Know-Center at SemEval-2016 Task 5:
Using Word Vectors with Typed Dependencies
for Opinion Target Expression Extraction

Stefan Falk

Know-Center GmbH

Inffeldgasse 13

Graz, 8010, Austria

`sfalk@know-center.at`

Andi Rexha

Know-Center GmbH

Inffeldgasse 13

Graz, 8010, Austria

`arexha@know-center.at`

Roman Kern

Know-Center GmbH

Inffeldgasse 13

Graz, 8010, Austria

`rkern@know-center.at`

Abstract

This paper describes our participation in SemEval-2016 Task 5 for Subtask 1, Slot 2. The challenge demands to find domain specific target expressions on sentence level that refer to reviewed entities. The detection of target words is achieved by using word vectors and their grammatical dependency relationships to classify each word in a sentence into *target* or *non-target*. A heuristic based function then expands the classified target words to the whole target phrase. Our system achieved an F1 score of 56.816% for this task.

1 Introduction

Nowadays, modern technologies allow us to collect customer reviews and opinions in a way that changed the sheer amount of information available to us. For that matter the requirement to extract useful knowledge from this data rose up to a point where machine learning algorithms can help to accomplish this much faster and easier than humanly possible. Natural language processing (NLP) emerges as an interfacing tool between human natural language and many technical fields such as machine learning and information extraction.

This article describes our approach towards Opinion Target Expression (OTE) extraction as defined by Task 5 for Subtask 1, Slot 2 of the SemEval-2016 (Pontiki et al., 2016) challenge. The core goal behind Slot 2 in Subtask 1 of Task 5 is to extract consecutive words which, by means of a natural language, represent the opinion target expression. The opinion target expression is that part of a sentence

which stands for the entity towards which an opinion is being expressed. An example could be the word "*waitress*" in the sentence "*The waitress was very nice and courteous the entire evening.*".

The evaluation for Slot 2 fell into evaluation phase A, where provided systems were tested in order to return a list of target expressions for each given sentence in a review text. Each target expression was an annotation composed of the index of the starting and end character of the particular expression as well as its corresponding character string.

For our system we decided to used word vectors (Mikolov et al., 2013a; Mikolov et al., 2013b). Word vectors (Bengio et al., 2003) are distributed representations which are designed to carry contextual information of words if their training meets certain criteria. We also used typed grammatical dependencies to extract structural information from sentences. Furthermore we used a sentiment parser to determine the polarity of words.

2 External Resources

Our system uses Stanford dependencies (Chen and Manning, 2014) and utilizes the Stanford Sentiment Treebank (Socher et al., 2013) for sentiment word detection.

3 System for Slot 2: Opinion Target Extraction

For the Opinion Target Extraction (OTE) task, in order to extract different features, we followed a supervised approach. We train and test different combinations of these features first at the word level and

266

following on the provided training data[1] on sentence level before using our classifier for the final evaluation. There are two essential steps performed by our system to correctly annotate opinion target expressions.

1. Classify each word of a sentence as either *target* or *non-target*

2. Given each *target* word, find the full target phrase

For classification we use a L2-regularized L2-loss support vector dual classification[2] provided by the LIBLINEAR (Fan et al., 2008) library. In the second step we use heuristics, based on observations and statistical information we extracted from the training data. They key obversvation is that target expressions are usually composed of noun phrases and/or proper nouns. In all trials we allow only certain Part of Speech (PoS) tags for target words which are NN, NNS, NNP, NNPS and FW from the Penn Treebank (Marcus et al., 1993) listed in Table1.

PoS-Tag	Name
NN	Noun, singular or mass
NNS	Noun, plural
NNP	Proper noun, singular
NNPS	Proper noun, plural
FW	Foreign word

Table 1: Used PoS-Tags and their meaning according to the Penn Treebank.

4 Features

In this section we describe the different set of features we evaluated and how they can be extracted.

4.1 Token

We obtain tokens by using the Stanford Parser and extract all tokens from the available reviews used for training. We are then able to use tokens as a feature for the classifier.

4.2 Word Vector Feature

As another feature for words we are using the pre-trained word vectors of Google News dataset[3]. Each

---

[1] Using the English data set
[2] Implementation of a Support Vector Regression Machine
[3] https://code.google.com/archive/p/word2vec/

word vector is a 300-dimensional, real-valued vector.

4.3 Combined Typed Dependencies Feature

Using Stanford dependencies, we extract for each word in a sentence its typed dependencies to other words in the sentence.
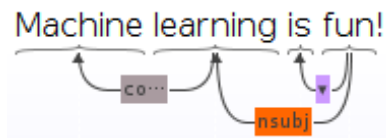

Figure 1: Shown are typed dependencies from Stanford dependencies visualized with grammarscope.

Given the sentence "*Machine learning is fun!*", the feature for "*learning*" is `compound;nsubj` which are the present relations for this word. We extract all typed dependency combinations from all provided words in the training set and use these in a Bag of Words (BoW) sparse vector model. In order to normalize this feature we order the relations alphabetically and remove duplicates. For example `det;amod;amod` gets normalized to `amod;det`.

4.4 Individual Typed Dependencies Feature

Another approach is to look at the dependencies individually. We use the set of present grammatical relations as feature vector and set corresponding fields to 1 if the word does own such a relation and 0 otherwise. We are testing the two possible options of *directed* and *undirected* dependencies to see if this additional information has an impact on the end result. A short overview of a textual representation of these features can be seen in Table 2.

4.4.1 Undirected

In the undirected approach we extract the relations of each word from the data and use the resulting set of *present* relations as feature vector. From the training set we extracted 105 different undirected relations. Here the directional information of the grammatical dependency is lost.

4.4.2 Directed

For the directed approach we preserve the direction in terms of *incoming* or *outgoing* relations for each grammatical relation. As an example, the word

"*learning*" from Figure 1 has an outgoing relation `compound+` and an incoming relation `nsubj-` where + depicts the outgoing relation and − the incoming respectively. This way we found 164 different relations in the training set.

Typed Dependency Features		
Directed	**Word**	**Feature**
no	*learning*	`coumpund;subj;`
yes	*learning*	`coumpund-;subj+;`

Table 2: The table shows two example of *undirected* and *directed* typed dependency features for the word "*learning*" in the sentence "*Machine learning is fun!*".

4.5 Sentimend Dependency Feature

For a given word we determine whether it has a grammatical relation to a sentiment word. A sentiment word is a word that can have a positive or negative meaning for example "*breathtaking*" in "*The food was breathtaking!*". We are not considering a directional approach which makes this a binary feature.

5 Results

This section describes the results we achieved on the restaurant domain of the SemEval-2016 aspect based sentiment analysis (ABSA) on Task 5, Slot 2. It also explains how we trained and tested our system only on the provided training data.

5.1 Word-Level Feature Evaluation

We determine how well our different features are performing by splitting the train data available and using 80% training and 20% test data. In Table 3 the performance on the *target*-word class of the individual features are shown depicting the performance of classifying single words as targets or non-targets.

The results for the similarly token-based approach outperforms the other approaches. The weighted average for *Token* settles at 0.696 and very similar *Token + combined typed dependencies* at 0.697. None of the word vector approaches outperforms these two.

5.2 Testing Features

To test our features we use the same training/testing split of the SemEval-2016 training data and utilize

Evaluation class: target-word			
Feature	**Prec.**	**Rec.**	**F1 score**
			wt. avg.
Token	0.649	0.749	0.696
+ comb. typed dep.	0.657	0.741	0.697
w2v	0.595	0.649	0.621
+ comb. typed dep.	0.610	0.696	0.650
+ indiv. typed dep.			
undirected	0.595	0.587	0.591
undir. + sentiment	0.587	0.598	0.592
directed	0.568	0.674	0.617
dir. + sentiment	0.574	0.679	0.622

Table 3: The resulting F1 scores for the *target*-word class using different features on word-level over a 80/20 training/test split of the provided training data.

it to train the classifier and run the SemEval-2016 evaluation tool respectively. In order to annotate the Opinion Target Expressions (OTE) our system first classifies single tokens of a sentence into *target* or *non-target* and further tries to complete the target expression. The completion of the target expression is heuristic based and looks at existing incoming or outgoing *compound* relations using Stanford dependencies (Chen and Manning, 2014). Each compound relation is added to the target phrase and correspondingly extended.

Feature	**Prec.**	**Rec.**	**F1 score**
Token	0.498	0.488	0.493
+ comb. typed dep.	0.460	0.433	0.446
w2v	0.484	0.598	0.535
+ comb. typed dep.	0.498	0.520	0.509
+ indiv. typed dep.			
undirected	0.482	0.550	0.513
undir. + sentiment	0.466	0.552	0.505
directed	0.465	0.623	0.532
dir. + sentiment	0.465	0.625	0.533

Table 4: Shown are evaluation F1 scores given by the SemEval-2016 evaluation tool for different features and feature combinations used for training on a 80/20 training/test split of the provided training data.

In Table 4 we can see the results for the evaluation. It shows that despite having a better result on word-level, the token-based approach falls behind the word vector approach. It is interesting to see, that adding the *undirected* grammatical relations as feature does not improve the F1 score but performs even worse than the pure *w2v* approach. However,

taking *directed* dependencies into account does improve the results again. We can see that for directed dependencies the recall improves but in contradiction the precision declines resulting in a higher miss-classification rate and thus in a lower F1 score than we were hoping to see.

5.3 Official Evaluation Results: Restaurant domain

Our submitted system is using the individual (directed) typed dependencies and the sentiment information combined with word vectors as features. The official results for participating *unconstrained* systems for Slot 2: Opinion Target Extraction can be seen in Table 5. The table shows the F1-score for all participating *unconstrained* systems. Our system was able to outperform the baseline and a few others. Considering only unconstrained systems, Know-Center reached rank 6 out of 10 (excluding the baseline results).

System	F1 score
NLANGP	72.340%
AUEB.	70.441%
UWB	67.089%
GTI	66.553%
bunji	64.882%
Know-Center	**56.816%**
BUAP	50.253%
Baseline	44.071%
IHS-R.	43.808%
IIT-T.	42.603%
SeemGo	34.323%

Table 5: Shown are the official evaluation results for Subtask 1, Slot 2 of Task 5 from the SemEval-2016 challenge for the Restaurant domain. The table shows only results for *unconstrained* systems.

6 Conclusions and Future Work

In this paper, we presented our approach for SemEval-2016 Task 5 for Subtask 1, Slot 2 in order to introduce ourselves to this particular evaluation task. Our solution might have potential for improvement and might be able to reach a much better ranking than what it achieved in the course of this challenge. Therefore, we will continue our work by focusing on finding the correct target phrase annotation given one or more target words. A drawback

of our solution is the heuristic based selection of the *full* target phrase and we are curious about how we can improve our results with more sophisticated techniques for target phrase labelling.

Acknowledgments

References

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. 2003. A neural probabilistic language model. *J. Mach. Learn. Res.*, 3:1137–1155, March.

Danqi Chen and Christopher Manning. 2014. A Fast and Accurate Dependency Parser using Neural Networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 740–750, Doha, Qatar, October. Association for Computational Linguistics.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of english: The penn treebank. *COMPUTATIONAL LINGUISTICS*, 19(2):313–330.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. *CoRR*, abs/1310.4546.

Maria Pontiki, Dimitrios Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, Véronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeny Kotelnikov, Nuria Bel, Salud María Jiménez-Zafra, and Gülşen Eryiğit. 2016. SemEval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of*

the 10th International Workshop on Semantic Evaluation, SemEval '16, San Diego, California, June. Association for Computational Linguistics.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Stroudsburg, PA, October. Association for Computational Linguistics.