The 2017 Conference on Computational Linguistics and Speech Processing
ROCLING 2017, pp. 308-322

Multi-Domain Aspect Extraction using Support Vector Machines

Nadheesh Jihan, Yasas Senarath, Dulanjaya Tennekoon, Mithila Wickramarathne, and Surangika Ranathunga

Department of Computer Science and Engineering,

University of Moratuwa, Katubedda 10400, Sri Lanka

{nadheeshj.13, wayasas.13, dulanjayatennekoon.13, mithwick.13, surangika}@cse.mrt.ac.lk

Abstract

This paper describes a system to extract aspect categories for the task of aspect based sentiment analysis. This system can extract both implicit and explicit aspects. We propose a one-vs-rest Support Vector Machine (SVM) classifier preceded by a state of the art preprocessing pipeline. We present the use of mean embeddings as a feature along with two other new features to significantly improve the accuracy of the SVM classifier. This solution is extensible to customer reviews in different domains. Our results outperform the best recorded F1 score in the SemEval-2016 Task 5

dataset consisting of customer reviews from restaurant and laptop domains.

Keywords: Aspect Extraction, Sentiment Analysis, Supervised Machine Learning, SVM, Preprocessing, Mean Embedding

1. Introduction

The Internet has become the means of expressing opinion and view of consumers of products and services. Information contained in these reviews is of great value to other consumers as well as the companies that own those products and services. However, consumer reviews are often unstructured and noisy. Manual analysis of this huge amount of data for information is impossible. Automatic sentiment analysis of customer reviews has therefore, become a priority for the research community in the recent years.

Conventional sentiment analysis of text focuses on the opinion of the entire text or the

sentence. In the case of consumer reviews, it has been observed that customers often talk about multiple aspects of an entity and express an opinion on each aspect separately rather than expressing opinion towards the entity as a whole [1]. Aspect Based Sentiment Analysis (ABSA) has emerged to tackle this issue. The goal of Aspect Based Sentiment Analysis is to identify aspects present in the text, and the opinion expressed for each aspect [2].

One of the most crucial tasks of ABSA is to extract aspects from the review text. The state of the art systems have trouble in working with multiple domains, detecting multiple aspects in a single sentence, handling a large number of hierarchical aspects and detecting implicit aspects where the aspect is to be inferred from the context [3]. The objective of our research is to develop new techniques that would be able to perform aspect extraction from customer reviews with high accuracy, across multiple domains.

A one-vs-rest Support Vector Machine (SVM) classifier and a list of carefully selected features are at the core of our supervised machine learning approach for aspect extraction. We identified that when Mean Embeddings are provided as a feature to the SVM classifier, results get improved significantly. The system was further enhanced using a clever text pre-processing pipeline complemented with context sensitive spell correction. Our system is able to outperform the best results submitted for SemEval-2016 Task 5[i], in both restaurant and laptop domains.

The rest of the paper is organized as follows. In section 2, related work is discussed. Section 3 explains the SemEval-2016 Task 5 dataset. Section 4 elaborates our system in detail. Experimental results are discussed in section 5. Finally, section 6 concludes our paper.

2. Related Work

Aspect extraction is an important and challenging task in sentiment analysis [4]. There is previous work on aspect extraction based on different approaches. Frequency based methods of aspect extraction consider frequent words likely to be aspects. Frequent nouns and noun phrases are considered as frequent words in this approach [3]. Hu and Liu [5] consider single nouns and compound nouns to be aspects. However, not each frequent word in a review sentence refers an aspect, thus this assumption leads to low accuracies in the aspect extraction process.

Syntax-based methods for aspect extraction use syntactical relations between a sentiment word and the aspect it is about. The ability to find low frequent aspects is an advantage in this approach. Still, to have a good recall, many grammatical relations needed to be found. To address this challenge, the double propagation algorithm is used by Qiu et al. [6] and Zhang et al. [7]. Yet, the presence of implicit aspects is not addressed in this approach.

It has been observed that machine learning approaches have excelled in aspect extraction task in the recent literature [3]. Many supervised classifiers have been used for aspect extraction in the literature [8].

Hercig et al. [9] present a system that uses a maximum entropy classifier for aspect category detection. The system is fed in with a massive number of features in order to get competitive results. These features are categorized under semantic, constrained and unconstrained features. However, despite using many features, this classifier was not able to outrank the best performing systems at SemEval-2016 Task 5. It is observed that most of the best performing supervised machine learning models use SVM [10].

In contrast to the supervised machine learning methods, Toh et al. [11] presented a hybrid approach, which uses deep learning techniques along with a binary classifier[ii]. The model has been evaluated with restaurant-domain and laptop-domain datasets of SemEval-2016 Task 5. This model has achieved the best score in the SemEval-2016 Task 5, with an F1 score of 0.7303

for restaurant domain and 0.5194 for the laptop domain. We consider these results as our benchmark results. We try to outperform this complex system using a simple SVM combined with carefully crafted features.

3. SemEval-2016 Task 5 Dataset

The existence of a dataset such as the one provided by SemEval-2016 Task 5 provides a standardized evaluation technique to publish our results, and it can be compared fairly with other systems, which are evaluated on the same dataset. Previously many different researchers used various datasets in their publications, making it difficult to compare and contrast the techniques discussed. SemEval-2016 Task 5 consists of several subtasks and slots [12]. Our system focuses on Slot 1 - Aspect category identification in Subtask 1 - sentence level ABSA. Details of subtask 1 is as follows,

The task is to identify all opinion tuples when opinionated text is given about a target entity. Subtask 1 is composed of 3 slots.

- Slot 1 - Aspect category: Identify entity E and attribute A pairs (denoted E#A) in a given sentence. E and A are chosen from predefined entity types and attribute labels, respectively.
- Slot 2 - Opinion target extraction: Extraction of expression used in the sentence to refer to the entity identified in E#A pair.
- Slot 3 - Sentiment polarity: Identify polarity labels (“positive”, “negative”, “neutral”) for each identified E#A pair.

Our goal is to identify all aspect categories mentioned in each sentence. SemEval-2016 Task 5 dataset of English reviews for restaurant (training: 2000, testing 676 sentences) and laptop (training: 2500, testing 808 sentences) domains are used to train our SVM classifier. Training sentences have been annotated for opinions with respective aspect category while taking the context of the whole review into consideration. The sentences are classified under

12 and 81 classes in the restaurant and laptop domains, respectively.

4. System Description

In this section, we present our aspect extraction system. Our goal is to extract all the relevant aspect categories for a given sentence. We developed an SVM classifier for this task and evaluated its accuracy. The structure of our system is illustrated in Figure 1.

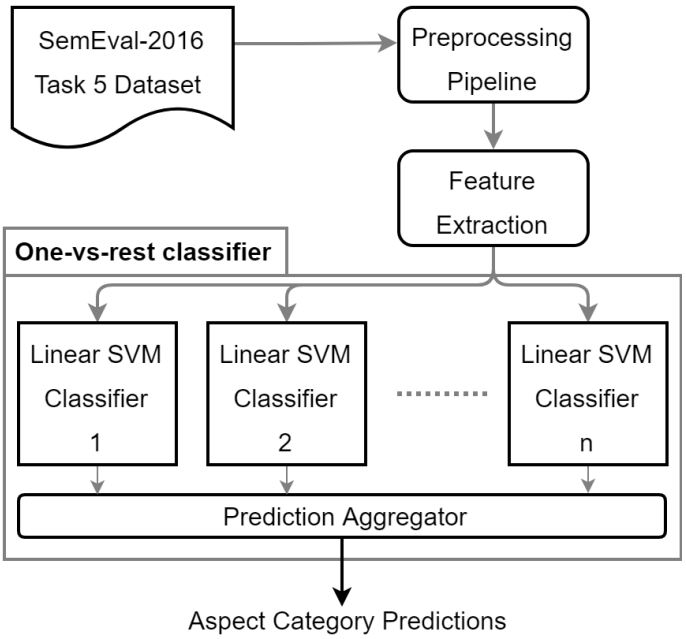

Figure 1 System Structure

4.1 Preprocessing

The dataset is stripped out of unnecessary content such as HTML, and the encoding of text is corrected. The pipeline neutralizes incorrect consecutive letters in words entered due to the excitement of the reviewer (s.a “sooooo”) as done by Machácek [13].

We then present the sentences for spell correction. Both isolated word[iii] and context sensitive spell correction[iv] were experimented with. We observed that context sensitive spell correction performs far superior than isolated word spell correction. For example, consider the sentence “I lve in the neightborhood and lve their piza”. Individual word spell correction corrected both occurrences of “lve” by “live” where context sensitive spell correction replaces the first “lve” by “live” and the second “lve” by “love”.

The reason to specifically use this service is that Bing spell check API[iv] provides correct context sensitive spell correction via machine learning and a statistical machine translation based on a highly contextual algorithm.

After the spell correction, English concatenations present in the sentences were expanded (i.e. 'can't' become 'cannot'). Punctuations present in the data were removed and all symbols were replaced with their word meanings using regular expressions (i.e. - % will be replaced using percent). Moreover, all occurrences of numerical prices and "$" symbols are replaced with a price indicator word. Finally, we remove commonly occurring English articles such as "a" and "an" from the text. Converting all the characters to lowercase is not performed at the preprocessing stage since text features such as Named Entities are case sensitive. However, when creating the lemmatized bag of words, the text is lowercased.

4.2 Features

The SVM classifier requires informative and effective features to improve the results. We came up with following features, which were extracted from the preprocessed text to train and test the SVM classifier. We identified some of these features from recent publications made on SemEval-2016 Task 5. Moreover, we introduce some of our own features, which were not tried out in previous studies for aspect extraction.

Shown below is the feature combination that contributed to the best F1 score of the SVM classifier according to 5-fold cross validation. The last 3 features on the list are newly introduced.

4.2.1 Lemmatized bag of words

Used Stanford CoreNLP[v] to tokenize the text, and the stop words were removed. Then the tokens were lemmatized and were provided as a feature to the SVM. UWB system [9] and BUTknot system [13] at SemEval-2016 have lemmatized the text in a similar manner. Lemmatized bag of words is the base feature of our system.

4.2.2 Custom built word lists

Restaurant domain - We manually compiled a collection of restaurant food and drink names. The food name list contains 1302 items and the drinks list consists of 1400 items.

Laptop domain - We built a collection of laptop manufacturer names, operating systems, processors, display resolutions, CPU quality, hard disks and laptop model series.

Custom word lists were used in past research and could be observed in the BUTknot system [13] at SemEval-2016.

4.2.3 Opinion target annotations

We extracted the opinion targets that were annotated in the training dataset. Lemmatized opinion targets were fed as a feature to the SVM with the respective category of the opinion target. BUTknot system [13] at SemEval-2016 has taken a similar approach.

4.2.4 Frequent words per category based on tf-idf score

We built a custom list of frequent words per category in the laptop domain. UWB system [9] at SemEval-2016 has implemented this feature in their approach. We used equation (3) to extract most important words for each of the categories and manually filtered noise words such as stop words and numbers. We created a document per category by combining all the sentences belonging to a particular aspect category together.

$$\boldsymbol{tf(word, category)} = \frac{f_w}{n_w} \quad (1)$$

$$\boldsymbol{idf(word, categories)} = \mathbf{log}\left(\frac{c_n}{1 + c_w}\right) \quad (2)$$

$$\boldsymbol{tf - idf}\ (\boldsymbol{word, category}) = \boldsymbol{tf(word, category)} * \boldsymbol{idf(word, categories)} \quad (3)$$

Where,

tf - term frequency score of a word in a given category

idf - inverse document frequency score of a given word among the categories

f_w - number of times a given word appears in a category

n_w - total number of words in the category

c_n - total number of categories

c_w - number of categories containing the given word

4.2.5 Presence of price in the text

Presence of price in numeric form in the raw text is fed as a feature. This feature is important to distinguish the price aspect of the respective entities. BUTknot system [13] at SemEval-2016 has presented similar feature in their approach.

4.2.6 Presence of exclamation mark in the text

Use of exclamation mark to express excitement is used as a feature. UWB system [9] at SemEval-2016 has used this feature in their approach.

4.2.7 Bag of five words at the end of sentence

The last five words of a sentence excluding stop words are fed as a feature to the SVM. UWB system [9] at SemEval-2016 has incorporated this feature in their classifier.

4.2.8 Named Entity Recognition (NER)

Indicated the presence of a person, organization, product or location in the text as a feature. SpaCy[vi] was used for NER extraction. Saias system [14] has used a similar feature to extract opinion target expression for SemEval-2015 Task 12. Ahiladas et al. [15] have used NER to extract food names in their Ruchi system [15]. IIT-TUDA at SemEval-2016 Task 5 by Kumar et al. [16] has also used NER for opinion target extraction. In contrast, we provided the extracted NER tags as direct features to the SVM classifier.

4.2.9 Head Nouns

We extract the head noun per sentence phrase, therefore a given sentence with more than one phrase would contain multiple head nouns. Part of speech (POS) tag is considered to select

a noun. Stanford CoreNLP[v] is used to parse the sentences and obtain POS tags of the words. Singular noun (NN), plural noun (NNS), proper noun (NNP), plural proper noun (NNPS) POS tags are considered when extracting nouns. If multiple nouns are present in the same sentence phrase, rightmost noun is selected as the head noun. Presence of each extracted head noun is presented to the SVM as a feature. Therefore, a feature is introduced to the SVM for each head noun identified.

This feature is not observed in past research. Instead, in past research, a single head noun per sentence has been used. For example, UWB system [9] at SemEval-2016 Task 5 has incorporated Bag of head words as a feature to their classifier. They have used the head of the sentence parse tree as the headword. Consider the sentence "The food was well prepared and the service impeccable". The word "food" is the head of the sentence parse tree and thus considered as the head noun of the sentence. However, our approach would pick up both "food" and "service" words from the separate sentence phrases of the sentence. This helps to capture multiple features describing multiple aspect categories present in a single sentence. We found that getting the head of the sentence from the sentence parse tree does not always provide correct head word as seen in the ablation results by Toh et al. [11].

4.2.10 Mean Embedding using word2vec

Word embedding represents a class of techniques that represent individual words as real-valued vectors in predefined vector space. Word2vec is a group of related models that are used to produce word embeddings [17]. Mean embedding vector for each sentence was calculated using word2vec GoogleNews vector pre-trained model[vii] and used as a feature for the SVM. This feature was not used in past research in aspect extraction. Equation (4) can be used to obtain the mean embedding vector.

$$\boldsymbol{MEV} = \sum_{\boldsymbol{i=1}}^{\boldsymbol{n}} \frac{\boldsymbol{vec(word_i)}}{\boldsymbol{n}} \quad \textbf{(4)}$$

Where,

MEV - Mean embedding vector

n - Number of words in the sentence

$vec(w)$ - Embedding of word w

4.3 SVM classifier

$$\boldsymbol{g(x) = w^T \phi(x) + b} \quad (5)$$

A Support Vector Machines (SVM) is a discriminative model used in machine learning. It uses the discriminant function shown in equation (5), where w is the weights vector, b is the bias, and $\phi(x)$ denotes nonlinear mapping from input space to high-dimensional feature space.

The parameters w and b are learnt automatically on the training dataset based on the principle of maximized margin as indicated in (6).

$$\min_{w,b} \ \frac{1}{2} W^T W + C \sum_{i=1}^{N} \xi_i \quad (6)$$

$$s.t. \begin{cases} y_i g(x_i) \geq 1 - \xi_i \\ \xi_i \geq 0, i = 1, \dots, N \end{cases}$$

where ξ_i denotes the slack variables and C is the penalty coefficient. Instead of solving this problem directly, it is converted to an equivalent quadratic optimization problem using Lagrange multipliers.

The training sample $(\tilde{x}_i, y_i)$ is called a support vector when satisfying the Lagrange multiplier $\alpha_i > 0$. By introducing a kernel function, the discriminant function can be represented as in equation (7).

$$\boldsymbol{g(x) = \sum_{i=1}^{\tilde{N}} \alpha_i y_i K(\tilde{x}_i, x)} \quad (7)$$

We used a one-vs-rest multi-label support vector machine classifier to classify the text into multiple categories. Therefore, in the restaurant domain, 12 classifiers were used and in the laptop domain, 81 SVM classifiers were used. A sentence may be categorized into multiple categories. We used cross-validation for selecting the optimal parameters of the classifier.

According to Joachims [18], most text categorization problems are linearly separable. Due

to this reason and the higher dimensionality of the feature vectors, it was more suitable to use a linear kernel. Furthermore, training an SVM with a linear kernel is faster compared to other kernels and there is only one parameter (regularization parameter) to be optimized in the Linear SVM.

5. Experimental Results

Table 1 presents results of ablation experiments on the testing data of the two domains using the SVM classifier. It is evident that the Mean Embeddings feature contributes significantly to increase the accuracy of the system compared to other features in the two domains we considered.

Table 1 Experimental Results for SVM

Feature	**Restaurant F1**	**Laptop F1**
Lemmatization	0.6034	0.3731
+ Exclamation mark	0.6022	0.3712
+ End words	0.6081	0.4146
+ Named Entities	0.6111	0.4282
+ Has price	0.6134	0.4255
+ Term list	0.6692	0.4774
+ Head nouns	0.685	0.4906
+ Mean embeddings	0.7203	0.4991
+ Preprocessing	0.7418	0.5221
Benchmark	0.7303	0.5194

The benchmark system uses a complex hybrid model with Convolutional Neural Network (CNN) and Feedforward Neural Network(FNN), whereas we achieve better results using a simple SVM fed with clever features.

Highlighting the significance of preprocessing for the features used with SVM, the F1 score drops to 0.7203 and 0.4991 in restaurant and laptop domain respectively without the preprocessing pipeline.

The inclusion of context sensitive spell correction during data preprocessing was not observed in past literature and we emphasize that context sensitive spell correction helps to perform more accurate aspect extraction. This is because customer reviews are written by laypeople, and reviews are often written using short-hand versions of words typed in a hurry using a mobile device. The final F1 result increased by 1.78% when isolated spell correction (0.7288) was replaced with context sensitive spell correction (0.7418) in the restaurant domain.

6. Conclusion

In this paper, we presented an effective SVM classifier that performs better than the state-of-the-art classifiers for aspect extraction. Moreover, we introduced a pre-processing pipeline to enhance the accuracy of the classifier. All features to the SVM classifier except the custom compiled lists can be automatically tuned for a new domain. We were able to outperform the best F1 score reported for the SemEval-2016 Task 5 in both restaurant and laptop domains using our classifier. We observed the use of deep learning for aspect extraction as an emerging trend in the field. Therefore, as future work we hope to perform more research on aspect extraction using deep learning techniques. Moreover, we would like to experiment the benefits of a hybrid classifier that uses deep learning and supervised machine learning.

References

[1] B. Lu, M. Ott, C. Cardie and B. K. Tsou, "Multi-aspect sentiment analysis with topic models," in *Data Mining Workshops (ICDMW), 2011 IEEE 11th International Conference on*, 2011.

[2] C. Lin and Y. He, "Joint sentiment/topic model for sentiment analysis," in *Proceedings of the 18th ACM conference on Information and knowledge*

management, 2009.

[3] K. Schouten and F. Frasincar, "Survey on aspect-level sentiment analysis," *IEEE Transactions on Knowledge and Data Engineering,* vol. 28, pp. 813-830, 2016.

[4] T. A. Rana and Y.-N. Cheah, "Aspect extraction in sentiment analysis: comparative analysis and survey," *Artificial Intelligence Review,* vol. 46, pp. 459-483, 2016.

[5] M. Hu and B. Liu, "Mining opinion features in customer reviews," in *AAAI*, 2004.

[6] G. Qiu, B. Liu, J. Bu and C. Chen, "Expanding domain sentiment lexicon through double propagation.," in *IJCAI*, 2009.

[7] L. Zhang, B. Liu, S. H. Lim and E. OḂrien-Strain, "Extracting and ranking product features in opinion documents," in *Proceedings of the 23rd international conference on computational linguistics: Posters*, 2010.

[8] W. Medhat, A. Hassan and H. Korashy, "Sentiment analysis algorithms and applications: A survey," *Ain Shams Engineering Journal,* vol. 5, pp. 1093-1113, 2014.

[9] T. Hercig, T. Brychc\in, L. Svoboda and M. Konkol, "Uwb at semeval-2016 task 5: Aspect based sentiment analysis," in *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*, 2016.

[10] B. Wang and M. Liu, *Deep Learning for Aspect-Based Sentiment Analysis,* Stanford University report, https://cs224d. stanford. edu/reports/WangBo. pdf, 2015.

[11] Z. Toh and J. Su, "NLANGP at SemEval-2016 Task 5: Improving Aspect Based

Sentiment Analysis using Neural Network Features.," in *SemEval@ NAACL-HLT*, 2016.

[12] M. Pontiki, D. Galanis, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, M. AL-Smadi, M. Al-Ayyoub, Y. Zhao, B. Qin, O. De Clercq and others, "SemEval-2016 task 5: Aspect based sentiment analysis," in *ProWorkshop on Semantic Evaluation (SemEval-2016)*, 2016.

[13] J. Machácek, "BUTknot at SemEval-2016 Task 5: Supervised Machine Learning with Term Substitution Approach in Aspect Category Detection.," in *SemEval@ NAACL-HLT*, 2016.

[14] J. Saias, "Sentiue: Target and aspect based sentiment analysis in semeval-2015 task 12," in *Association for Computational Linguistics*, 2015.

[15] B. Ahiladas, P. Saravanaperumal, S. Balachandran, T. Sripalan and S. Ranathunga, "Ruchi: Rating individual food items in restaurant reviews," in *12th International Conference on Natural Language Processing*, 2015.

[16] A. Kumar, S. Kohail, A. Kumar, A. Ekbal and C. Biemann, "IIT-TUDA at SemEval-2016 Task 5: Beyond Sentiment Lexicon: Combining Domain Dependency and Distributional Semantics Features for Aspect Based Sentiment Analysis.," in *SemEval@ NAACL-HLT*, 2016.

[17] T. Mikolov, K. Chen, G. Corrado and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781,* 2013.

[18] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," *Machine learning: ECML-98,* pp. 137-142, 1998.

[i] http://alt.qcri.org/semeval2016/task5/

[ii] https://github.com/JohnLangford/vowpal_wabbit/wiki

[iii] https://github.com/blatinier/pyhunspell

[iv] https://azure.microsoft.com/en-us/services/cognitive-services/spell-check/

[v] https://stanfordnlp.github.io/CoreNLP/

[vi] https://spacy.io/

[vii] https://code.google.com/archive/p/word2vec/