Procesamiento del Lenguaje Natural, Revista nº 42, marzo de 2009, pp. 47-54 recibido 13-01-09, aceptado 02-03-09

A Semantic Relatedness Approach to Classifying Opinion from Web Reviews

Un método de clasificación de opiniones de críticas extraídas de la Web basado en la proximidad semántica

Alexandra Balahur
DLSI, University of Alicante
Ap. de Correos 99, 03080 Alicante
abalahur@dlsi.ua.es

Andrés Montoyo
DLSI, University of Alicante
Ap. de Correos 99, 03080 Alicante
montoyo@dlsi.ua.es

Resumen: Los últimos años han marcado el inicio y la rápida expansión de la web social, donde cada persona puede expresar su libre opinión sobre diferentes "objetos", tales como productos, personas, tópicos de política etc. en blogs, foros o portales Web de comercio electrónico. A su vez, el rápido crecimiento del volumen de información en la web ha ido permitiendo a los usuarios la toma de decisiones mejores y más informadas. A raíz de esta expansión ha surgido la necesidad de desarrollar sistemas especializados de PLN que automáticamente escaneen la web en busca de las opiniones expuestas (que recuperen, extraigan y clasifiquen las opiniones existentes dada una consulta). La minería de opiniones (análisis de sentimientos) ha demostrado ser un problema difícil debido a la gran variabilidad semántica del texto libre. En este artículo se propone un método para extraer, clasificar y resumir opiniones sobre productos concretos utilizando críticas realizadas en la Web. El método se basa en una taxonomía de características de productos previamente construida, el cálculo de la proximidad semántica entre conceptos por medio de la Distancia Normalizada de Google y el método de aprendizaje automático SVM. Finalmente, demostramos que nuestro enfoque supera los resultados base de la tarea y ofrece una alta precisión y una alta confianza en las clasificaciones obtenidas.
Palabras clave: Minería de opiniones, resúmenes automaticos, Distancia Normalizada de Google, aprendizaje automatico SVM.

Abstract: Recent years have marked the beginning and rapid expansion of the social web, where people can freely express their opinion on different “objects”, such as products, persons, topics etc. on blogs, forums or e-commerce sites. While the rapid growth of the information volume on the web allowed for better and more informed decisions from users, its expansion led to the need to develop specialized NLP systems that automatically mine the web for opinions (retrieve, extract and classify opinions of a query object). Opinion mining (sentiment analysis) has been proven to be a difficult problem, due to the large semantic variability of free text. In this article, we propose a method to extract, classify and summarize opinions on products from web reviews, based on the prior building of product characteristics taxonomy and on the semantic relatedness given by the Normalized Google Distance and SVM learning. We prove that our approach outperforms the baselines and has a high precision and classification confidence.
Keywords: Opinion mining, summarization, Normalized Google Distance, SVM machine learning.

1 Introduction

Recent years have marked the strong influence of the “participative, social web” on the lives of both consumers and producer companies. This phenomenon encouraged the development of specialized sites – blogs, forums, as well as the inclusion of a review component in the already existing e-commerce sites, where people can write and read opinions and comments on their “objects” of interest – products, people, topics,

ISSN 1135-5948

etc. Basically, one is able to obtain a high volume of data representing opinion on anything. However, a high volume of information introduces a great back draw: the time spent for reading all the data available and the language barrier. The solution is obvious - a system that automatically analyzes and extracts the values of the features for a given product, independent of the language the customer review is written in. Such an NLP system can then present the potential buyer with percentages of positive and negative opinions expressed about each of the product features and possibly make suggestions based on buyer preferences. What follows is a description of such a system that presently works on Spanish and English.

2 Motivation and Contribution

In the approach proposed, we concentrated on two main problems that had not been addressed so far by research in the field. The first one was that of discovering the features that will be quantified. As previously noticed in (Liu, 2007), features are implicit or explicit. To this respect, apart from a general class of features (and their corresponding attributes), that are applicable to all products, we propose a method to discover product specific features and feature attributes using knowledge from WordNet and ConceptNet. The second problem we addressed was that of quantifying the features in a product-dependent manner, since, for example, small for the size of a digital camera is a positive fact, whereas for an LCD display it is a rather negative one. We accomplished this by classifying the feature attributes using positive and negative examples from a corpus of customer opinions that was polarity annotated depending on the product category and SMO SVM machine learning (Platt, 1998) with the Normalized Google Distance (Cilibrasi and Vitanyi, 2006). We will illustrate the manner in which we solved the above mentioned problems with examples and discuss on the issues raised at each step by using different methods, tools and resources.

3 Related Work

Previous work in customer review classification includes document level sentiment classification using unsupervised methods (Turney, 2002), machine learning techniques (Pang, Lee and Vaithyanathan, 2002), scoring of features (Dave, Lawrence and Pennock, 2003) , using PMI, syntactic relations and other attributes with SVM (Mullen and Collier, 2004), sentiment classification considering rating scales (Pang, Lee and Vaithyanathan, 2002), supervised and unsupervised methods (Chaovalit and Zhou, 2005) and semisupervised learning (Goldberg and Zhu, 2006). Research in classification at a document level included sentiment classification of reviews (Ng, Dasgupta and Arifin, 2006), sentiment classification on customer feedback data (Gamon et al., 2005), comparative experiments (Cui, Mittal and Datar, 2006). Other research has been conducted in analysing sentiment at a sentence level using bootstrapping techniques (Riloff and Wiebe, 2003), considering gradable adjectives (Hatzivassiloglou and Wiebe, 2000), semisupervised learning with the initial training set identified by some strong patterns and then applying NB or self-training (Wiebe and Riloff, 2005), finding strength of opinions (Wilson, Wiebe and Hwa, 2004) sum up orientations of opinion words in a sentence (or within some word window) (Kim and Hovy, 2004), (Lin et al., 2006), determining the semantic orientation of words and phrases (Tuney and Littman, 2003), identifying opinion holders (Stoyanov and Cardie, 2006), comparative sentence and relation extraction and feature-based opinion mining and summarization (Tuney, 2002). The approach we use is grounded on the feature-based opinion summarization paradigm, whose theoretical background can be found in (Hu and Liu, 2004) and (Liu, 2007). Relevant research done in feature-based opinion summarization can be found in (Turney, 2002) , (Pang, Lee and Vaithyanathan, 2002), (Popescu and Etzioni, 2005), (Hu and Liu, 2004) and (Ding, Liu and Yu, 2008). However, present research has not included the discovery of implicit features and furthermore, it has left the problem of explicit features dependent on the mentioning of these features in the individual user reviews or not. The method we propose is language and customer-review independent. It extracts a set of general product features, finds product specific features and feature attributes and is thus applicable to all possible reviews in a product class. We describe the steps performed to obtain the features for each product class and the manner in which input text is processed to obtain the opinion expressed by customers.

4 System Architecture

Our method consists of two distinct steps: pre-processing and main processing, each containing a series of sub modules and using different language tools and resources.

4.1 Pre-processing

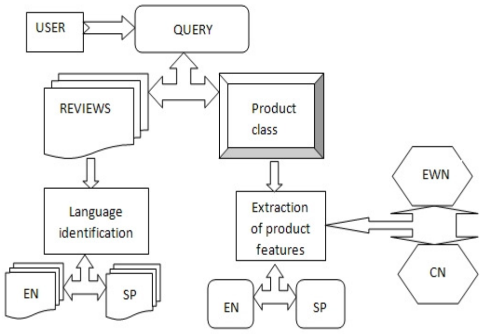

Figure 1: Pre-processing stage

As depicted in Figure 1, in our approach, we start from the following scenario: a user enters a query about a product that he/she is interested to buy. The search engine will retrieve a series of documents containing the product name, in different languages. Further on, two parallel operations are performed: the first one uses language identifier software to filter and obtain two categories - one containing the reviews in English and the other the reviews in Spanish. The second operation implies a modified version of the system described in (Kozareva and Montoyo, 2007) for the classification of person names. We use this system in order to determine the category the product queried belongs to. Once the product category is determined, we proceed to extracting the product specific features and feature attributes. This is accomplished using WordNet and ConceptNet and the corresponding mapping to Spanish using EuroWordNet. Apart from the product specific class of features and feature attributes, we consider a core of features and feature attributes that are product-independent and whose importance determines their frequent occurrence in customer reviews.

1) *Product-independent features and feature attributes:*

There are a series of features that are product independent and that are important to any prospective buyer. We consider these as forming a core of product features. For each of these concepts, we retrieve from WordNet the synonyms which have the same Relevant Domain (Vázquez, Montoyo and Rigau, 2004), the hyponyms of the concepts and their synonyms and attributes, respectively.

2) *Using WordNet to extract product specific features and feature attributes:* Once the product category has been identified, we use WordNet to extract the product specific features and feature attributes. We accomplish this in the following steps:

- For the term defining the product category, we search its synonyms in WordNet (Fellbaum, 1999)
- We eliminate the synonyms that do not have the same top relevant domain as the term defining the product category
- For the term defining the product, as well as each
- for each of the remaining synonyms, we obtain their meronyms from in WordNet, which constitute the parts forming the product.
- Since WordNet does not contain much detail on the components of most of new technological products, we use ConceptNet (Liu and Singh, 2004) to complete the process of determining the specific product features. We explain the manner in which we use ConceptNet in the following section.

After performing the steps described above, we conclude the process of obtaining the possible terms that a customer buying a product will comment on. The final step consists in finding the attributes of the features discovered by applying the "has attributes" relation in WordNet to each of the nouns representing product features. In the case of nouns which have no term associated by the *"has attribute"* relation, we add as attribute features the concepts found in *ConceptNet* under the *OUT* relations *PropertyOf* and *CapableOf*. In case the concepts added are adjectives, we further add their synonyms and antonyms from WordNet.

3) *Using ConceptNet to extract product specific features and feature attributes:*

In order to obtain additional features for the product in question, we add the concepts that are related to the term representing the concept with terms related in ConceptNet by the *OUT* relations *UsedFor* and *CapableOf* and the *IN* relations *PartOf* and *UsedFor*.

4) *Mapping concepts using EuroWordNet:*

We employ EuroWordNet and map the features and feature attributes, both from the main core of words, as well as the product specific ones that were previously discovered for English, independent of the sense number, taking into account only the preservation of the relevant domain. Certainly, we are aware of the noise introduced by this mapping, however in the preliminary research we found that the concepts introduced that had no relation to the product queried did not appear in the user product reviews.

5) Discovering overlooked product features: The majority of product features we have identified so far are parts constituting products. However, there remains a class of undiscovered features that are indirectly related to the product. These are the features of the product constituting parts, such as battery life, picture resolution, and auto mode. Further, we propose to extract these overlooked product features by determining bigrams made up of target words constituting features and other words in a corpus of customer reviews. In the case of digital cameras, for example, we considered a corpus of 200 customer reviews on which we ran Pedersen's Ngram Statistics Package (Banerjee and Pedersen, 2003) to determine target co-occurrences of the features identified so far. As measure for term association, we use the Pointwise Mutual Information score. In this manner, we discover bigram features such as "battery life", "mode settings" and "screen resolution".

4.2 Main Processing

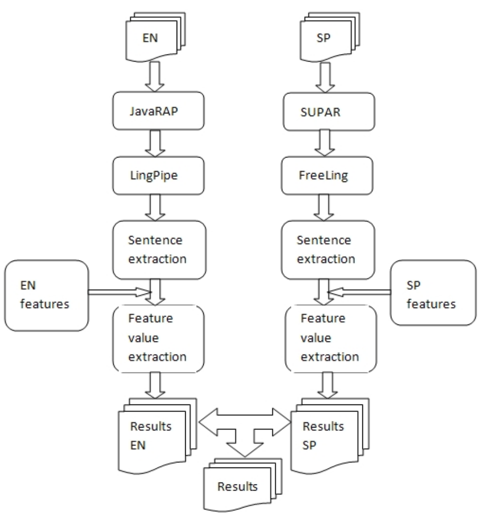

Figure 2: Main processing stage

The main processing in our system is done in parallel for English and Spanish. In the next section, we will briefly describe the steps followed in processing the initial input containing the customer reviews in the two considered language and offer as output the summarized opinions on the features considered. We part from the reviews filtered according to language. For each of the two language considered, we used a specialized tool for anaphora resolution - JavaRAP[1] for English and SUPAR (Ferrández, Palomar and Moreno, 1999) for Spanish. Further on, we separate the text into sentences and use a Named Entity Recognizer to spot names of products, brands or shops. Using the lists of general features and feature attributes, product-specific features and feature attributes, we extract from the set of sentences contained in the text only those containing at least one of the terms found in the lists.

1) Anaphora resolution: In order to solve the anaphoric references on the product features and feature attributes, we employ two anaphora resolution tools - JavaRAP for English and SUPAR for Spanish. Using these tools, we replace the anaphoric references with their corresponding referents and obtain a text in which the terms constituting product features could be found.

Using JavaRAP, we obtain a version of the text in which pronouns and lexical references are resolved. For example, the text: *''I bought this camera about a week ago,and so far have found it very very simple to use, takes good quality pics for what I use it for (outings with friends/family, special events). It is great that it already comes w/ a rechargeable battery that seems to last quite a while...''*, by resolving the anaphoric pronominal reference, becomes *''I bought this camera about a week ago, and so far have found <this camera > very very simple to use, takes good quality pics for what I use <this camera > for (outings with friends/family, special events). It is great that <this camera> already comes w/ a rechargeable battery that seems to last quite a while...''*.

SUPAR (Slot Unification Parser for Anaphora Resolution). We use SUPAR in the same manner as JavaRAP, to solve the anaphora for Spanish.

1 http://www.comp.nus.edu.sg/~qiul/NLPTools/JavaRAP.html

2) Sentence chunking and NER: Further on, we split the text of the customer review into sentences and identify the named entities in the text. Splitting the text into sentences prevents us from processing sentences that have no importance as far as product features that a possible customer could be interested in are concerned.

We use LingPipe to split the customer reviews in English into sentences and identify the named entities referring to products of the same category as the product queried. In this manner, we can be sure that we identify sentences referring to the product queried, even the reference is done by making use of the name of another product. For example, in the text *"For a little less, I could have bought the Nikon Coolpix, but it is worth the extra money."*, anaphora resolution replaces <*it*> with <*Nikon Coolpix*> and this step will replace it with <*camera*>. We employ FreeLing in order to split the customer reviews in Spanish into sentences and identify the named entities referring to products of the same category as the product queried.

3) Sentence extraction: Having completed the feature and feature attributes identification phase, we proceed to extracting for further processing only the sentences that contain the terms referring to the product, product features or feature attributes. In this manner, we avoid further processing of text that is of no importance to the task we wish to accomplish. For example, sentences of the type "I work in the home appliances sector." will not be taken into account in further processing. Certainly, at the overall level of review impact, such a sentence might be of great importance to a reader, since it proves the expertise of the opinion given in the review. However, for the problems we wish to solve by using this method, such a sentence is of no importance.

4) Sentence parsing: Each of the sentences that are filtered by the previous step are parsed in order to obtain the sentence structure and component dependencies. In order to accomplish this, we use Minipar (Lin, 1998) for English and FreeLing for Spanish. This step is necessary in order to be able to extract the values of the features mentioned based on the dependency between the attributes identified and the feature they determine.

5) Feature value extraction: Further on, we extract features and feature attributes from each of the identified sentences, using the following rules:

1) We introduce the following categories of context polarity shifters, in which we split the modifiers and modal operators in two categories - positive and negative:

- *negation:* no, not, never etc.
- *modifiers:* positive (extremely, very, totally etc.) and negative (hardly, less, possibly etc.) - modal operators: positive (must, has) and negative (if, would, could etc.)

2) For each identified feature that is found in a sentence, we search for a corresponding feature attribute that determines it. Further on, we search to see if the feature attribute is determined by any of the defined modifiers. We consider a variable we name ***valueOfModifier***, with a default value of -1, that will account for the existence of a positive or negative modifier of the feature attribute. In the affirmative case, we assign a value of 1 if the modifier is positive and a value of 0 if the modifier is negative. If no modifier exists, we consider the default value of the variable. We extract triplets as *(feature, attributeFeature, valueOf Modifier)*. In order to accomplish this, we use the syntactic dependency structure of the phrase, we determine all attribute features that determine the given feature (in the case of Minipar, they are the ones connected by the *"mod"* and *"pred"* relations).

3) If a feature attribute is found without determining a feature, we consider it to implicitly evoke the feature that it is associated with in the feature collection previously built for the product. *"The camera is small and sleek."* becomes *(camera, small, -1)* and *(camera, sleek, -1)*, which is then transformed by assigning the value *"small"* to the *"size"* feature and the value *"sleek"* to the *"design"* feature.

5 Assigning polarity to feature attributes

In order to assign polarity to each of the identified feature attributes of a product, we employ SMO SVM machine learning and the Normalized Google Distance (NGD). The main advantage in using this type of polarity assignment is that NGD is language independent and offers a measure of semantic similarity taking into account the meaning given to words in all texts indexed by Google from the World Wide Web.

The set of anchors contains the terms *{featureName, happy, unsatisfied, nice, small, buy}*, that have possible connection to all possible classes of products and whose polarity is known. Further on, we build the classes of positive and negative examples for each of the feature attributes considered. From the corpus of annotated customer reviews, we consider all positive and negative terms associated to the considered attribute features. We then complete the lists of positive and negative terms with their WordNet synonyms. Since the number of positive and negative examples must be equal, we will consider from each of the categories a number of elements equal to the size of the smallest set among the two, with a size of at least 10 and less or equal with 20. We give as example the classification of the feature attribute *"tiny"*, for the *"size"* feature. The set of positive feature attributes considered contains 15 terms such as (big, broad, bulky, massive, voluminous, large-scale etc.) and the set of negative feature attributes considered is composed as opposed examples, such as (small, petite, pocket-sized, little, etc.). We use the anchor words to convert each of the 30 training words to 6-dimensional training vectors defined as $v(j,i) = NGD(w_i,a_j)$, where a_j with j ranging from 1 to 6 are the anchors and w_i, with i from 1 to 30 are the words from the positive and negative categories. After obtaining the total 180 values for the vectors, we use SMO SVM to learn to distinguish the product specific nuances. For each of the new feature attributes we wish to classify, we calculate a new value of the vector $vNew(j,word)=NGD(word, a_j)$, with j ranging from 1 to 6 and classify it using the same anchors and trained SVM model. In the example considered, we had the following results (we specify between brackets the word to which the scores refer to):

(small)1.52, 1.87, 0.82, 1.75, 1.92,1.93, positive
(little)1.44, 1.84, 0.80, 1.64, 2.11,1.85, positive
(big)2.27, 1.19, 0.86, 1.55, 1.16, 1.77, negative
(bulky) 1.33, 1.17 ,0.92,1.13,1.12,1.16, negative

The vector corresponding to the "tiny" attribute feature is:

(tiny) 1.51, 1.41, 0.82, 1.32, 1.60, 1.36.

This vector was classified by SVM as positive, using the training set specified above. The precision value in the classifications we made was between 0.72 and 0.80, with a kappa value above 0.45.

6 Summarization of feature polarity

For each of the features identified, we compute its polarity depending on the polarity of the feature attribute that it is determined by and the polarity of the context modifier the feature attribute is determined by, in case such a modifier exists. Finally, we statistically summarize the polarity of the feature attributes, as shown in Formula (1) and Formula (2):

$$F_{pos}(\mathrm{i}) = \frac{\#\mathrm{pos\_feature\_attributes(i)}}{\#\mathrm{feature\_attributes(i)}} \quad (1)$$

$$F_{neg}(\mathrm{i}) = \frac{\#\mathrm{neg\_feature\_attributes(i)}}{\#\mathrm{feature\_attributes(i)}} \quad (2)$$

The results shown are triplets of the form (feature, % Positive Opinions, % Negative Opinions).

7 Evaluation and discussion

For the evaluation of the system, we annotated a corpus of 50 customer reviews for each language, collected from sites as amazon.com, newegg.com, dealsdirect.com, ciao.es, shopmania.es, testfreaks.es and quesabesde.com. The corpus was annotated at the level of feature attributes, by the following scheme: *<attribute> [name of attribute] <feature> [feature it determines] </feature> <value> [positive / negative] </value> </attribute>*.

It is difficult to evaluate the performance of such a system, since we must take into consideration both the accuracy in extracting the features that reviews comment on, as well as the correct assignation of identified feature attributes to the positive or negative category. Therefore, we measured the system performance in terms of precision, recall and accuracy. The results obtained are summarized in Table 1. We show the scores for each of the two languages considered separately and the combined score when using both systems for assigning polarity to feature attributes of a product. In the last column, we present a baseline, computed as average of using the same formulas, but taking into consideration, for each feature, only the feature attributes we considered as training examples for our method. We can notice how the use of NGD helped the system acquire significant new knowledge about the polarity of feature attributes.

	Eng	Sp	Combined	Baseline Eng	Baseline Sp
SA	0.82	0.80	0.81	0.21	0.19
FIP	0.80	0.78	0.79	0.20	0.20
FIR	0.79	0.79	0.79	0.40	0.40

Table 1: System results

The problems encountered were largely related to the use of informal language, disregard of spelling rules and punctuation marks.

8 Conclusions and future work

In this paper we presented a method to extract, for a given product, the features that could be commented upon in a customer review. Further, we have shown a method to acquire the feature attributes on which a customer can comment in a review. Moreover, we presented a method to extract and assign polarity to these product features and statistically summarize the polarity they are given in the review texts in English and Spanish. The method for polarity assignment is largely language independent (it only requires the use of a small number of training examples) and the entire system can be implemented in any language for which similar resources and tools as the ones used for the presented system exist. The main advantage obtained by using this method is that one is able to extract and correctly classify the polarity of feature attributes, in a product dependent manner. Furthermore, the features in texts are that are identified are correct and the percentage of identification is high. Also, the polarity given in the training set determines the polarity given to new terms, such that "large" in the context of "display" will be trained as positive and in the case of "size" as negative. The main disadvantage consists in the fact that SVM learning and classification is dependent on the NGD scores obtained with a set of anchors that must previously be established. This remains a rather subjective matter. The most important problem we encountered is that concerning the informal language style, which makes the identification of words and dependencies in phrases sometimes impossible.

Future work includes the development of a method to extend the list of product-dependent features and feature attributes, alternate methodologies for polarity assignation to product dependent feature attributes and finally, the application of a textual entailment system to verify the quality of the feature extracted and the assigned polarity.

References

Banerjee, S., Pedersen,T.: The Design, Implementation and Use of the Ngram Statistics Package. In Proceedings of the Fourth International Conference on Intelligent Text Processing and Computational Linguistics, February 17-21, 2003, Mexico City. (2003)

Chaovalit, P., Zhou, L.: Movie Review Mining: a Comparison between Supervised and Unsupervised Classification Approaches. In Proceedings of HICSS-05, the 38th Hawaii International Conference on System Sciences. (2005)

Cilibrasi, D., Vitanyi, P.: Automatic Meaning Discovery Using Google. IEEE Journal of Transactions on Knowledge and Data Engineering. (2006)

Cui, H., Mittal, V., Datar, M.: Comparative Experiments on Sentiment Classification for Online Product Reviews. In Proceedings of the 21st National Conference on Artificial Intelligence AAAI 2006. (2006)

Dave, K., Lawrence, S., Pennock, D.: Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews. In Proceedings of WWW-03. (2003)

Ding, X., Liu, B., Yu, P.,: A Holistic Lexicon - Based Approach to Opinion Mining. In Proceedings of WSDM, 2008. (2008)

Fellbaum(ed.), C.: WordNet: An Electronic Lexical Database. First edn. MIT Press. (1999)

Ferrández, A., Palomar, M., Moreno, L.: An Empirical Approach to Spanish Anaphora Resolution. Machine Translation. Special Issue on Anaphora Resolution In Machine Translation. Special Issue on Anaphora Resolution In Machine Translation. (1999)

Gamon, M., Aue, S., Corston-Oliver, S., Ringger, E.: Mining Customer Opinions from Free Text. Lecture Notes in Computer Science. (2005)

Goldberg, A.B., Zhu, J.: Seeing stars when there aren't many stars: Graph-based semi-supervised learning for sentiment categorization. In HLT-NAACL 2006 Workshop on Textgraphs: Graph-based Algorithms for Natural Language Processing. (2006)

Hatzivassiloglou, V., Wiebe, J.: Effects of adjective orientation and gradability on sentence subjectivity. In Proceedings of COLING 2000. (2000)

Hu, M., Liu, B.: Mining Opinion Features in Customer Reviews. In Proceedings of Nineteenth National Conference on Artificial Intelligence AAAI-2004. (2004)

Kim, S.M., Hovy, E.: Determining the Sentiment of Opinions. In Proceedings of COLING 2004. (2004)

Kozareva, Z., Montoyo, A.: Discovering the Underlying Meanings and Categories of a Name through Domain and Semantic Information. In: Proceedings of the Conference on Recent Advances in Natural Language Processing RANLP 2007. (2007)

Lin, D.: Dependency-based Evaluation of MINIPAR. In Workshop on the Evaluation of Parsing Systems. (1998)

Lin, W.H., Wilson, T., Wiebe, J., Hauptman, A.: Which Side are You On? Identifying Perspectives at the Document and Sentence Levels. In Proceedings of the Tenth Conference on Natural Language Learning CoNLL'06. (2006)

Liu, B.: Web Data Mining. Exploring Hyperlinks, Contents and Usage Data. First edn. Springer (2007)

Liu, H., Singh, P.: ConceptNet: A Practical Commonsense Reasoning Toolkit. BT Technology Journal 22. (2004)

Mullen, T., Collier, N.: Sentiment Analysis Using Support Vector Machines with Diverse Information Sources. In Proceedings of EMNLP 2004. (2004)

Ng, V., Dasgupta, S., Arifin, S.M.N.: Examining the Role of Linguistic Knowledge Sources in the Automatic Identification and Classification of Reviews. In Proceedings 40th Annual Meeting of the Association for Computational Linguistics. (2006)

Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? Sentiment classification using machine learning techniques. In Proceedings of EMNLP-02, the Conference on Empirical Methods in Natural Language Processing. (2002)

Platt, J.: Sequential minimal optimization: A fast algorithm for training support vector machines. Microsoft Research Technical Report MSRTR- 98-14. (1998)

Popescu, A.M., Etzioni, O.: Extracting Product Features and Opinions from Reviews. In Proceedings of EMNLP 2005. (2005)

Riloff, E., Wiebe, J.: Learning Extraction Patterns for Subjective Expressions. In Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing. (2003)

Stoyanov, V., Cardie, C.: Toward Opinion Summarization: Linking the Sources. In: COLING-ACL 2006 Workshop on Sentiment and Subjectivity in Text. (2006)

Turney, P., Littman, M.: Measuring praise and criticism: Inference of semantic orientation from association. ACM Transactions on Information Systems 21. (2003)

Turney, P.: Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics. (2002)

Vázquez, S., Montoyo, A., Rigau, G.: Using relevant domains resource for word sense disambiguation. In Proceedings of the ICAI 2004. (2004)

Wiebe, J., Riloff, E.: Creating Subjective and Objective Sentence Classifiers from Unannotated Texts. In Proceedings of the 6th International Conference on Computational Linguistics and Intelligent Text Processing (CICLing-05). (2005)

Wilson, T., Wiebe, J., Hwa, R.: Just how mad are you? Finding strong and weak opinion clauses. In: Proceedings of AAAI 2004. (2004)