LT3: Applying Hybrid Terminology Extraction to Aspect Based Sentiment Analysis

Orphée De Clercq, Marjan Van de Kauter, Els Lefever and Véronique Hoste
LT^3, Language and Translation Technology Team
Department of Translation, Interpreting and Communication – Ghent University
Groot-Brittanniëlaan 45, 9000 Ghent, Belgium
Firstname.Lastname@UGent.be

Abstract

The LT3 system perceives ABSA as a task consisting of three main subtasks, which have to be tackled incrementally, namely aspect term extraction, aggregation and polarity classification. For the first two steps, we see that employing a hybrid terminology extraction system leads to promising results, especially when it comes to recall. For the polarity classification, we show that it is possible to gain satisfying accuracies, even on out-of-domain data, with a basic model employing only lexical information.

1 Introduction

There exists a large interest in sentiment analysis of user-generated content. Until recently, the main research focus has been on discovering the overall polarity of a certain text or phrase. A noticeable shift has occurred to consider a more fine-grained approach, known as aspect based sentiment analysis (ABSA). For this task the goal is to automatically identify the aspects of given target entities and the sentiment expressed towards each of them. In this paper, we present the LT3 system that participated in this year's SemEval 2015 ABSA task. Though the focus was on the same domains (restaurants and laptops) as last year's task (Pontiki et al., 2014), it differed in two ways. This time, entire reviews were to be annotated and for one subtask the systems were confronted with an out-of-domain test set, unknown to the participants.

The task ran in two phases. In the first phase (Phase A), the participants were given two test sets (one for the laptops and one for the restaurants domain). The restaurant sentences were to be annotated with automatically identified <*target*, *aspect category*> tuples, the laptop sentences only with the identified aspect categories. In the second phase (Phase B), the gold annotations for the above two datasets, as well as for a hidden domain, were given and the participants had to return the corresponding polarities (positive, negative, neutral). For more information we refer to Pontiki et al. (2015).

We tackled the problem by dividing the ABSA task into three incremental subtasks: (i) aspect term extraction, (ii) aspect term aggregation and (iii) aspect term polarity estimation (Pavlopoulos and Androutsopoulos, 2014). The first two are at the basis of Phase A, whereas the final one constitutes Phase B. For the first step, viz. extracting terms (or *targets*), we wanted to test our in-house hybrid terminology extraction system (Section 2). Next, we performed a multiclass classification task relying on a feature space containing both lexical and semantic information to aggregate the previously identified terms into the domain-specific and predefined aspects (or *aspect categories*) (Section 3). Finally, we performed polarity classification by deriving both general and domain-specific lexical features from the reviews (Section 4). We finish with conclusions and prospects for future work (Section 5).

2 Aspect Term Extraction

Before starting with any sort of classification, it is essential to know which entities or concepts are present in the reviews. According to Wright (1997), these "words that are assigned to concepts used in

the special languages that occur in subject-field or domain-related texts" are called terms. Translated to the current challenge, we are thus looking for words or terms specific to a specific domain or interest, such as the restaurant domain.

In order to detect these terms, we tested our in-house terminology extraction system TExSIS (Macken et al., 2013), which is a hybrid system combining linguistic and statistical information. For the linguistic analysis, TExSIS relies on tokenized, Part-of-Speech tagged, lemmatized and chunked data using the LeTs Preprocess toolkit (Van de Kauter et al., 2013), which is incorporated in the architecture. Subsequently, all words and chunks matching certain Part-of-Speech patterns (i.e. nouns and noun phrases) were considered as candidate terms. In order to determine the specificity of and cohesion between these candidate terms, we combine several statistical filters to represent the termhood and unithood of the candidate terms (Kageura and Umino, 1996). To this purpose, we employed Log-likelihood (Rayson and Garside, 2000), C-value (Frantzi et al., 2000) and termhood (Vintar, 2010). All these statistical filters were calculated using the Web 1T 5-gram corpus (Brants and Franz, 2006) as a reference corpus.

After a manual inspection of the first output for the training data, we formulated some filtering heuristics. We filter out terms consisting of more than six words, terms that refer to location names or that contain sentiment words. Locations are found using the Stanford CoreNLP toolkit (Manning et al., 2014) and for the sentiment words, we filter those terms occurring in one of the following sentiment lexicons: AFINN (Nielsen, 2011), General Inquirer (Stone et al., 1966), NRC Emotion (Mohammad and Turney, 2010; Mohammad and Yang, 2011), MPQA (Wilson et al., 2005) and Bing Liu (Hu and Liu, 2004).

The terms that resulted from this filtered TExSIS output, supplemented with those terms that were annotated in the training data but not recognized by our terminology extraction system, were all considered as candidate terms. Finally, this list of candidate targets was further extended by also including coreferential links as null terms. Coreference resolution of each individual review was performed with the Stanford multi-pass sieve coreference resolution system (Lee et al., 2011). We should also point out that we only allowed terms to be identified in the test data when a sentence contains a subjective opinion. This was done by running it through the above-mentioned sentiment lexicons.

3 Phase A

Given a list of possible candidate terms, the next step consists in aggregating these terms to broader aspect categories. As our main focus was on combining aspect term extraction with aggregation and since no targets were annotated for the laptops, we decided to focus on the restaurants domain. The organizers provided the participants with training data consisting of 254 annotated restaurant reviews. The task was then to assign each identified term to a correct aspect category.

For the classification task, we relied on a rich feature space for each of the candidate targets and performed classification into the domain-specific categories. Whereas the annotations allow for a two-step classification procedure by first classifying the main categories and afterwards the subcategories, we chose to perform the joint classification as this yielded better results in our exploratory experiments.

3.1 Feature Extraction

For all candidate terms present in our data sets we derived a number of lexical and semantic features. For those candidate targets that have been recognized as anaphors (see Section 2), these features were derived based on the corresponding antecedent.

First of all, we derived bag-of-words token unigram features of the sentence in which a term occurs in order to represent some of the lexical information present in each of the categories.

The main part of our feature vectors, however, was made up of semantic features, which should enable us to aggregate our aspect terms into the predefined categories. These semantic features consist of:

1. **WordNet features**: for each main category, a value is derived indicating the number of (unique) terms annotated as aspect terms from that category in the training data that (1) co-occur in the

synset of the candidate term or (2) which are a hyponym/hypernym of a term in the synset. In case the candidate term is a multi-word term whose full term is not found, this value is calculated for all nouns in the multi-word term and the resulting sum is divided by the number of nouns.

2. **Cluster features**: using the implementation of the Brown hierarchical word clustering algorithm (Brown et al., 1992) by Liang (2005), we derived clusters from the Yelp dataset[1]. Then, we derived for each main category a value indicating the number of (unique) terms annotated as aspect terms from that category in the training data that co-occur with the candidate term in the same cluster. Since clusters can only contain single words, we calculate this value for all the nouns in a multi-word term and take the mean of the resulting sum.

3. **Linked Open Data (LOD) features**: using DBpedia (Lehmann et al., 2013), we included binary values indicating whether a candidate term occurs in one of the following DBpedia categories: *Foods*, *Cuisine*, *Alcoholic_beverages*, *Non-alcoholic_beverages*, *Atmosphere*, *People_in_food_and_agriculture_occupations* or *Food_services_occupations*. These features were automatically derived using the RapidMiner Linked Open Data Extension (Paulheim et al., 2014).

4. **Training data features**: number of annotations in the training data for each of the main categories. We filtered out candidate terms for which all of these feature values are "0", but decided to keep proper nouns and proper noun phrases.

3.2 Classification and Results

For all our experiments, we used LIBSVM (Chang and Lin, 2001). In order to tune our system, we split the training data into a train (90%) and test fold (10%) and ran various rounds of experiments, after which we manually analyzed the output. Based on this analysis, we were able to derive some post-processing heuristics to rule out some of the low-hanging fruit (i.e. misclassification which could be ruled out univocally). To do so, we built a dictionary containing all targets annotated in the training data, together with their associated category label(s). In case our classifier assigns a main category to a target term that is never associated with the respective target in the training dictionary, we overrule the classification output and replace it by the (most frequent) category-subcategory label that is associated with this target in the training dictionary.

The results of our system on the final test set and rank are presented in Table 1, where Slot 1 refers to the aspect category classification and Slot 2 to the task of finding the correct opinion target expressions (or terms).

Slot	Precision	Recall	F-score	Rank
Slot 1	51.54	56.00	53.68	8/15
Slot 2	36.47	79.34	49.97	13/21
Slot 1,2	29.44	44.73	35.51	6/13

Table 1: Results of the LT3 system on Phase A

For the design of our system we wanted to focus most on the combination of Slot 1 and 2, i.e. finding the target terms and being able to classify them in the correct category. This is the most difficult task of all three, hence the lower F-scores in general (Pontiki et al., 2015). Though there is much room for improvement for our system, we do observe that our rank increases for this more difficult task. Our precision scores are rather low, but we obtain the best recall scores for Slot 2 and Slot 1,2. For Slot 1,2 we are able to find 378 of the 845 possible targets, resulting in the best recall score of all participating systems (e.g. 44.73 compared to a recall score of 41.73 obtained by the winning team).

This leads us to conclude that there's quite some room for improvement for the aggregation phase. Normally, the similarity between terms is first computed after which some sort of clustering is performed

4 Phase B

In recent years, sentiment analysis has been a popular research strand. An example is last year's SemEval task 9 Sentiment Analysis in Twitter, which drew over 45 participants. The competition revealed that the best systems use supervised machine learning techniques and rely much on lexical features in the form of n-grams and sentiment lexicons (Rosenthal et al., 2014). For Phase B, in which we had all gold standard terms and aspect categories avail-

[1] https://www.yelp.com/academic_dataset

able, we decided to extend our LT3 system with another classification round where we classify every aspect as positive, negative or neutral. All features are derived from the sentence in which the terms were found and we participated in all three domains.

4.1 Feature Extraction

We implemented a number of lexical features. First of all, we derived bag-of-words token unigram features. Then, we also generated features using two of the more well-known sentiment lexicons: General Inquirer (Stone et al., 1966) and Bing Liu (Hu and Liu, 2004) and a manually constructed list of negation cues based on the training data of SemEval-2014 task 9 (Van Hee et al., 2014). Moreover, for both the restaurants and laptops domain we created a list of all the domain-specific positive, negative and neutral words based on the training data. For the hotels we were not able to compile such a list.

Finally, we also included PMI features based on three domain-specific datasets. PMI (pointwise mutual information) values indicate the association of a word with positive and negative sentiment: the higher the PMI score, the stronger the word-sentiment association. We calculated this for each unigram based on the word-sentiment associations found in the respective training dataset. PMI values were calculated as follows:

$$PMI(w) = PMI(w, positive) - PMI(w, negative) \quad (1)$$

As the equation shows, the association score of a word with negative sentiment is subtracted from the word's association score with positive sentiment. For the restaurants domain we relied on the Yelp dataset (cfr. Section 3.1), for the laptops domain on a subset of the Amazon electronics dataset (McAuley and Leskovec, 2013), and for the hidden – hotel – domain we worked with reviews collected from TripAdvisor (Wang et al., 2011). All datasets were filtered by only including reviews with strong subjective ratings (e.g. we preferred a 5 star rating for positive reviews over one of 3 stars).

4.2 Classification and Results

We again used LIBSVM as our learner. For the restaurants and laptops domain, we used the respective training data sets. For the hidden (hotel) domain, we only used the restaurants training data since we assumed hotels to be more similar to restaurants than they are to laptops. The results of our system are presented in Table 2.

Domain	Accuracy	Rank
Restaurants	75.03	4/14
Laptops	73.76	5/13
Hotels	80.53	2/9

Table 2: Result of the LT3 system on Phase B

Our results show that using only lexical features already results in quite satisfying accuracy scores for all three domains. Considering the hotels dataset, we can conclude that having training data available from a very similar domain does already result in a satisfying accuracy (our system has the second-best score on the hidden domain). In the future, we will investigate the performance gain when also including domain-specific training data.

5 Conclusions and Future Work

We presented the LT3 system, which is able to tackle the aspect based sentiment analysis task incrementally by first deriving candidate terms, after which these are classified into various categories and polarities. Applying a hybrid terminology extraction system to the first phase seems to be a promising approach. Our experiments revealed that we are able to receive high recall for the task of deriving targets and aspect categories using a variety of lexical and semantic features. When it comes to the polarity estimation, we see that a classifier mostly relying on lexical information achieves a satisfying performance, even on out-of-domain data.

Based on our results, we see different directions for follow-up research. For the term extraction, we will focus on more powerful filtering techniques. With respect to term aggregation, we will explore new techniques of clustering our list of candidate terms in different manners. Furthemore, we will explore in future experiments to which extent deeper syntactic, semantic and discourse modelling leads to better polarity classification. Since the TExSIS system was developed as a multilingual framework (Macken et al., 2013), we are currently translating the LT3 system so that it can handle Dutch reviews.

References

Thorsten Brants and Alex Franz. 2006. Web 1T 5-gram Version 1 LDC2006T13. Web Download.

Peter F. Brown, Peter V. deSouza, Robert L. Mercer, Vincent J. Della Pietra, and Jenifer C. Lai. 1992. Class-based n-gram models of natural language. *Computational Linguistics*, 18:467–479.

Chih-Chung Chang and Chih-Jen Lin, 2001. *LIBSVM: a library for support vector machines*.

Katerina Frantzi, Sophia Ananiadou, and Hideki Mima. 2000. Automatic recognition of multi-word terms: the C-value/NC-value method. *International Journal on Digital Libraries*, 3(2):115–130.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, KDD04*, pages 168–177, New York, NY. ACM.

Kyo Kageura and Bin Umino. 1996. Methods of automatic term recognition. A review. *Terminology*, 3(2):259–289.

Heeyoung Lee, Yves Peirsman, Angel Chang, Nathanael Chambers, Mihai Surdeanu, and Dan Jurafsky. 2011. Stanford's multi-pass sieve coreference resolution system at the conll-2011 shared task. In *Proceedings of the CoNLL-2011 Shared Task*.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, Soren Auer, and Christian Bizer. 2013. DBpedia – A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia. *Semantic Web Journal*.

Percy Liang. 2005. Semi-supervised learning for natural language. In *MASTERS THESIS, MIT*.

Lieve Macken, Els Lefever, and Véronique Hoste. 2013. TExSIS: Bilingual Terminology Extraction from Parallel Corpora Using Chunk-based Alignment. *Terminology*, 19(1):1–30.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*.

Julian McAuley and Jure Leskovec. 2013. Hidden factors and hidden topics: Understanding rating dimensions with review text. In *Proceedings of the 7th ACM Conference on Recommender Systems*, RecSys '13, pages 165–172.

Saif Mohammad and Peter Turney. 2010. Emotions Evoked by Common Words and Phrases: Using Mechanical Turk to Create an Emotion Lexicon. In *Proceedings of the NAACL-HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, LA, California.

Saif Mohammad and Tony Yang. 2011. Tracking Sentiment in Mail: How Genders Differ on Emotional Axes. In *Proceedings of the 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis (WASSA 2011)*, pages 70–79, Portland, Oregon. ACL.

Finn Nielsen. 2011. A new anew: Evaluation of a word list for sentiment analysis in microblogs. In *Proceedings of the ESWC2011 Workshop on Making Sense of Microposts: Big things come in small packages.*

Heiko Paulheim, Petar Ristoski, Evgeny Mitichkin, and Christian Bizer. 2014. Data mining with background knowledge from the web. In *Proceedings of the 5th RapidMiner World.*

John Pavlopoulos and Ion Androutsopoulos. 2014. Aspect term extraction for sentiment analysis: New datasets, new evaluation measures and an improved unsupervised method. In *Proceedings of the 5th Workshop on Language Analysis for Social Media (LASM)@ EACL.*

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35, Dublin, Ireland, August. Association for Computational Linguistics and Dublin City University.

Maria Pontiki, Dimitris Galanis, Harris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. Semeval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, Denver, Colorado.

Paul Rayson and Roger Garside. 2000. Comparing corpora using frequency profiling. In *Proceedings of the workshop on Comparing corpora, 38th annual meeting of the Association for Computational Linguistics*, pages 1–6, Hong Kong, China.

Sara Rosenthal, Alan Ritter, Preslav Nakov, and Veselin Stoyanov. 2014. Semeval-2014 task 9: Sentiment analysis in twitter. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 73–80, Dublin, Ireland, August. Association for Computational Linguistics and Dublin City University.

Philip J. Stone, Dexter C. Dunphy, Marshall S. Smith, and Daniel M. Ogilvie. 1966. The General Inquirer: A Computer Approach to Content Analysis. *MIT Press*.

Marjan Van de Kauter, Geert Coorman, Els Lefever, Bart Desmet, Lieve Macken, and Véronique Hoste. 2013.

LeTs Preprocess: The multilingual LT3 linguistic preprocessing toolkit. *Computational Linguistics in the Netherlands Journal*, 3:103–120.

Cynthia Van Hee, Marjan Van de Kauter, Orphee De Clercq, Els Lefever, and Veronique Hoste. 2014. Lt3: Sentiment classification in user-generated content using a rich feature set. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 406–410, Dublin, Ireland, August. Association for Computational Linguistics and Dublin City University.

Špela Vintar. 2010. Bilingual term recognition revisited: The bag-of-equivalents term alignment approach and its evaluation. *Terminology*, 16:141–158.

Hongning Wang, Yue Lu, and ChengXiang Zhai. 2011. Latent aspect rating analysis without aspect keyword supervision. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, pages 618–626.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing, HLT05*, pages 347–354, Stroudsburg, PA. ACL.

Sue Ellen Wright. 1997. Term selection: the initial phase of terminology management. In Sue Ellen Wright and Gerhard Budin, editors, *Handbook of terminology management*, pages 13–23. John Benjamins, Amsterdam.