

Usage of Abstract Features in Semantic Sentiment Analysis

Mohammed Almashraee¹, Kia Teymourian¹, Dagmar Monett-Diaz², and
Adrian Paschke¹

¹ Freie Universität Berlin, Corporate Semantic Web Working Group

² Department of Computer Science, Berlin School of Economics and Law
{almashraee,kia,paschke}@inf.fu-berlin.de
dagmar.monett-diaz@hwr-berlin.de

Abstract. Feature-based sentiment analysis can be realized on different types of object features. Some of these features might be about technical aspects of the objects and some others might be application-oriented features. The application-oriented features are more abstract features and can be of interest to the broad number of people than only the technical experts of specific products. In this paper, we propose an approach for extraction of abstract object features from a set of sub-features. In our approach we use a knowledge base about the application domain to extract related sub-features of high-level abstract features. On the basis of such related sub-features our approach performs the extraction of more abstract features that are only implicitly included in the analysis text.

1 Introduction

People express their opinions about certain objects using some of their features. For example in the photography application domain, the users of digital cameras express their opinions about the features of cameras like *flash* or *lens*. Consumers make use of the expressed opinions to know about the quality of an object and the quality of some of its aspects so that they can make the right purchase decisions [1,2].

The sentiment analysis result of technical-oriented features like *flash*, *lens*, *optical zoom*, *shutter*, *sensor quality* are interesting for professional photographers who are familiar with the technical details of cameras and know which one of the features are important for which kind of photography modes. Contrary to that, non-expert users are interested in features which are more abstract and are application-oriented. For example non-experts are interested to know if the camera can make good pictures of kids or if the camera can take pictures of landscape during their vacations. Such implicit high-level abstract features are mostly not explicitly mentioned in the review corpus or are only implicitly mentioned in some of the review items. The main difference between abstract features and sub-features is that abstract features are non-technical aspects which are rarely available among related reviews, while sub-features are technical aspects which can be found frequently in the review corpus.

In this paper, we propose an approach for the semantic sentiment analysis of abstract features based on the related sub-features. The abstract features can be derived from the explicitly mentioned sub-features that are related to

the abstract feature. The system in general performs sentiment analysis on the review text on the basis of an extracted set of sub-features. Table 1 shows some examples of such abstract features and the related sub-features.

Table 1. Examples of Abstract- and Sub-Features of Digital Cameras

Abstract-Features	Related Sub-Features
Night photo	Flash, Lens, Image Processor, Sensors
Portrait	Optical Zoom, Lens, Image Processor
Sports	Shutter, Image Processor, Flash, Sensors
Landscape	Optical Zoom, Shutter, Flash, Sensors
Kids Photography	Shutter, Image Processor, Sensors, Flash

As an example consider the following of review text³: *“It automatically selects the **best shooting settings** for optimal quality based on the environmental factors (**lightning** I guess) to provide point’n’ shoot **simplicity**. **16.0 Megapixels**, with loads of resolution pictures are **still clear**. **High resolution** is also **good for producing biggest printouts**. **5x Optical Zoom** is **sufficient** in most cases. **DIGIC 4 Image Processor** is not as fast as **DIGIC 5** though **fast and powerful enough** to give you advanced system options, provide **quick-shoot with reliable performance** and **low battery consumption**. As far as I know **DIGIC 4** is currently **Canon’s most efficient processor** for **budget cameras**. BTW it has some **Eco mode**, that is said to be providing even **faster warm-up times** and **saves the standard battery**, but I haven’t tested it yet. **Very lightweight**, just put it into your **pocket**, can take it everywhere. Like A2300 it **lacks optical image stabilization**, though it’s got digital image stabilization. **1/2.3” sensor**, well, **entry level CCD** providing good pictures, **not of a DSLR quality**, that’s all I can say.”*

2 Abstract Features in Semantic Sentiment Analysis

Our approach consist of the following processing tasks:

1. *Feature Extraction*: In this task the related sub-features are identified.
2. *Knowledge-based Feature Annotation*: By using a knowledge-based annotator the sub-features can be annotated with their background knowledge resources.
3. *Feature Preparation*: The background knowledge for each annotated resource is retrieved from knowledge base and enriched to them.
4. *Sentiment Relation Calculation*: Based on the specified relations of sub-features to the abstract features in a background knowledge base the sentiment of abstract feature are calculated.

As a general conceptual solution, we propose to parse the text to collect features, names, name phrases and other parts which constitute the features. We split each review into sentences and then parse each sentence to extract

³ Review Example from Amazon Online Store
<http://www.amazon.com/Canon-PowerShot-A2500-Stabilized-2-7-Inch/dp/B00B5HE2UG/>

the feature(s) it contains. For knowledge-based feature extraction we propose to use a knowledge-based feature annotation that can recognize names of concepts or entities have been mentioned in the text. Using knowledge-based resource annotation systems like DBpedia Spotlight⁴ or AlchemyAPI⁵ it is possible to collect the target features from the review text. Such entity annotation system can be used with a knowledge base specially made for the application domain.

The knowledge-based feature annotation and feature preparation system can extract from the given example features like: “*best shooting settings*”, “*lightning*”, “*good result*”, “*shoot simplicity*”, “*16.0 Megapixels*”, “*DIGIC 4 Image Processor*” and “*faster warm-up times*”.

Annotation is a task of adding more information to an existing object like text, image and video. The major advantage of using semantic annotation is that we can relate the entities to their knowledge base resources so that we can extract background knowledge about them. As a general conceptual solution, the set of extracted features from the feature extraction task is enriched and extended using entity recognition and ontological reasoning. The feature enrichment process is realized using a knowledge-based annotator. The examples of such features and their knowledge base types are shown in Table 2.

Table 2. Examples of Features and their Knowledge Base Types

Enriched Features	Knowledge Base Types
Shoot Setting	camera-onto:Camera_Setting
Megapixels	camera-onto:Image_Quality
Eco mode	camera-onto:Camera_Shooting_Mode
DIGIC 4	camera-onto:Image_Processor
DIGIC 5	camera-onto:Image_Processor

We propose to start with a set of ontological relationships that can be used to extract further knowledge resources like equivalence, direct hypernyms and direct hyponyms. This list can be extended with additional relationships depending on the structure of the ontology in use and on its granularity. The sentiment value of each resource can be computed based on the sentiment of the related sub-feature. We propose the following correspondences for the set of ontological relationships we are considering:

1. equivalence = the same sentiment value is given to the sub-feature
2. hyperonymy = a factor to be applied to the sentiment value of sub-feature
3. hyponymy = a factor to be applied to the sentiment value of sub-feature

These factors should be specified manually in the ontology by the domain experts who are familiar with the technical details and the relations of sub-features to the target abstract features. The ontology should include the required knowledge about the application domain, e.g., in our example it should conceptualize the concept of camera and the world of photography so that one can extract the related concepts, e.g., for the “*Night Photography*”.

As an example for the sentiment calculation for abstract features, we consider the calculation for the *night photography* and *kids photography*. By inferencing

⁴ <http://spotlight.dbpedia.org/>

⁵ <http://www.alchemyapi.com>

on an ontology about the relations of the sub-features to each other and to the abstract features, we can calculate different effecting factors that can be used for the sentiment calculation for the abstract features. We process the sentiment for each of the related sub-features for the complete corpus of the product. For example we use the following extracted calculation for sentiment of the abstract features “*Night Photography*” and “*Kids Photography*”.

$$S_{NightPhotography} = 0.1*S_{Flash} + 0.3*S_{Lens} + 0.4*S_{ImageProcessor} + 0.2*S_{Sensors}$$

$$S_{KidsPhotography} = 0.1*S_{Shutter} + 0.5*S_{ImageProcessor} + 0.3*S_{Sensors} + 0.1*S_{Flash}$$

In the above example the sentiment factors of sub-features are extracted by using an ontology that include the relations between abstract features and sub-features. Our approach is highly depending on the existence of an ontology that can describe the relations between the features and can be used for inferencing on feature relations.

3 Conclusion and Future Work

Our main research question in this research is “*To which extent is it possible to use ontological background knowledge to derive abstract upper-level features based on more technical sub-features*”? To answer this question, we structure the solution into three main tasks and from each task we tackle a number of sub-tasks. The first task extracts features from reviews using tools from natural language processing. The second task extends the number of collected features based on entity recognition and ontological reasoning. The third task finds relations between features and maps sub-features into related abstract features.

We are still working on the third task to collect as many features as possible relevant to our domain and then to find relations among the collected features and to associate sub-features to their related abstract-features.

In this paper we proposed the relating and inferring more general features from a given set of more specific ones. We first extract the available features and then extend the list of extracted features using a related domain ontology. We then analyze the collected features and find relations among them. The relations are further realized in terms of real weight values.

Our future work is to specify more details of the usage of background domain ontology for the extraction of the relevant features, e.g., the reasoning on the background knowledge bases can help to understand more about the features that are not explicitly connected to the abstract features in the ontology. We also need to find methods to relate and evaluate specific features to more abstract ones. Furthermore, we have to evaluate our approach on a large corpus while using a large enough domain ontology.

References

1. S. Kim, J. Zhang, Z. Chen, A. H. Oh, and S. Liu, “A hierarchical aspect-sentiment model for online reviews,” in *AAAI*, 2013.
2. Y. Jo and A. H. Oh, “Aspect and sentiment unification model for online review analysis,” in *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*, WSDM ’11, (New York, NY, USA), pp. 815–824, ACM, 2011.