# The Benefit of Concept-Based Features for Sentiment Analysis

Kim Schouten and Flavius Frasincar

Erasmus University Rotterdam
PO Box 1738, NL-3000 DR
Rotterdam, the Netherlands
{schouten, frasincar}@ese.eur.nl

**Abstract.** Sentiment analysis is an active field of research, moving from the traditional algorithms that operated on complete documents to fine-grained variants where aspects of the topic being discussed are extracted, as well as their associated sentiment. Recently, a move from traditional word-based approaches to concept-based approaches has started. In this work, it is shown by using a simple machine learning baseline, that concepts are useful as features within a machine learning framework. In all our experiments, the performance increases when including the concept-based features.

## 1 Introduction

Sentiment analysis is an active field of research, and much progress has been made since the early algorithms that could only predict polarity for complete documents. Nowadays, advanced methods are available that can detect the various aspects of the topic being discussed and their associated polarity. However, methods for sentiment analysis tend to lean heavily on machine learning, leaving only a small role for natural language processing. Traditionally, a bag-of-words approach is used where the features for a machine learning algorithm are simple binary features denoting the presence or absence of a word. While these methods perform well, classifying the majority of the cases correctly, their performance has reached a plateau since word-based approaches cannot correctly classify all cases (e.g., they fail to account for the grammatical structure in the text and its associated semantics). For the remaining, harder cases, more advanced methods are required. In [4], a move from traditional word-based approaches to concept-based methods is advocated, and in this paper we would like to demonstrate the usefulness of concepts for the task of sentiment analysis.

To support the previous claim, we have set up a basic linear support vector machine (SVM) for the task of sentence polarity classification, and aspect (category) detection. Since the number of features is very large and the number of data points is relatively small, a linear SVM is best suited here. By having both word-based features, grammar-based features, and concept-based features, we show that concepts are always beneficial to add to the set of features, as in our experiments the results always improve.

The remainder of this paper is structured as follows: first, we describe some of the existing work in this field. Then, we discuss our baseline methods for sentence polarity classification, followed by our baseline method for aspect (category) detection. Then, to complete the package, we describe a method for aspect polarity classification. Each method is evaluated in their respective section. Last, conclusions are drawn and some pointers for future work are given.

## 2 Related Work

For a field as new as concept-centric sentiment analysis, there are already a number of approaches proposed. First, there is the set of works presented at last year's Semantic Web Evaluation Challenge. Furthermore, there are other pioneering works that present semantic approaches towards sentiment analysis.

In [17], a concept parser is used to first extract all the concepts in each sentence. The concept parser is a set of handcrafted rules executed on the dependency parse tree output. Then the aspects are extracted in a similar fashion, using an elaborate system of handcrafted rules. In addition to the dependency parse tree and the already found concepts, these rules utilize a manually created lexicon to detect implicit aspects and an opinion lexicon (i.e., SenticNet [3]). The sentiment analysis was also performed with a rule-based method, but when no concept was found that was in SenticNet, a basic machine learning method was employed as a fall-back mechanism. A similar method is proposed in [24], where a semantic role labeling component is used after the syntactic parser. On top of that, a set of handcrafted rules is executed that describe patterns, using semantic role information and syntactic information, that denote aspects.

The work presented in [6] presents a machine learning method for polarity detection, where the traditional bag-of-words approach is complemented with semantic features. A graph-based approach [19] is used to extract the concepts from the reviews, and then SentiConceptNet [23] is used, together with a term weighting scheme, to construct the concept features, that thus consist of a weighted concept term times the concept's sentiment score.

A lexicon-based method is given in [14], where given a seed set of adjectives where the polarity is known, new adjectives are found using the conjunction rule [10] (i.e., if an unknown adjective is conjoined to a known positive adjective with 'and', then the unknown adjective is also positive). In addition, its synonyms are also added to the known list of adjectives together with its antonyms, which will get the opposite polarity score.

An ontology forms the core element of [7], modeling the space of online reviews. It is populated with instances from DBPedia [11], using lexicalizations from the DBPedia Lexicalization Dataset [13]. These lexicalizations are expanded by analyzing words appearing in a similar context (i.e., the set of words around a term). This allows new concepts, that are not already described in the ontology to be found as well. In addition, it includes prior information, like word lists of generally positive words and generally negative words. Furthermore, it employs a list of association concepts, where prior information is encoded as a <concept, opinion, sentiment> triple (e.g., <beer, cold, positive>).

An ontology-based approach is advocated in [25], as well. Here, term frequency is used to find the most descriptive words for a given product concept in product descriptions on the Web. Then, all synonyms and hyponyms are added as lexicalizations of that concept to the ontology. In this way, a concept is described by a set of weighted terms, with weights denoting the association degree between the word and the underlying concept. This association degree is based, both on presence and absence of terms in its context. Then, all adjectives, nouns, and verbs that are not identified as aspects are considered as possible sentiment words. When these sentiment candidates can be paired to an already known aspect, the aspect-opinion pair is complete. All sentiment candidates that are not paired with a known aspect are considered as yet unknown aspects.

In [15], the labels of the classes and instances in the employed ontology are used to find the aspects described in the ontology in the text, without making use of specific lexicalizations. However, it features different weights for the different aspects, according to, for example, how often this aspect is mentioned by users in their reviews. To compute the sentiment score, SentiWordNet [1] is used.

A different approach is taken in [8], where fuzzy logic is used to model the relationships between the polarity of a concepts and the domain, as well as the aboutness. To that end, a two-level graph is used, where the first level models the relations between concepts, whereas the second level models the relations between concepts and sentiment given various domains (i.e., the same word can be positive in one domain, but negative in another). A preliminary membership function is defined using the training data, having a triangle shape. These membership functions are refined in a later step, to arrive at trapezoid functions, by propagating learned information through the two-level graph. Using this method, the various membership functions will influence each other (e.g., if a semantically related concept has a strong positive polarity in a given domain, than the current concept most likely is positive in that domain as well).

Also using fuzzy logic is [21], where a fuzzy sentiment ontology is built. It uses eight different sentiment classes (i.e., expect, joy, love, surprise, anxiety, sorrow, angry, and hate [18]). Every word has different membership values for the different sentiment classes, corresponding to the semantic similarity between that word and the word denoting the sentiment class. These values differ for different meanings of a word.

Our machine learning baseline is most similar to [6], since it also uses a machine learning algorithm with concept features. However, we use a word sense dismabiguation step to link words to concepts. The rule-based approach is not concept-centric and is presented as an additional baseline.

## 3 Sentence Polarity Classification

The sentence polarity classification task, is an elementary task that is concerned with finding the overall polarity of a sentence. We have two methods, a rule-based method based and a machine learning method.

The rule-based method is based on OASYS [5], but with an updated formula to compute the sentiment score for words. The sentiment of each word is computed by adding $\frac{1}{sentenceLength}$ for each positive sentence this word appears in and subtracting the same value for each negative sentence this word appears in. Furthermore, negation and amplification are taken into account as well. The computation of the sentiment score of a word then becomes

$$sentiment(w) = \frac{1}{freq(w)} \sum_{s \text{ having } w} \frac{polarity(s)}{mod_s(w) \times length(s)} \tag{1}$$

where $sentiment(w)$ is the sentiment score computed for word $w$, $freq(w)$ is the number of times this word appears in the training data, $s$ having $w$ is a sentence that contains word $w$, $polarity(s)$ is either +1 for positive sentences, and -1 for negative sentences (as taken from the annotated training data), and $length(s)$ is the number of words in sentence $s$. When the current word $w$ has a 'neg' dependency, the modifier $mod_s(w)$ is -0.9 to denote negation. When $w$ has a 'advmod' relation with a word that is in the General Inquirer [22] 'Overstatement' list, the $mod_s(w)$ is 1.4 to denote amplification, and conversely, its value is 0.6 when the word in the 'advmod' relation is in the General Inquirer 'Understatement' list.

Furthermore, an offset value is computed as the average of: the average sentiment score of positive sentence and the average sentiment score of negative sentences. This is to offset any inbalance between positive and negative sentences in the dataset.

When processing unseen sentences, the sentiment of the sentence is the sum of the word sentiment scores, computed as

$$sentiment(s) = \frac{1}{length(s)} \sum_{w \in [s]} mod_s(w) \times sentiment(w), \tag{2}$$

where $sentiment(w)$ is the sentiment score for that word, as defined above, $lengths(s)$ is the number of words in sentence $s$, $[s]$ denotes the bag of words representation of sentence $s$, and $mod_s(w)$ represents a modifier for negation and amplification as in Eq. 1.

The machine learning method is based on a linear Support Vector Machine, using a variety of features. The first set of features is constructed by encoding the presence or absence of the lemmas of corpus words in a sentence ($\mathcal{L}$). In a similar fashion, the next set of features consists of the concepts that these words represent ($\mathcal{C}$). We use the Lesk [12] algorithm for word sense disambiguation, linking the lemmas to concepts in WordNet [9]. Then, we encode the presence or absence of each concept in a sentence. A third set of features is made by encoding grammatical lemma bigrams ($\mathcal{LG}$) of the form "`lemma - grammatical relation type - lemma`" (e.g., "house-amod-big", where 'amod' stands for adjectival modifier). A fourth set of feature is created by encoding grammatical polarity bigrams ($\mathcal{PG}$) of the form "`word polarity - grammatical relation type - word polarity`" (e.g., "neutral-amod-positive'). Last, we encode some general polarity characteristics of the sentence ($\mathcal{PC}$): whether there are more positive than negative words, whether there are positive words in the sentence, and whether there are negative words in the sentence. To get the word polarities, we use the General Inquirer lexicon [22], using the 'Positiv' and 'Negativ' word list. In the future, we would like to also incorporate polarity information from SentiWordNet, as this is concept-based instead of word-based like the General Inquirer.

### 3.1 Data

For the sentence polarity classification task, the data set used is the Multi-Domain Sentiment Dataset from Blitzer et al. [2]. It contains 2429 sentences, taken from Amazon reviews, from various product domains (e.g., books, movies, games, etc.). We use the binary version of the polarity annotations, where polarity is simply positive or negative. Some sentences are very short and contain only a few words, while others are extremely long, with more than 150 words. About 58% of the sentences is labeled positive, with the remaining 42% being labeled as negative.

### 3.2 Evaluation

To evaluate the proposed method, we use ten-fold cross-validation. The training data is split into ten equal parts, and the algorithm is tested on each of the ten parts, having been trained on the nine other parts. Since sentences are assigned randomly to one of the ten folds, the results can vary a little bit with each run.

From the results in Table 1 one can clearly see that the traditional bag-of-words approach is well-performing. However, a small but noticeable improvement can be seen when adding concepts from WordNet to the feature set. Whatever combination of features is used, it is always better to also include the WordNet concepts, showing the added value of these kinds of features.

In Table 2 and 3, the results of the rule-based method and the machine learning method using all feature sets are shown for the individual positive and negative labels. Interestingly, where precision for positive and negative are similar, recall is much lower for negative labels than for positive labels. A possible reason for this is that people generally use the same kind of language to denote a positive opinion, whereas there are many more ways of saying something negative about a product or service (e.g., people try to write a critical review in a polite manner, but also the wide variety of negative words). This bigger variety poses problems for recall, since the algorithm will encounter new forms of negative opinions which it has not seen before in the training data.

**Table 1:** Results for task 1: sentence polarity classification.

| Used feature sets | | | | | Precision |
|---|---|---|---|---|---|
| Majority Baseline | | | | | 58.46 |
| Rule-based | | | | | 74.90 |
| $\mathcal{L}$ | | | | | 73.69 |
| | $\mathcal{C}$ | | | | 71.76 |
| | | $\mathcal{LG}$ | | | 65.34 |
| | | | $\mathcal{PG}$ | | 62.49 |
| | | | | $\mathcal{PC}$ | 58.46 |
| $\mathcal{L}$ | $\mathcal{C}$ | | | | 75.71 |
| $\mathcal{L}$ | $\mathcal{C}$ | $\mathcal{LG}$ | | | 76.33 |
| $\mathcal{L}$ | $\mathcal{C}$ | $\mathcal{LG}$ | $\mathcal{PG}$ | | 76.16 |
| $\mathcal{L}$ | $\mathcal{C}$ | $\mathcal{LG}$ | $\mathcal{PG}$ | $\mathcal{PC}$ | 76.12 |
| $\mathcal{L}$ | | $\mathcal{LG}$ | $\mathcal{PG}$ | $\mathcal{PC}$ | 75.05 |
| $\mathcal{L}$ | | $\mathcal{LG}$ | $\mathcal{PG}$ | | 74.80 |
| $\mathcal{L}$ | | $\mathcal{LG}$ | | | 75.21 |

**Table 2:** Rule-based learning results for positive and negative labels when using all feature sets.

| | Precision | Recall | $F_1$-score |
|---|---|---|---|
| Overall | 74.90 | 74.90 | 74.90 |
| Negative | 71.24 | 66.37 | 68.72 |
| Positive | 77.21 | 80.96 | 79.04 |

**Table 3:** Machine learning results for positive and negative labels when using all feature sets.

| | Precision | Recall | $F_1$-score |
|---|---|---|---|
| Overall | 76.12 | 76.12 | 76.12 |
| Negative | 74.07 | 65.41 | 69.47 |
| Positive | 77.31 | 83.73 | 80.39 |

# 4 Aspect Detection

For aspect detection we use a limited set of aspects, which is known beforehand, so we can train a binary classifier for each aspect. We use a linear Support Vector Machine, with similar setup to the sentence polarity task. The feature sets used are the lemmas of the words in the sentence ($\mathcal{L}$), the concepts to which the words in a sentence refer to ($\mathcal{C}$), and the grammatical lemma bigrams ($\mathcal{LG}$). The polarity oriented feature sets used in the sentence polarity task are not used here, since classifying sentiment is not performed in this task.

## 4.1 Data

The dataset used for this task the is the official SemEval-2015 training data on restaurants [16]. It contains 277 reviews on restaurants, each containing one or more sentences. Each sentence is annotated with zero or more opinions, with each opinion having multiple information slots. The first slot contains the actual words in the sentence on which this opinion was voiced. For implicit opinions, that are not literally mentioned in the sentence, this slot is empty. The second slot is the category of the aspect, denoted as the combination of entity and attribute. The entity part represents high-level concepts, like 'Food', 'Location', 'Service', etc. The attribute part represents a subclass or attribute of that high-level concept. Example of attributes used in the restaurant data are 'Quality', 'Prices', 'General', etc. Combining the two yields both specific categories like 'Food#Quality', but also very general categories like 'Restaurant#General'. Note that the SVM described above learns these categories without taking the fact into account that they consist of two semantically related parts. A list of all category labels can be found in Table 4 below.

**Table 4:** The set of category labels for the SemEval restaurant data.

| Category | Attribute | Frequency |
|---|---|---|
| Ambience | General | 183 |
| Drinks | Prices | 15 |
| Drinks | Quality | 34 |
| Food | General | 1 |
| Food | Prices | 54 |
| Food | Quality | 581 |
| Food | Style Options | 93 |
| Location | General | 20 |
| Restaurant | General | 269 |
| Restaurant | Miscellaneous | 62 |
| Restaurant | Prices | 48 |
| Service | General | 268 |

### 4.2 Evaluation

The various combinations of the three feature sets are evaluated using ten-fold cross-validation. Since reviews, and the sentences they contain, are randomly assigned to one of the folds, the results may differ slightly with each run. The results can be seen in Table 5 below.

**Table 5:** Results for task 2: aspect category classification.

| Features | | | Precision | Recall | $F_1$-measure |
|---|---|---|---|---|---|
| $\mathcal{L}$ | | | 86.91 | 44.56 | 58.91 |
| | $\mathcal{C}$ | | 88.97 | 37.06 | 52.33 |
| | | $\mathcal{LG}$ | 91.36 | 24.30 | 38.40 |
| $\mathcal{L}$ | $\mathcal{C}$ | | 85.04 | 51.57 | 64.21 |
| $\mathcal{L}$ | | $\mathcal{LG}$ | 86.06 | 48.91 | 62.37 |
| | $\mathcal{C}$ | $\mathcal{LG}$ | 89.62 | 45.41 | 60.27 |
| $\mathcal{L}$ | $\mathcal{C}$ | $\mathcal{LG}$ | 84.61 | 53.51 | 65.56 |

Similar to the sentence polarity classification task, we can see the contribution of the concept-based features. Note that concepts on their own do not work as well as lemmas, since not all words are related to a concept, and hence, information is lost by only having concepts as features. This explains why especially recall is much lower for concepts than for lemmas. Nevertheless, precision is better for concepts then for lemmas, showing the adequacy of accounting for word semantics.

The current set of features is highly accurate, but does not have enough coverage as shown by the recall score. For that, more robust features, that generalize well to unseen data, are needed. In future work, we would like to exploit the relational structure of domain ontologies to increase the coverage of concept-based approaches.

## 5 Aspect Polarity Classification

For the polarity classification of the opinions on aspects, we use the same method as reported in [20]. We start by creating a sentiment lexicon from the annotated opinions. This domain-specific lexicon is then used to determine the sentiment of the opinions that have no sentiment annotation. The intuition behind this method is that the sentiment of words depends on the domain, and hence, it is convenient to automatically extract the word sentiment from the annotated corpus. Words that often appear close to positive aspects are likely to be positive, whereas words that often appear close to negative aspects are likely to be negative. Since sentiment is carried by expressions and not by single words alone, we also create lexicon entries for bigrams and trigrams. In each sentence, the distance between each n-gram and each aspect is computed and the sentiment of the aspect, discounted by the distance, is added to the overall sentiment value for that n-gram. This is shown in Eq. 3.

$$sentiment_g = \frac{1}{freq_g} \cdot \sum_{s \in S_g} p \cdot t_{order(g)} \cdot \sum_{a \in A_s} \frac{polarity_a}{(distance_{a,g})^m}, \tag{3}$$

where $g$ is the n-gram (i.e., word unigram, bigram, or trigram), $freq_g$ is the frequency of n-gram $g$ in the data set, $s$ is a sentence in $S_g$, which is the set of sentences that contain n-gram $g$, $p$ is a parameter to correct for the overall positivity of the data set, $t$ is a parameter that corrects for the relative influence of the type of n-gram (i.e., different values are used for unigrams, bigrams, and trigrams), $a$ is an aspect in $A_s$, which is the set of aspects in sentence $s$, $polarity_a$ is 1 when aspect $a$ is positive and $-1$ when $a$ is negative, and $m$ is a parameter that determines how strong the discounting by the distance should be. The distance $distance_{a,g}$ is computed as the minimum amount of words between the aspect $a$ and the n-gram $g$ (i.e., both an n-gram and an aspect can consist of multiple words, in which case the closest two are used to compute this distance). We set $t_{order(g)}$ to 1, 5, and 4 for unigrams, bigrams, and trigrams, respectively. Furthermore we set $p = 2$ and $m = 1$. These values were determined by manual experimentation.

The sentiment of an aspect is computed by taking the sentiment value of each n-gram from the lexicon, dividing it by the distance between that n-gram and the aspect, and summing it up, as shown in Eq. 4. For this, it is assumed that each aspect only appears once in a sentence.

$$sentiment_{a,s_a} = \sum_{g \in s_a} \frac{sentiment_g}{(\min distance_{g,a})^m}, \tag{4}$$

where, in addition to the definitions in the previous equation, $g$ is an n-gram in $s_a$, which is the sentence in which aspect $a$ occurs. For each occurrence of an n-gram, its sentiment value is added to a total score for that aspect. When this score exceeds zero, it will be annotated as ‘positive’, and with a score below zero, it will be annotated as ‘negative’. Since there are only a few neutral cases in our data set, and many more positive than negative aspects, we default to ‘positive’, when the total score is zero (e.g., this can also happen when no sentiment-bearing words are in this sentence). Neutral sentiment, although present in the training data, is ignored in this method.

For implicit opinions, where the target slot is ‘null’, the distance in the above formulas cannot be computed, and hence a distance of 1 is used instead.

### 5.1 Evaluation

This method is also evaluated on the official SemEval-2015 restaurant training data (cf. Sect. 4.1). The data set is heavily biased towards positive opinions: 1198 opinions are positive, 403 are negative, and 53 are neutral. As with the other methods, the aspect polarity classification method is evaluated using ten-fold cross-validation. Results are shown in Table 6 below.

**Table 6:** Results for the aspect polarity classification algorithm.

| | Precision | Recall | $F_1$-score |
|---|---|---|---|
| Overall | 76.30 | 76.30 | 76.30 |
| Negative | 54.55 | 52.11 | 53.30 |
| Positive | 83.03 | 87.81 | 85.35 |
| Neutral | 0 | 0 | 0 |

## 6 Conclusion

In this work, we have shown that including concept-based features always leads to improved performance. Since this is already the case, even in a relatively straightforward setup like this, it is clear that more advanced ways of handling semantic information will increase performance even more.

In terms of future work, we would like to incorporate sentiment lexicons, like sentic.net and SentiWordNet, as well as general knowledge bases like DBPedia, domain ontologies, and semantic lexicons like WordNet. This enables us to include more information about relations between concepts. Now, only grammatical relations were included, but conceptual relations are all the more interesting.

## Acknowledgment

The authors are partially supported by the Dutch national program COMMIT.

## 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 7th Language Resources and Evaluation Conference (LREC 2010). pp. 2200–2204 (2010)
2. Blitzer, J., Dredze, M., Pereira, F.: Biographies, Bollywood, Boom-boxes and Blenders: Domain Adaptation for Sentiment Classification. In: Proceedings of the Association for Computational Linguistics. pp. 187–205. ACL (2007)
3. Cambria, E., Olsher, D., Rajagopal, D.: SenticNet 3: A common and common-sense knowledge base for cognition-driven sentiment analysis. In: Proceedings of AAAI2014. pp. 1515–1521 (2014)
4. Cambria, E., Schuller, B., Xia, Y., Havasi, C.: New Avenues in Opinion Mining and Sentiment Analysis. IEEE Intelligent Systems 28(2), 15–21 (2013)
5. Cesarano, C., Dorr, B., Picariello, A., Reforgiato, D., Sagoff, A., Subrahmanian, V.: Oasys: An opinion analysis system. AAAI Spring Symposium on Computational Approaches to Analyzing Weblogs (2004)
6. Chung, J.K.C., Wu, C.E., Tsai, R.T.H.: Polarity Detection of Online Reviews Using Sentiment Concepts: NCU IISR Team at ESWC-14 Challenge on Concept-Level Sentiment Analysis. In: Proceedings of the 1st Semantic Web Evaluation Challenge, Communications in Computer and Information Science, vol. 475, pp. 53–58. Springer International Publishing (2014)

7. Coden, A., Gruhl, D., Lewis, N., Mendes, P., Nagarajan, M., Ramakrishnan, C., Welch, S.: Semantic Lexicon Expansion for Concept-Based Aspect-Aware Sentiment Analysis. In: Proceedings of the 1st Semantic Web Evaluation Challenge, Communications in Computer and Information Science, vol. 475, pp. 34–40. Springer International Publishing (2014)
8. Dragoni, M., Tettamanzi, A., da Costa Pereira, C.: A Fuzzy System for Concept-Level Sentiment Analysis. In: Proceedings of the 1st Semantic Web Evaluation Challenge, Communications in Computer and Information Science, vol. 475, pp. 21–27. Springer International Publishing (2014)
9. Fellbaum, C.: WordNet: An Electronic Lexical Database. MIT Press (1998)
10. Hatzivassiloglou, V., McKeown, K.R.: Predicting the Semantic Orientation of Adjectives. In: Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics (ACL1998). pp. 174–181. Association for Computational Linguistics (1997), http://dx.doi.org/10.3115/976909.979640
11. Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P.N., Hellmann, S., Morsey, M., van Kleef, P., Auer, S., Bizer, C.: DBpedia - A large-scale, multilingual knowledge base extracted from Wikipedia. Semantic Web 6(2), 167–195 (2015)
12. Lesk, M.: Automatic Sense Disambiguation Using Machine Readable Dictionaries: How to Tell a Pine Cone from an Ice Cream Cone. In: 5th Annual International Conference on Systems Documentation (SIGDOC 1986). pp. 24–26. ACM (1986)
13. Mendes, P.N., Jakob, M., Bizer, C.: DBpedia: A Multilingual Cross-domain Knowledge Base. In: Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC2012). pp. 1813–1817 (2012)
14. Ofek, N., Rokach, L.: Unsupervised Fine-Grained Sentiment Analysis System Using Lexicons and Concepts. In: Proceedings of the 1st Semantic Web Evaluation Challenge, Communications in Computer and Information Science, vol. 475, pp. 28–33. Springer International Publishing (2014)
15. Peñalver-Martinez, I., Garcia-Sanchez, F., Valencia-Garcia, R., Ángel Rodríguez-García, M., Moreno, V., Fraga, A., Sánchez-Cervantes, J.L.: Feature-Based Opinion Mining through Ontologies. Expert Systems with Applications 41(13), 5995 – 6008 (2014)
16. Pontiki, M., Galanis, D., Papageogiou, H., Manandhar, S., Androutsopoulos, I.: SemEval-2015 Task 12: Aspect Based Sentiment Analysis. In: Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015) (2015)
17. Poria, S., Ofek, N., Gelbukh, A., Hussain, A., Rokach, L.: Dependency Tree-Based Rules for Concept-Level Aspect-Based Sentiment Analysis. In: Proceedings of the 1st Semantic Web Evaluation Challenge, Communications in Computer and Information Science, vol. 475, pp. 41–47. Springer International Publishing (2014)
18. Quan, C., Ren, F.: A Blog Emotion Corpus for Emotional Expression Analysis in Chinese. Computer Speech and Language 24(4), 726 – 749 (2010)
19. Rajagopal, D., Cambria, E., Olsher, D., Kwok, K.: A Graph-based Approach to Commonsense Concept Extraction and Semantic Similarity Detection. In: Proceedings of the 22nd International Conference on World Wide Web Companion (WWW2013Companion. pp. 565–570. International World Wide Web Conferences Steering Committee (2013)
20. Schouten, K., Frasincar, F., de Jong, F.: Commit-p1wp3: A co-occurrence based approach to aspect-level sentiment analysis. In: Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014). pp. 203–207. Association for Computational Linguistics and Dublin City University (2014)

21. Shi, W., Wang, H., He, S.: Sentiment Analysis of Chinese Microblogging Based on Sentiment Ontology: A Case Study of '7.23 Wenzhou Train Collision'. Connection Science 25(4), 161–178 (2013)
22. Stone, P.J., Dunphy, D.C., Smith, M.S., Ogilvie, D.M.: The General Inquirer: A Computer Approach to Content Analysis. MIT Press (1966)
23. Tsai, A.R., Wu, C.E., Tsai, R.H., Hsu, J.: Building a Concept-Level Sentiment Dictionary Based on Commonsense Knowledge. IEEE Intelligent Systems 28(2), 22–30 (2013)
24. Virk, S., Lee, Y.H., Ku, L.W.: Sinica Semantic Parser for ESWC'14 Concept-Level Semantic Analysis Challenge. In: Proceedings of the 1st Semantic Web Evaluation Challenge, Communications in Computer and Information Science, vol. 475, pp. 48–52. Springer International Publishing (2014)
25. Yin, P., Wang, H., Guo, K.: Feature-Opinion Pair Identification of Product Reviews in Chinese: A Domain Ontology Modeling Method. New Review of Hypermedia and Multimedia 19(1), 3–24 (2013)