TOES: A Taxonomy-Based Opinion Extraction System

Fermín L. Cruz, José A. Troyano, F. Javier Ortega, and Fernando Enríquez

University of Seville
Av. Reina Mercedes s/n 41012, Sevilla Spain
{fcruz,troyano,javierortega,fenros}@us.es

Abstract. Feature-based opinion extraction is a task related to opinion mining and information extraction which consists of automatically extracting feature-level representations of opinions from subjective texts. In the last years, some researchers have proposed domain-independent solutions to this task. Most of them identify the feature being reviewed by a set of words from the text. Rather than that, we propose a domain-adaptable opinion extraction system based on feature taxonomies (a semantic representation of the opinable parts and attributes of an object) which extracts feature-level opinions and maps them into the taxonomy. The opinions thus obtained can be easily aggregated for summarization and visualization. In order to increase precision and recall of the extraction system, we define a set of domain-specific resources which capture valuable knowledge about how people express opinions on each feature from the taxonomy for a given domain. These resources are automatically induced from a set of annotated documents. The modular design of our architecture allows building either domain-specific or domain-independent opinion extraction systems. According to some experimental results, using the domain-specific resources leads to far better precision and recall, at the expense of some manual effort.

1 Introduction

Sentiment analysis is a modern subdiscipline of NLP which deals with subjectivity, affects and opinions in texts (a good survey on this subject can be found in [5]). Within sentiment analysis, the *feature-based opinion extraction* is a task related to information extraction, which consists in extracting structured representations of opinions on features of some object from subjective texts [3,6,2]. For example, given the sentence *"The customer service is terrible"*, a negative opinion on feature *customer service* should be extracted. Some researchers have proposed several approaches to this task, often unsupervised, domain-independent ones. In most cases, they select a few words from the sentence representing the feature affected by the opinion (*opinion target* or *feature words*, depending on authors). This approach implies some problems. First, sometimes the same feature can be named in different ways. For example, *customer service* is also known as *helpline* or *help desk* in some contexts. So a further matching problem must be solved in order to be able to aggregate opinions on the same feature. Besides, some features may include others; for example, someone looking for opinions about the *sound quality* of an audio system would be interested not only in those sentences explicitly referring to the sound quality (e.g., "The *sound quality* is superb", "Very clean, outstanding *sound*"), but also in sentences talking about some other related features (e.g., "The *low end* is clear and the *high* is twangy"). Dealing with these issues is important in order to properly aggregate the extracted opinions and exploit the whole amount of available information.

2 Our Approach

The main guidelines of our approach are (1) building a feature taxonomy for each new domain, so our system will extract opinions on those features and map them into the taxonomy, and (2) automatically generating domain-specific, feature-level resources which capture valuable knowledge about how people express opinions on each feature for a given domain. These resources lead to a higher quality opinion extraction, at the expense of a small manual effort to annotate some documents from the selected domain.

2.1 Feature Taxonomy

The feature taxonomy contains the set of features for which opinions will be extracted in a given domain. Besides, it contains a set of feature words for each feature. All these pairs (*feature*,*feature words*) are hierarchically organized: the object class itself is the root node of the taxonomy, with a set of features hanging on it. Each feature can be recursively decomposed into a set of subfeatures (see figure 1). The taxonomy hierarchy is useful to aggregate opinions to produce summaries.

The feature taxonomy is built in two steps. First, a list of feature words is generated from the corpus using an active-learning method. Then, an expert pro-

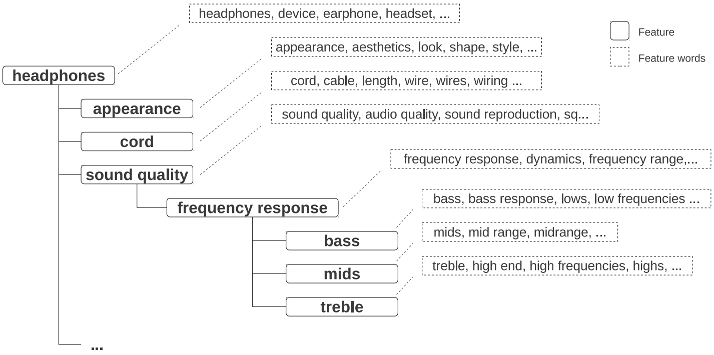
headphones, device, earphone, headset, ...
Feature
Feature words
appearance, aesthetics, look, shape, style, ...
headphones
cord, cable, length, wire, wires, wiring ...
appearance
sound quality, audio quality, sound reproduction, sq...
cord
frequency response, dynamics, frequency range,...
sound quality
bass, bass response, lows, low frequencies ...
frequency response
mids, mid range, midrange, ...
bass
treble, high end, high frequencies, highs, ...
mids
treble
...

Fig. 1. An extract from the feature taxonomy for the *headphones* domain

duces the taxonomy, grouping feature words by feature and building a hierarchy. The whole process takes no more than a few minutes.

2.2 Domain-Specific Resources

A distinctive part of our approach is the definition of resources that capture knowledge about domains and the way people write reviews on them. To generate these resources, we start from a manual effort (although computer assisted) in order to describe a feature taxonomy and annotate opinions in a set of documents. Then we apply some algorithms in order to extract relevant information about key concepts of the annotated opinions. The resources include, between others, dependency patterns linking feature words and opinion words, opinion lexicons containing semantic orientation estimations for the opinion words more commonly used in the domain, and lists of lexical indicators to detect implicit features[1].

2.3 System Architecture

Our opinion extraction system is comprised of a set of independent abstract components, each one dealing with a different subtask. They can be combined in a wide variety of pipelines in order to complete the extraction task. This modular design together with the multiple implementations of each component make up an experimental setup that enables us to test different approaches.

Let us give a brief description of some of these components. The *feature word annotators* discover features explicitly mentioned in the input reviews. The *implicit feature annotators* discover implicitly mentioned features. Given some previously annotated feature words, the *opinion word linkers* intend to link them to related opinion words. The *opinion classifiers* decide if a previously annotated opinion is a positive or a negative one.

3 Experimentation

Some experiments were performed over a corpus of 587 reviews of headphones from *Epinions.com*. A feature taxonomy was built and the opinions appearing in the documents were annotated[2]. All the experiments were done using 10-fold cross-validation. We evaluated two subproblems: given a sentence, *opinion recognition* consists in identifying the existence of opinions, including determining the feature that opinion refers to; *opinion classification* consists in deciding the polarity of previously recognized opinions. We tested four different approaches (see table 1). In the first three experiments, domain-independent pipelines were

[1] A feature is implicit if it is not explicitly mentioned in the text.

[2] The annotated dataset is available in http://www.lsi.us.es/~fermin/index.php/Datasets, including three different domains: headphones, hotels and cars.

Table 1. Experimental results for *headphones* domain

				Opinion
	Opinion Recognition			Classification
Experiment	p	r	$F_{\frac{1}{2}}$	accuracy
PMI-IR	0,6092	0,3039	0,5073	0,8706
WordNet	0,6756	0,3002	0,5405	0,8940
SentiWordnet	0,6744	0,3643	0,5763	0,8688
Resource-based	0,7869	0,5662	0,7300	0,9503

used. They all employ a window-based opinion word linker, and classify opinions using three different techniques from literature: the PMI-IR algorithm [7], an algorithm based on lexical distances in WordNet [4] and a state-of-art domain-independent opinion lexicon named SentiWordNet [1]. The fourth experiments were done using a domain-specific pipeline whose components make use of the domain-specific resources. The results obtained by the latter pipeline are far better than those obtained by the domain-independent pipelines. We also conducted some experiments to measure the impact of the number of annotated documents in the results; we found that just a few hours of annotation are enough to largely overcome the results obtained by the resource-free pipelines.

References

1. Baccianella, S., Esuli, A., Sebastiani, F.: Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In: Proceedings of the Seventh Conference on International Language Resources and Evaluation, LREC 2010. European Language Resources Association (ELRA), Valletta (2010)
2. Ding, X., Liu, B., Yu, P.S.: A holistic lexicon-based approach to opinion mining. In: WSDM 2008: Proceedings of the International Conference on Web search and Web Data Mining, pp. 231–240. ACM, New York (2008)
3. Hu, M., Liu, B.: Mining and summarizing customer reviews. In: Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), pp. 168–177 (2004)
4. Kamps, J., Marx, M., Mokken, R.J., De Rijke, M.: Using wordnet to measure semantic orientation of adjectives. National Institute 26, 1115–1118 (2004)
5. Pang, B., Lee, L.: Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval 2(1-2), 1–135 (2008)
6. Popescu, A.-M., Etzioni, O.: Extracting product features and opinions from reviews. In: Proceedings of the Human Language Technology Conference and the Conference on Empirical Methods in Natural Language Processing, HLT/EMNLP (2005)
7. Turney, P.: Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In: Proceedings of the Association for Computational Linguistics (ACL), pp. 417–424 (2002)