF-IDF as document-level features can improve the model performance. In terms of vector representation method, document-to-vector (doc2vec) was expected to capture the semantic meaning of words given the comments they reside in and thus outcompete bag-of-n-grams method. However, in the case study of Reddit sarcasm classification, doc2vec failed the expectation, where further studies are required to give a more thorough comparison.

References

[1] *Monkeylearn.com*, 2020. [Online]. Available: https://monkeylearn.com/sentiment-analysis/.

[2] D. Sarkar, "Emotion and Sentiment Analysis: A Practitioner's Guide to NLP - KDnuggets", *KDnuggets*, 2020. [Online]. Available: https://www.kdnuggets.com/2018/08/emotion-sentiment-analysis-practitioners-guide-nlp-5.html.

[3] J. Chen, S. Yan and K. Wong, "Verbal aggression detection on Twitter comments: convolutional neural network for short-text sentiment analysis", *Neural Computing and Applications*, vol. 32, no. 15, pp. 10809-10818, 2018. Available: 10.1007/s00521-018-3442-0.

[4] Y. Gal and Z. Ghahramani, "A theoretically grounded application of dropout in recurrent neural networks", *In Advances in neural information processing systems*, pp. 1019-1027, 2016.

[5] K. Clarkson and M. Reza, "Was That Sarcasm? – A Survey of Machine Learning Models for Classifying Sarcastic Comments on Reddit Using Word Embeddings", 2020.

[6]"Gensim: topic modelling for humans", *Radimrehurek.com*, 2020. [Online]. Available: https://radimrehurek.com/gensim/.