Aspect Based Sentiment Analysis - Domain Adaptation

ONLP Final Project

Rotem Yagil
ID: 026515973
M.Sc. Computer Science Student
The Open University of Israel
ONLP Course
rotem.yagil@gmail.com

Maor Eitan
ID: 308047943
M.Sc. Computer Science Student
The Open University of Israel
ONLP Course
maor.eitan.me@gmail.com

September 2019

Abstract

This paper describes the Aspect Based Sentiment Analysis (ABSA) SemEval-2016 Task 5 subtask 1 [1], the ways we tried to solve the task and how we improved our solution in order to make it work better on dataset from different domain (a domain adaptation theory which was suggested by Caroline Brun and Vassilina Nikoulina from Naver Labs Europe [2]).

1 Introduction

The goal of the ABSA task is to detect opinions from given user input, classify them into aspects and decide the sentiment of the opinion: **positive, negative or neutral**.

Each Aspect contains two parts: entity and attribute. In our project, we focused on restaurants reviews, so there are 6 entities: **restaurant, food, drinks, ambience, service, location** and 5 attributes: **general, prices, quality, style options, miscellaneous**.

The combination of these 6 entities and 5 attributes creates 30 different aspects that the restaurants reviews have to be classified to and each aspect needs to be classified as well to different 3 sentiments, so eventually we get 90 categories that the classifier needs to decide for each review.

In addition to that goal, we wanted to adapt our classifier to work better on datasets from different domains (data from sources that are different from the source of the training dataset) and maintain its robustness, because testing is usually made on data from the same domain as training dataset and performance might downgrade on real world.

2 Datasets

For training and testing the model, we used the **SemEval-2016 ABSA Restaurants reviews subtask 1** dataset in english, that was annotated according to the aspects and sentiments described above.

For testing the model on different domain and check its performance on real world situation, we used the **Foursquare restaurants reviews** dataset that was annotated by Caroline Brun and Vassilina Nikoulina, Naver Labs Europe, according to the same annotation guidelines above.

Example of annotated dataset row, with review (as "text"), opinions, aspects (as "category", ENTITY#ATTRIBUTE) and sentiments (as "polarity"):

<text>Judging from previous posts this used to be a good place, but not any longer.</text>

<Opinions>

<Opinion target="place" category="RESTAURANT#GENERAL" polarity="negative" from="51" to="56"/>

</Opinions>

3 ABSA System Models (See Figure 1)

Our ABSA system contains two models: baseline model and deep learning model, which helps the baseline model to predict better results on different domain datasets.

3.1 Baseline Model

The ABSA system first train the baseline model on SemEval dataset. In order to make the model prediction more focused, the aspects list is minimized from 30 possibilities to the most common aspect in the training dataset. Every review is tokenized, tagged to part of speech and stop words are filtered out, then a pandas data frame (table) is created from the reviews. Similarly, the testing process is also contains tokenization, POS tagging, filtering stop words and creating data frame.

Then, the model is fitted with the reviews (X vector) and the corresponding aspects (Y vector) for each sentiment separately (1 if the aspect sentiment is equal to the current sentiment, 0 otherwise).

The testing process is on two datasets, one from the same domain (SemEval) and the other from different domain (Foursquare). The model predicts (from the testing reviews X vector) the aspects classification for the current sentiment (Y vector) and then the evaluation metrics, accuracy, precision,recall, f1 and classification report, are calculated on the predicted Y vector vs. the gold Y vector.

Eventually, the predicted Y vector of each sentiment is converted to aspects, described by words, for writing the results to CSV file.

3.2 Deep Learning Model

The ABSA system runs secondly, in the same way as described in section 3.1, but this time in the training process, a second model is also trained with the training dataset.

The deep learning model uses keras, a deep learning library, and keras model, Sequential, which allows to stack layers of neural networks. The model extracts all the aspects and sentiments tuples, for example **(RESTAURANT#GENERAL, positive)**, from the training dataset and adds 4 fully connected layers (Dense) and the output layer size is the number of tuples.

Training dataset reviews are encoded in vectors (word embedding) using bag of words technique and a reviews data frame is created. The aspects and sentiments tuples are also get encoded and model is fitted with the encoded vectors.

The testing process uses Spacy to filter stop words, punctuation and tag the words from the reviews (SemEval and Foursquare testing datasets) to POS. A testing data frame is created from the encoded vectors created from the words in the reviews and finally, the model predict what is the probability of each aspect and sentiment tuple to exist in the reviews.

3.3 Domain Adaptation

One of our project goal was to improve the ABSA system robustness when the testing data is from different domain, different dataset which the model was trained from.

In the first round, only the baseline model predicts the classification to aspects and as a result, many of the reviews from the different domain (Foursquare dataset) remain without classification to aspects and sentiments. The conclusion is that the baseline model works well on reviews from the same domain, but fails many times on reviews from different domain.

In the second round, both models, baseline and deep learning, are predicting the classification and the final result is bitwise "or" between the two predicting vectors, so both models complete each other. As a result, this time only a negligible number of reviews from the different domain remain unclassified, so the conclusion is that the state of the art deep learning model empower the system robustness and makes it more domain adaptable.

4 Evaluation

In order to evaluate our ABSA system, we used sklearn metrics library, with accuracy, precision, recall, f1 and classification report for each sentiment separately (see table 1 example).

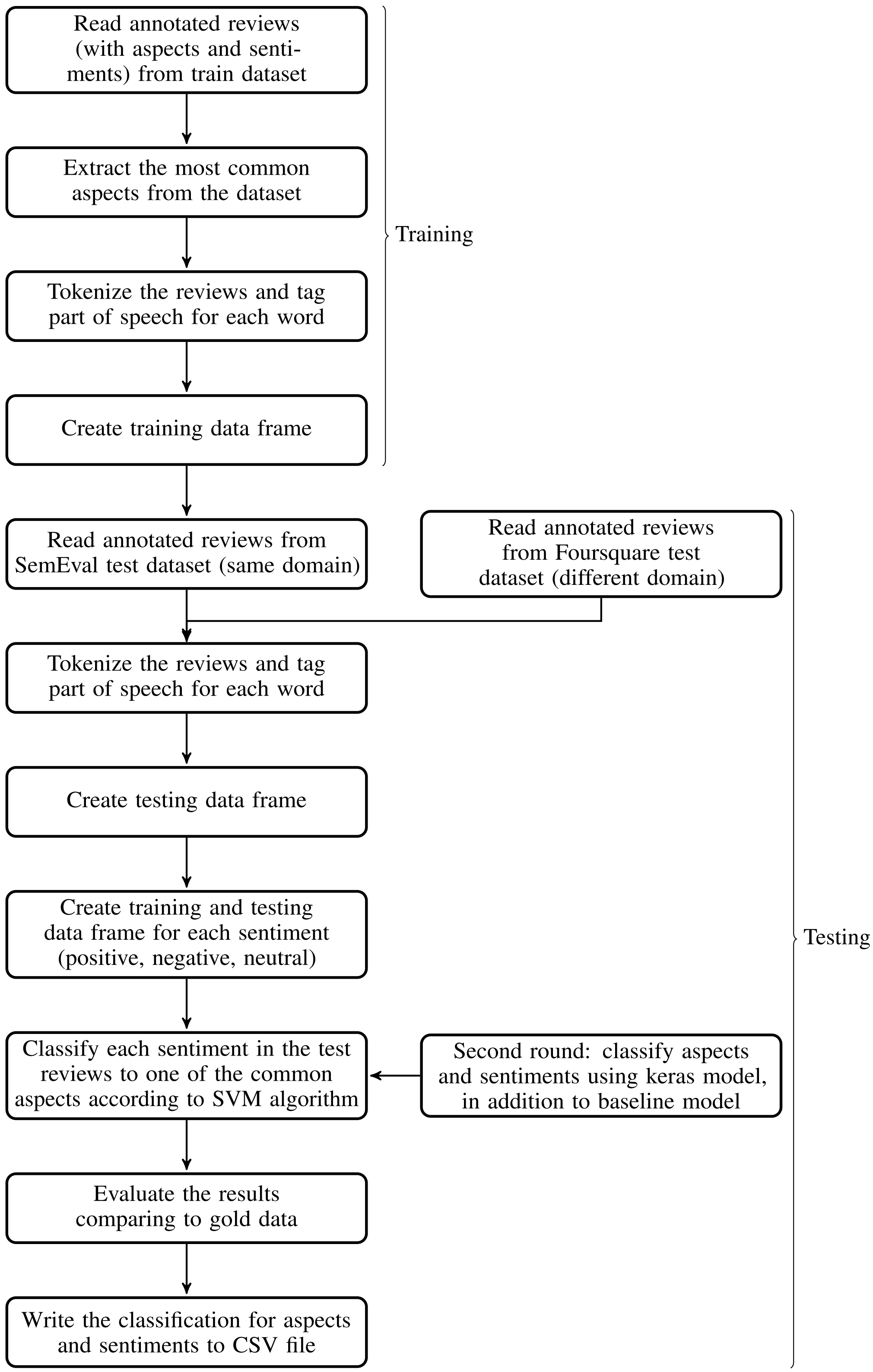

Figure 1: Absa system workflow

Table 1: SemEval Positive Classification Report - First Round

	precision	recall	f1-score	support
AMBIENCE#GENERAL	0.91	0.94	0.93	33
DRINKS#PRICES	0	0	0	0
DRINKS#QUALITY	1	1	1	15
DRINKS#STYLE_OPTIONS	0.91	1	0.95	10
FOOD#PRICES	0.8	0.67	0.73	6
FOOD#QUALITY	0.92	0.92	0.92	72
FOOD#STYLE_OPTIONS	0.56	0.93	0.7	15
LOCATION#GENERAL	0.83	0.71	0.77	7
RESTAURANT#GENERAL	0.88	0.93	0.9	54
RESTAURANT#MISCELLANEOUS	0.77	0.77	0.77	13
RESTAURANT#PRICES	0.8	0.67	0.73	6
SERVICE#GENERAL	0.81	0.93	0.87	42
micro avg	0.85	0.91	0.88	273
macro avg	0.77	0.79	0.77	273
weighted avg	0.86	0.91	0.88	273
samples avg	0.77	0.81	0.78	273

5 Conclusion

The ABSA task was not easy for us, especially to make it domain adaptable. We tried to use pre-trained word embedding techniques such as word2vec (Google), GloVe (Stanford), in order to enrich the corpus, but we did not get any improvements. In addition, we tried to train the model with reviews from different domain, so it could predict better result in testing process, but we observed no improvements either.

We would like to continue our research in order to find better approaches and state of the art techniques to improve our results.

References

[1] SemEval-2016 Task 5: Aspect Based Sentiment Analysis.

[2] Caroline Brun and Vassilina Nikoulina. Aspect Based Sentiment Analysis into the Wild.