XRCE: Hybrid Classification for Aspect-based Sentiment Analysis

Caroline Brun, Diana Nicoleta Popa, Claude Roux
Xerox Research Centre Europe
6, chemin de Maupertuis
38240 Meylan, France
{caroline.brun, diana.popa, claude.roux}@xrce.xerox.com

Abstract

In this paper, we present the system we have developed for the SemEval-2014 Task 4 dedicated to Aspect-Based Sentiment Analysis. The system is based on a robust parser that provides information to feed different classifiers with linguistic features dedicated to aspect categories and aspect categories polarity classification. We mainly present the work which has been done on the restaurant domain[1] for the four subtasks, aspect term and category detection and aspect term and category polarity.

1 Introduction

Aspect Based Sentiment Analysis aims at discovering the opinions or sentiments expressed by a user on the different aspects of a given entity ((Hu and Liu, 2004); (Liu, 2012)). A wide range of methods and techniques have been proposed to address this task, among which systems that use syntactic dependencies to link source and target of the opinion, such as in (Kim and Hovy, 2004), (Bloom et al., 2007), or (Wu et al., 2009). We have developed a system that belongs to this family, (Brun, 2011), as we believe that syntactic processing of complex phenomena (negation, comparison, ...) is a crucial step to perform aspect-based opinion mining. In this paper, we describe the adaptations we have made to this system for SemEval, and the way it is applied to category and polarity classification.

2 Description of the System

In this section, we describe the different components of the system.

2.1 Existing System

In order to tackle the Semeval'14 Task 4, (Pontiki et al., 2014), we used our existing aspect-based opinion detection system. The opinion detection system we built relies on a robust deep syntactic parser, (Ait-Mokhtar et al., 2001), as a fundamental component, from which semantic relations of opinion are calculated. Parsing here includes tokenization, morpho-syntactic analysis, tagging which is performed via a combination of hand-written rules and HMM, Named Entity Detection, chunking and finally, extraction of dependency relations between lexical nodes. These relations are labeled with deep syntactic functions. More precisely, a predicate (verbal or nominal) is linked with what we call its deep subject (SUBJ-N), its deep object (OBJ-N), and modifiers. In addition, the parser calculates more sophisticated and complex relations using derivational morphologic properties, deep syntactic properties (subject and object of infinitives in the context of control verbs), and some limited lexical semantic coding.

Syntactic relations already extracted by a general dependency grammar, lexical information about word polarities, sub categorization information and syntactic dependencies are all combined within our robust parser to extract the semantic relations. The polarity lexicon has been built using existing resources and also by applying classification techniques over large corpora, while the semantic extraction rules are handcrafted, see (Brun, 2011) and (Brun, 2012) for the complete description of these different components. The system outputs a semantic dependency called SENTIMENT which can be binary, i.e. linking opinionated terms and their targets, or unary, i.e. just the polar term in case the target of the

[1] We have not performed any domain adapation for the laptop corpus and only submitted a run for the subtask 1, term detection.

Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), pages 838–842,
Dublin, Ireland, August 23-24, 2014.

opinion hasn't been detected. For example, when parsing *I was highly disappointed by their service and food.*, the systems outputs the following dependencies:

SUBJ_N(disappointed,food)
SUBJ_N(disappointed,service)
OBJ_N(disappointed,I)
MANNER_PRE(disappointed,highly)
SENTIMENT_NEGATIVE(disappointed,service)
SENTIMENT_NEGATIVE(disappointed,food)

In this system, aspects terms are not explicitly extracted, however all non-polar arguments of the SENTIMENT dependency are potential aspect terms. Moreover, this system considers only positive and negative opinions, but does not cover the neutral and conflict polarities.

2.2 System Adaptation

The opinion detection system described in the previous section has been adapted for the SemEval2014 Task4, in two ways: some lexical acquisition has been performed in order to detect the terms of the domain, and some rules have been developed to detect multi-word terms and to output semantic dependencies associating their polarity to terms and categories.

2.2.1 Lexical Enrichment and Term Detection

As said before, the existing system encodes a reasonable amount of polar vocabulary. However, as the task implies domain knowledge to detect the terms, we have first extracted the terms from the training corpus and encoded their words into our lexicons, assigning to them the semantic features food, service, ambiance and price. We have then extended the list with Wordnet synonyms. To improve coverage, we have also extracted and filtered food term lists from Wikipedia pages and encoded them. More precisely, the list of food terms has been extracted from the Wikipedia "Food Portal", from the category "Lists_of_foods"[2]. At the end of this process, our lexicon has the following coverage: Polar words: 1265 negative, 1082 positive and Domain words: 761 food words, 31 price words, 105 ambiance words, 42 service words.

In order to detect the terms, some local grammar rules (based on regular expressions) have been developed taking into account the lexical semantic information encoded in the previous step. These rules detect the multi-words terms, e.g. *pastrami sandwiches*, group them under the appropriate syntactic category (noun, verb) and associate them with the corresponding lexical semantic feature,*food*, *service*, *ambiance*, *price*. In addition to this, in order to prepare the aspect category classification (c.f. section 2.3.3), a layer of semantic dependencies has been added to the grammar: If a domain term is detected in a sentence, a unary dependency corresponding to its category (FOOD, SERVICE, PRICE, AMBIANCE) is built.

2.2.2 Grammar Adaptation for Polarity Detection

The English grammar, which had been previously developed to detect sentiments, has also been adapted in order to extract the opinions associated to the terms and categories detected at the previous step.

If an aspect term is the second argument of a SENTIMENT relation, 2 dependencies, one for the term (OPINION_ON_TERM) and one for the corresponding category (OPINION_ON_CATEGORY) are built. They inherit the polarity (positive or negative) of the SENTIMENT dependency. If these dependencies target the same term and category and if they have opposite polarity, they are modified in order to bear the feature "conflict".

Then, if a sentence contains a term and if no SENTIMENT dependency has been detected, the OPINION_ON_TERM and OPINION_ON_CATEGORY are created with the polarity "neutral". Finally, if no terms have been detected in a sentence, there are two cases: (1) a SENTIMENT dependency has been detected somewhere in the sentence, the dependency OPINION_ON_CATEGORY(anecdote/misc), is created with the corresponding polarity (positive or negative); (2) no SENTIMENT dependency has been detected, the dependency OPINION_ON_CATEGORY(anecdote/misc), is created with polarity "neutral".

The dependency OPINION_ON_TERM links the terms to their polarities in the sentences and serves as input for the subtasks 1 and 3.

2.3 Classification

2.3.1 KiF (*Knowledge in Frame*)

The whole system, training and prediction, has been implemented in KiF (*Knowledge in Frame*), a script language that has been implemented into

[2] http://en.wikipedia.org/wiki/Category:Lists_of_foods

the very fabric of the rule-based Xerox Incremental Parser (XIP). KiF offers a very simple way to hybridize a rule-based parser with machine learning technique. For instance, a KiF function, which evaluates a set a features to predict a class, can be called from a rule, which could then be fired along the output of that function. KiF is a multi-threaded programming language, which is available for all platforms (Windows, Mac OS, Linux). It provides all the necessary objects (strings, containers or classes) and many encapsulations of dynamic libraries from different C programs such as classifiers (liblinear and libsvm), database (SQLite), or XML (libxml2), which can be loaded on the fly. All internal XIP linguistic structures are wrapped up into KiF objects. For example, linguistic features are available as maps, which can be modified and re-injected into their own syntactic nodes. The language syntax is a mix between Java (types are static) and Python (in the way containers are handled), but provides many implicit conversions to avoid code overloading with too many functions. KiF allows for an efficient integration of all aspects of linguistic analysis into a very simple framework, where XML documents can be analyzed and modified both with linguistic parsing and classifiers into a few hundred lines of code.

2.3.2 General Methodology

We focus on four main tasks: detecting the aspect terms and aspect categories and their corresponding polarities. While the detection of aspect terms and their corresponding polarities occurs at the grammar level, for the detection of aspect categories and their corresponding polarities we make use of the liblinear library (Fan et al., 2008) to train our models. We train one classifier for detecting the categories and further, for each category we train a separate classifier for detecting the polarities corresponding to that particular category. For both settings, we use 10-fold cross-validation. The two modules for aspect category classification and aspect category polarity classification are described in details further.

2.3.3 Aspect Category Classification

The sentence classification module is used to assign aspect categories to sentences. For each sentence, the module takes as input features the bag of words in the sentence as well as the information provided by the syntactic parser. The output consists of a list of categories corresponding to each sentence.

In the pre-processing stage stop words are removed (determinants, conjunctions). Further, we use the L2-regularized logistic regression solver from the liblinear library to train a model. The features considered are the word lemmas from the sentence along with their frequencies (term frequency). Apart from this, the information provided by the rule based component is also taken into account to increase the term frequency for terms belonging to the detected categories.

Such information can consist of: dependencies denoting the category to which a detected aspect term belongs (Food, Service, Price, Ambiance) and dependencies denoting the opinions on the detected aspect terms and categories (OPINION_ON_CATEGORY, OPINION_ON_TERM). For example for the following sentence: *"Fabulous service, fantastic food, and a chilled out atmosphere and environment"*, the salient dependencies produced by the syntactic parser are:

FOOD(food), AMBIANCE(atmosphere),
SERVICE(service), AMBIANCE(environment),
OPINION_ON_CATEGORY_POSITIVE(food),
OPINION_ON_CATEGORY_POSITIVE(service),
OPINION_ON_CATEGORY_POSITIVE(ambiance),
OPINION_ON_TERM_POSITIVE(food),
OPINION_ON_TERM_POSITIVE(service),
OPINION_ON_TERM_POSITIVE(atmosphere).

This yields the following features having an increase in their frequencies: food (+3), service (+3), atmosphere (+2), environment (+1), ambiance (+1).

Once the logistic regression is performed, each category is predicted with a certain probability. Since in one sentence there may be entities that refer to different categories, we set a threshold with respect to the probability values to be taken into account. We have tried different approaches to set this threshold. The best results on the training and trial data were obtained with a threshold of 0.25, (i.e. we kept only the categories with a probability over 0.25).

2.3.4 Aspect Category Polarity Classification

The approach to predict the polarity for each category is similar to the one predicting the categories for each sentence, with some differences as will be further detailed. The classification uses for features, the bag of words (term frequency), but also the polarity provided by XIP by the following dependencies: OPINION_ON_CATEGORY and SENTIMENT. Whenever these dependencies are detected, a feature is added to the classification of the form *polarity_category*. Thus for the previous example sentence: *Fabulous service, fantastic food, and a chilled out atmosphere and environment*, the additional dependencies considered are SENTIMENT_POSITIVE(atmosphere, chilled out), SENTIMENT_POSITIVE(food, fantastic), SENTIMENT_POSITIVE(service, Fabulous). After mapping back the terms to their corresponding categories, the added features are: *positive_ambiance*, *positive_food* and *positive_service*. Since the dependency OPINION_ON_CATEGORY is also detected by the parser for these categories, each of the above mentioned features will have a frequency of 2 in this case. Moreover, the polarity alone is also added as a feature. The training is performed using the L2-regularized L2-loss support vector classification solver from the same library (liblinear) and a model is generated for each category. Thus, depending on the categories detected within a certain sentence, the corresponding model is used to make the prediction regarding their polarities. The classifier's output represents the predicted polarity for one given category.

3 Evaluation

The corpus used for evaluating the system contains 800 sentences, 1134 aspect term occurrences, 1025 aspect category occurrences, 5 different aspect categories and 555 distinct aspect terms. All these belong to the restaurant domain.

3.1 Terms and Category Detection

When evaluating aspect terms and aspect categories detection, three measures were taken into account: precision, recall and the f1-measure.

For both aspect term extraction and aspect category detection, the baseline methodologies are presented in (Pontiki et al., 2014). Table 1 shows the results obtained using our approach as compared to the baseline for aspect term detection, whereas Table 2 outlines the results regarding aspect category detection in terms of the previously mentioned measures.

Furthermore, it is interesting to notice the increase in performance obtained by combining the bag-of-words features with the output of the parser as opposed to just using the bag-of words. These differences are outlined for aspect category detection in Table 2, where BOW denotes the system using the same settings, but just the bag-of-words features and XRCE denotes the submitted system where the bag-of-words features are augmented with parser output features.

Method	**Precision**	**Recall**	**F-Measure**
Baseline	0.627329	0.376866	0.470862
XRCE	0.862453	0.818342	0.839818

Table 1: Aspect term detection.

Method	**Precision**	**Recall**	**F-Measure**
Baseline	0.637500	0.483412	0.549865
BOW	0.77337	0.799024	0.785988
XRCE	0.832335	0.813658	0.822890

Table 2: Aspect category detection.

For both tasks of aspect term and aspect category detection, our system clearly outperforms the baseline, resulting in being ranked among the first 3 in the competition for the restaurant corpus.

3.2 Terms and Category Polarity Detection

Similarly, Table 3 shows the results in terms of accuracy on aspect term polarity detection and on aspect category polarity detection. Here, baseline methodologies are similar to the ones used for aspect category detection and also described in (Pontiki et al., 2014). Again, our system ranks high in the competition, achieving an overall accuracy of 0.77 for aspect term polarity detection and 0.78 for aspect category polarity detection. Furthermore, a comparison is also made between the current system and one that, using the same settings, would not take into account the features provided by the parser (BOW). The results emphasize the importance of using the merged version.

Method	**Task**	**Accuracy**
Baseline	Term polarity	0.552239
XRCE	Term polarity	0.776895
Baseline	Category polarity	0.563981
BOW	Category polarity	0.681951
XRCE	Category polarity	0.781463

Table 3: Aspect term and aspect category polarity.

Label	Precision	Recall	F-measure
conflict	NaN	0	NaN
negative	0.7857	0.7296	0.7566
neutral	0.5833	0.3214	0.4145
positive	0.7998	0.9272	0.8588

Table 4: Aspect term polarity (2).

Label	Precision	Recall	F-measure
conflict	0.5333	0.1538	0.2388
negative	0.726	0.6802	0.7023
neutral	0.5119	0.4574	0.4831
positive	0.8343	0.9117	0.8713

Table 5: Aspect category polarity (2).

3.3 Error Analysis

The results obtained with our system are unarguably competitive, but some remarks can be made regarding the most frequent causes of errors. In the task of aspect category classification, the choice of the threshold (0.25) may have constituted a factor impacting the performance. In the task of aspect term detection, the lexical coverage is one of the factors to explain the difference in performance between training/trial data and test data.

Table 4 contains the results obtained in terms of precision, recall and F-measure for each of the possible polarities for terms (positive, negative, neutral and conflict) and similarly does Table 5 for category polarities. In both cases we notice a clear decrease for these measures when predicting the *conflict* and *neutral* classes, with a higher decrease in the case of aspect term polarity detection. This can be explained by the fact that the syntactic parser was primarily customized to detect the *negative* and *positive* labels. This obviously had an impact on the final results as the information from the parser constituted some of the input features for the classification.

4 Conclusion

The combination of a symbolic parser, customized with specialized lexicons, with SVM classifiers proved to be an interesting platform to implement a category/polarity detection system. The symbolic parser on the one hand provides a versatile architecture to add lexical and multi-words information, augmented with specific rules, in order to feed classifiers with high quality features. However, some work will be needed to improve performances on the neutral and conflict polarities, which rely less on specific words, than on a more global interpretation of the content.

Acknowledgements

We would like to thank the Semeval task 4 organizers, as well as our colleague, Vassilina Nikoulina, for her help on this project.

References

Salah Ait-Mokhtar, Jean-Pierre Chanod, and Claude Roux. 2001. A multi-input dependency parser. In *IWPT*.

Kenneth Bloom, Navendu Garg, and Shlomo Argamon. 2007. Extracting appraisal expressions. In *In HLT-NAACL 2007*, pages 308–315.

Caroline Brun. 2011. Detecting opinions using deep syntactic analysis. In *RANLP*, pages 392–398.

Caroline Brun. 2012. Learning opinionated patterns for contextual opinion detection. In *COLING*, pages 165–174.

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.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *KDD*, pages 168–177.

Soo-Min Kim and Eduard Hovy. 2004. Determining the sentiment of opinions. In *Proceedings of the 20th International Conference on Computational Linguistics*, COLING '04, Stroudsburg, PA, USA.

Bing Liu. 2012. *Sentiment Analysis and Opinion Mining*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

Maria Pontiki, Dimitrios Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *International Workshop on Semantic Evaluation (SemEval)*.

Yuanbin Wu, Qi Zhang, Xuanjing Huang, and Lide Wu. 2009. Phrase dependency parsing for opinion mining. In *EMNLP*, pages 1533–1541. ACL.