Aspect Based Sentiment Analysis: Approaches and Methods
CS 583 Project Report

Mohit Adwani and Shubadra Govindan

Abstract

With the rise of social media activity, sentiment analysis is turning into a revenue generator. There are multiple platforms and forums that allow people to express their opinion and take on any topic. As most of these discussions revolve around comparisons of various products or topics, considering the context of the opinion or sentiment is essential.

The goal of this project to unearth the sentiment of the text with respect to a certain context or target term. The text body under consideration could have multiple target values, each having a different sentiment than the other. This project deduces the sentiment scores associated with the aspect term, that is the target term. A positive text is assigned a score of positive one, a neutral text is given a zero while a negative text is given a score of negative one.

To identify the sentiment towards the aspect term, we have calculated the weight of every word in the sentence (after cleaning and pre-processing and used the weight values to train and predict our models.
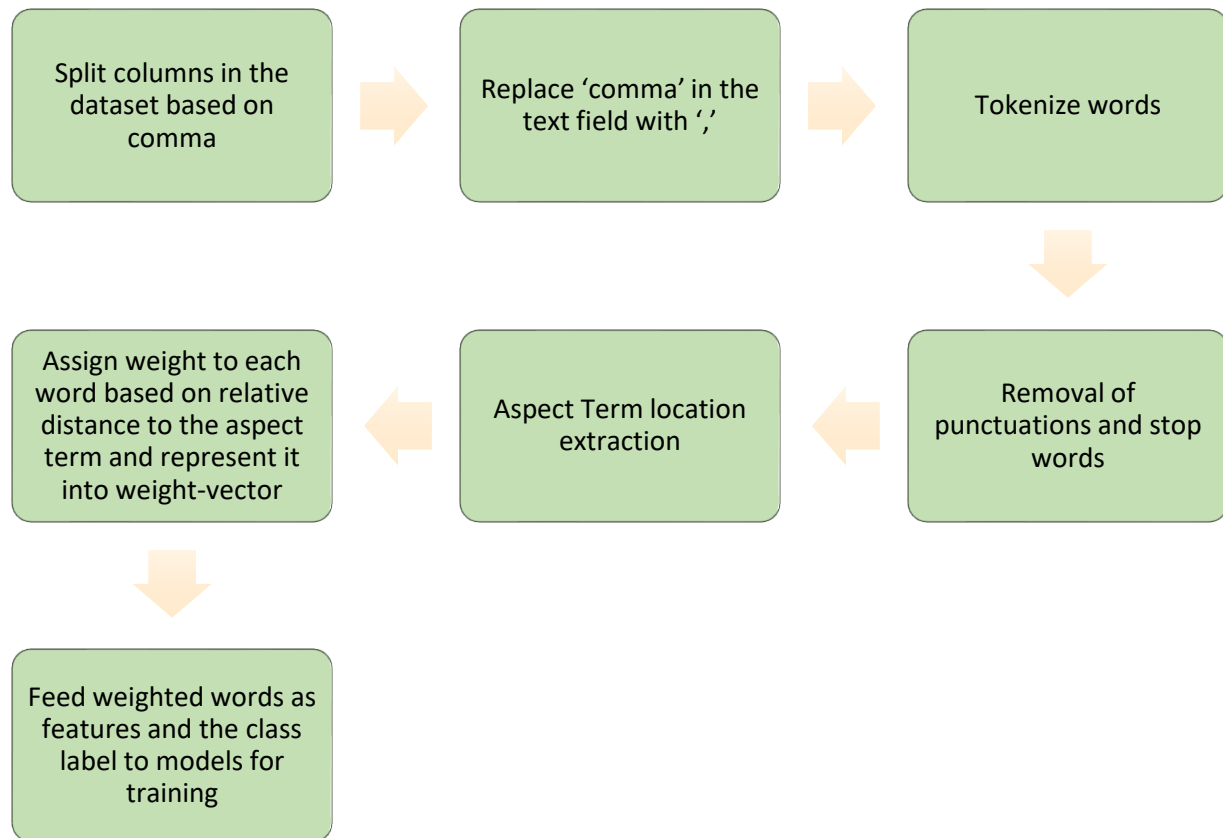
Introduction

Basic sentiment analysis provides the overall sentiment of the text in question. When there are multiple objects or context-based discussions, knowing the overall sentiment is not useful. For example, let us consider a sentence about coffee – 'the coffee at the quaint café was spectacular, although there was nothing exceptional about the lemon tart'. Given a sentence like this, the system often calculates the score of the sentiment based on overall positioning of key words. This project aims to unearth the sentiment towards a target word, that is the aspect (coffee in this case).

The dataset used for this project consists of various examples of text with example IDs associated with each of them, an aspect term for that sentence, a sentiment class label (1, 0 or -1) depending on the sentiment with respect to the aspect term, and the location of the aspect term. After data cleaning and pre-processing steps, the words are fit to various machine learning models. The accuracy score and f1 score are used to determine the efficiency of the model. Basic sentiment analysis was first carried out on the datasets for exploratory and learning purposes where the aspect term was not considered. Post this, the sentiment score towards an aspect term was determined. The models used were trained on the weighted word vectors as the inputs and the associated label for that text as the output. Models such as Neural Nets, Linear SVM, Naïve Bayes, Gaussian Naïve Bayes, Adaboost and Random Forests. For the basic sentiment analysis, we found the Linear SVM to work best while Neural Nets was found to perform better for the aspect - based sentiment analysis.

Techniques

Overall process flow of the model is depicted in the workflow chart below:

```
┌──────────────────┐      ┌──────────────────┐      ┌──────────────────┐
│ Split columns in │  →   │ Replace 'comma'  │  →   │ Tokenize words   │
│ the dataset based│      │ in the text field│      │                  │
│ on comma         │      │ with ','         │      │                  │
└──────────────────┘      └──────────────────┘      └──────────────────┘
                                                              ↓
┌──────────────────┐      ┌──────────────────┐      ┌──────────────────┐
│ Assign weight to │  ←   │ Aspect Term      │  ←   │ Removal of       │
│ each word based  │      │ location         │      │ punctuations and │
│ on relative      │      │ extraction       │      │ stop words       │
│ distance to the  │      │                  │      │                  │
│ aspect term and  │      │                  │      │                  │
│ represent it into│      │                  │      │                  │
│ weight-vector    │      │                  │      │                  │
└──────────────────┘      └──────────────────┘      └──────────────────┘
        ↓
┌──────────────────┐
│ Feed weighted    │
│ words as features│
│ and the class    │
│ label to models  │
│ for training     │
└──────────────────┘
```

The weight for each word remaining in the text after pre-processing was calculated based on the distance of the word from the aspect term. Weight is equal of the inverse of the distance of the word from the aspect term. The vocabulary containing the weighted values for the word (calculated weight if the word occurs in the sentence, else 0) are fed as features to the model to learn. The corresponding class labels are also used by the model to learn. Cross validation was performed on the data during training on models other than neural networks.

The classification methods tried include Neural Nets, Linear SVM, Naïve Bayes, Gaussian Naïve Bayes, Adaboost and Random Forests. Out of these models, we found Neural Nets to perform the best, followed by Linear SVM. In both the models, hyperparameter tuning was carried out to identify the best fit for the dataset and the same parameters were applied on the test dataset.

Evaluation (Results from Different Methods)

We applied basic sentiment analysis without considering the aspect term to get a base line model for comparing with aspect-based sentiment analysis models.
Few models that we tried were Naïve Bayes, Adaboost, SVM. SVM gave the best accuracy for basic sentiment analysis and hence we chose that model to train for aspect-based sentiment

analysis. Then we performed neural network model which improved the accuracy and F1-score. The evaluation was done using 10-fold cross validation.

Basic Sentiment Analysis:

Classifier: Linear SVM with <u>count vectorizer</u> (No aspect term considered) (with 70, 30 partition and 10-fold cross validation)

Data 1 (2313, 2969)

Target Class	Precision	Recall	F-score
0	73%	75%	74%
1	53%	44%	48%
-1	76%	81%	79%
Average	71%	72%	71%
Overall Accuracy	**71.61%**		

Data 2(3602, 3614)

Target Class	Precision	Recall	F-score
0	60%	61%	61%
1	48%	42%	45%
-1	83%	86%	84%
Average	72%	73%	72%
Overall Accuracy	**72.52%**		

Classifier: Linear SVM with <u>weight vector</u> (Aspect term considered; 10-fold cross validation on data)

Data 1 (2313, 2969)

Target Class	Precision	Recall	F-score
0	78%	70%	74%
1	39%	59%	47%
-1	83%	78%	80%
Average	75%	72%	73%
Overall Accuracy	**72.30%**		

Data 2(3602, 3614)

Target Class	Precision	Recall	F-score
0	49%	60%	54%
1	31%	52%	39%
-1	90%	76%	82%
Average	76%	70%	72%
Overall Accuracy	**70.32%**		

Classifier: Feed forward neural network with weight vector (10-fold cross validation)

Data 1 (2311, 2961)

Target Class	Precision	Recall	F-score
0	73%	77%	75%
1	63%	43%	51%
-1	77%	84%	81%
Average	73%	73%	73%
Overall Accuracy	**73.47%**		

Data 2(3602, 3614)

Target Class	Precision	Recall	F-score
0	65%	62%	64%
1	60%	37%	45%
-1	81%	92%	86%
Average	74%	76%	74%
Overall Accuracy	**75.62%**		

Conclusion

In this project, we performed aspect-based sentiment analysis on two datasets (Restaurant and laptop review datasets) using weight vector as the features, calculated based on the word distance from the aspect term. We pre-processed the data by removing stop words and punctuation. We applied machine learning models, linear SVM and feed forward neural network, and found that neural network model had a better overall accuracy and F1 score than SVM.

Our models worked best on data instances with negative class and poor on data instances with positive class. For future work, the model should be trained on more data and a deeper neural network model like LSTM RNN should be used for better performance.

References

Boiy, E. and Moens**,** MF. 2009. A machine learning approach to sentiment analysis in multilingual Web texts. Inf Retrieval (2009) 12: 526.

Zhang L., Liu B. 2016. Sentiment Analysis and Opinion Mining. In: Sammut C., Webb G. (eds) Encyclopedia of Machine Learning and Data Mining. Springer, Boston, MA