Web opinion mining: How to extract opinions from blogs?

Ali Harb [*]
EMA-LGI2P
Parc Scientifique G. Besse
30035 Nîmes Cedex, France
ali.harb@ema.fr

Gerard Dray
EMA-LGI2P
Parc Scientifique G. Besse
30035 Nîmes Cedex, France
gerard.dray@ema.fr

Michel Plantié
EMA-LGI2P
Parc Scientifique G. Besse
30035 Nîmes Cedex, France
michel.plantie@ema.fr

Mathieu Roche [†]
LIRMM CNRS 5506
UM II, 161 Rue Ada
F-34392 Montpellier, France
mathieu.roche@lirmm.fr

François Trousset
EMA-LGI2P
Parc Scientifique G. Besse
30035 Nîmes Cedex, France
francois.trousset@ema.fr

Pascal.Poncelet
EMA-LGI2P
Parc Scientifique G. Besse
30035 Nîmes Cedex, France
pascal.poncelet@ema.fr

ABSTRACT

With the growing popularity of the Web 2.0, we are more and more provided with documents expressing opinions on different topics. Recently, new research approaches were defined in order to automatically extract such opinions on the Internet. Usually they consider that opinions are expressed through adjectives and they extensively use either general dictionaries or experts in order to provide the relevant adjectives. Unfortunately these approach suffer the following drawback: for a specific domain either the adjective does not exist or its meaning could be different from another domain. In this paper, we propose a new approach focusing on two steps. First we automatically extract from the Internet a learning dataset for a specific domain. Second we extract from this learning set, the set of positive and negative adjectives relevant for the domain. Conducted experiments performed on real data show the usefulness of our approach.

Keywords

Text Mining, Opinion Mining, Association Rules, Semantic Orientation.

1. INTRODUCTION

With the fast growing development of the Web, and especially of the Web 2.0, the number of documents expressing opinions becomes more and more important. As illustration, let us consider the number of documents giving the opinions of users on a camera or on a movie. research topic addressed by different communities (e.g. Data Mining, Text Mining, Linguistic). Usually proposed approaches try to find positive or negative opinion features to build training sets and apply classification algorithms (based on several linguistic techniques) to automatically classify new documents extracted from the Web. Furthermore, they associate opinion semantic orientation with adjectives [15, 14, 16, 5, 7]. One of important issue is thus to define the list of relevant adjectives. Use either general dictionaries or expert in order to get positive and negative adjectives. Nevertheless, these approaches suffer the following drowback: for a specific domain either the adjective does not exist or its meaning could be different from another domain. Let consider the two following sentences "*The picture quality of this camera is high*" and "*The ceilings of the building are high*". In the first one, (i.e. an expressed opinion on a movie), the adjective *high* is considered as positive. In the second sentence (i.e. a document on architecture), this adjective is neutral. This example shows that an adjective is very correlated with a particular domain. In the same way, if we find that *a chair is comfortable*, such adjective will never be used when talking about movies. In this paper we would like to answer the two following questions: Is it possible to automatically extract from the web a training set for a particular domain? and how to extract sets of positive and negative adjectives?

The rest of the paper is organized as follows. Section 2 propose a brief overview of existing approaches for extracting opinions. Our approach, called AMOD (*Automatic Mining of Opinion Dictionaries*) is described in section 3. Conducted experiments performed on real data sets from blogs are provided in section 4. Section 5 concludes the paper.

2. RELATED WORK

As previously mentioned, most approaches consider adjectives as main source to express subjective meaning in a document. Generally speaking, semantic orientation of a document is determined by the combined effect of adjectives found in a document, on the basis of an anotated dictionary of adjectives which contain 3596 words labeled as positive or negative (i.e. Inquirer [13] or HM containing 1336 adjectives [5]). More recently, new approaches have enhanced

[*]Ecole des Mines d'Ales, Laboratoire de Génie Informatique et d'Ingénierie de Production

[†]Laboratoire d'Informatique de Robotique et de Microélectronique de Montpellier, Centre National de la Recherche Scientifique

adjective learning with such system as WordNet [8]. These approaches add synonyms and antonyms automatically [2]; or extract opinion related words [16, 6]. Final result Quality is strongly related to available dictionaries. Moreover these approaches are not able to detect differences between subject domains (for example the semantic orientation of the adjective "high"). To avoid this problem, more recent approaches use statistical methods based on adjective co-occurrence with an initial set of seed words. General principle is as follows: beginning with a set of positive and negative words (i.e. *good*, *bad*), try to extract adjectives situated nearby each other according to distance measure. The underlying assumption is that a positive adjective appears more frequently besides a positive seed word, and a negative adjective appears more frequently besides a negative seed word. Even if these approaches are efficient, they encounter the same weaknesses as previous techniques regarding domain related words.

3. THE AMOD APPROACH

This section presents an overview of the AMOD approach. The general process has three main phases (C.f. figure 1).

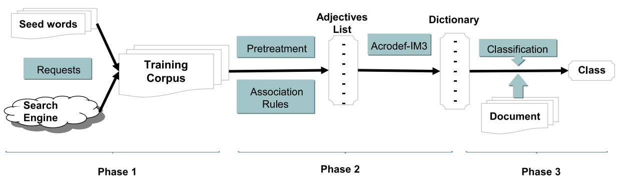

Figure 1: The main process of the AMOD approach

- **Phase 1: Corpora Acquisition learning phase.** This phases aims at automatically extracting, for a specific domain, documents containing positive and negative opinions from the Web.
- **Phase 2: Adjective extraction phase.** In this phase, we automatically extract sets of relevant positive and negative adjectives.
- **Phase 3: Classification.** The goal of this phase is to classify new documents by using the sets of adjectives obtained in the previous phase.

In this paper we particularly focus on the first two points. Classification task uses very simple operations, and will be enhance later on.

3.1 Phase 1: Corpora Acquisition learning phase

In order to find relevant adjectives, we first focus on the automatic extraction of a training set for a specific domain. So, we consider 2 sets P and N of seed words with respectively positive and negative semantic orientations as in [15].

$$P = \{good, nice, excellent, positive, fortunate, correct, superior\}$$

$$Q = \{bad, nasty, poor, negative, unfortunate, wrong, inferior\}$$

For each seed word, we use a search engine and apply a special request specifying: the application domain d, the seed word we are looking for and the words we want to avoid. For example, if we consider the Google search engine, to get movie opinions containing the seed word *good*, the following request is sent "+opinion +review +movies +*good* -bad -nasty -poor -negative -unfortunate -wrong -inferior". The results given by this request will be opinion documents on cinema containing the word *good* and without the following words: bad, nasty, poor, ... inferior. which is specialized in blog search. Therefore, for each positive seed word (resp. negative) and for a given domain, we automatically collect K documents where none of the negative set (resp. positive) appears. This operation build 14 learning corpora: 7 positives and 7 negatives.

3.2 Phase 2: Adjective extraction phase

In the corpora built in the previous phase we are provided with documents containing domain relevant seed adjectives. Therefore, with these domain relevant documents, this phase focuses on extracting adjectives which are highly correlated with seed adjectives. So, from the collected corpora, we compute correlations in collected documents between seed words and adjectives to enrich the seed word sets with new opinion and domain relevant adjectives. However, to avoid false positive or false negative adjectives we add new filter steps. We present these steps in the following subsections.

3.2.1 Preprocessing and association rules steps

To compute correlations between adjectives which will enrich an opinion dictionary, we must determine the Part-of-Speech tag (Verb, Noun, Adjective, etc.) of each word from the training corpus. So, we use the tool Tree Tagger [12], which automatically gives for each word of a text a Part-of-Speech tag and convert it to its lemmatised form. As in [14, 16, 5, 7], we consider adjectives as representative words to specify opinion. We then keep only the adjectives embedded in the documents from the TreeTagger results. Then we search for associations between adjectives from documents and seed words coming from positive and negative seed sets. The goal is to find if new adjectives are associated with the same opinion polarity than seed words. In order to get the correlations, we adapt an association rule algorithm [1] to our concern. More formally, let $I = \{adj_1, ...adj_n\}$ a set of adjectives, and D a set of sentences, where each sentence corresponds to a subset of elements of I. An association rule is thus defined as X→Y, where X⊂I, Y⊂I, and X ∩ Y = ⊘. The support of a rule corresponds to the percentage of sentences in D containing X∪Y. The rule X→Y has a confidence ratio c, if c% ofsentences from D containing X also contain Y.

Sentences could be part of text separated by some punctuation marks. Nevertheless in order to get more relevant adjectives we consider the following hypothesis: the more an adjective is close to a seed one, the more this adjective has the same semantic orientation. We thus define sentences by considering window sizes (WS). WS corresponds to the distance between a seed word and an adjective. For instance, if WS is set to 1 that means that a sentence is composed by one adjective before and one after the seed word. In the following sentence "*The movie is amazing, good acting, a lots of great action and the popcorn was delicious*", by considering the seed adjective *good*, with WS=1, we get the following sentence "*amazing, good, great*" and with WS=2: "*amazing,*

good, great, delicious".
The association rule algorithm is applied both for positive and negative corpora. At the end of this step we are thus provided with rules on adjectives for the positive (resp. negative) corpus. An example of such a rule is: *amazing, good* → *funny* meaning that when, in a sentence we have *amazing* and *good*, then very often (according to a support value s) we can get *funny*.

3.2.2 Filtering step

As we are interested in adjectives strongly correlated with seed words, from the results obtained in the previous step we only keep rules having more than one seed word. We then consider adjectives appearing in both positive and negative lists. Those correlated to several seed words having same orientation and having a high support are kept as learned adjectives only if their number of occurrences in each document of one corpus (e.g. the positive one) is greater than 1 while the number of occurrences in each documents in the other corpus (e.g. the negative one) is lower than 1. Otherwise they are removed.
Finally, to filter associations extracted in the previous step, we use a ranking function in order to delete the irrelevant adjectives associations placed at the end of a list. One of the most commonly used measures to find how to words are correlated (i.e. it exist a co-occurrence relationship between two words) is the Cubic Mutual Information ($MI3$) [4]. This empirical measure based on Church's Mutual Information (MI) [3], enhances the impact of frequent co-occurrences. Our approach relies on the dependence computation of two adjectives based on the number of pages returned by the queries "$adjective_1$ $adjective_2$" and "$adjective_2$ $adjective_1$"[1] on the Web. This dependence is computed in a given context C (e.g. the context $C = \{movies\}$). Then we apply the formula $AcroDef_{MI3}$ (1) described in [11].

$$AcroDef_{MI3}(adj1, adj2) =$$

$$\frac{(nb(\text{"}adj1\ adj2\text{"}\ and\ C) + nb(\text{"}adj2\ adj1\text{"}\ and\ C))^3}{nb(adj1\ and\ C) \times nb(adj2\ and\ C)} \quad (1)$$

3.3 Phase 3: Classification

The last step to consider is to classify each document in a positive or negative opinion. In a first step we use a very simple classification procedure. For each document to classify, we calculate its positive or negative orientation by computing the difference between the number of positive and negative adjectives, from both the previous lists, encountered in the studied document. We count the number of positive adjectives, then the number of negative adjectives, and we simply compute the difference. If the result is positive (resp. negative), the document will be classified in the positive class (resp. negative). Otherwise, the document is considered as neutral.
In order to improve the classification, we extend our method to consider adverbs used for inverting the polarities (e.g. not, neither nor, ..). For instance, let us consider the following sentence: *The movie is not bad, there is a lot of funny moment.* The adverb *not* inverses the polarity of the adjectif *bad* while *funny*, too far from *not*, is not affected. Furthermore, for the following adverbs: very, so, too we increase the semantic orientation degree by 30%.

[1]Here we consider that the request is done on Google and then brackets stands for looking for the real string respecting the order between adjectives

4. EXPERIMENTS

In this section, we present experiments conducted to validate our approach. First we present the adjective learning phase then classification results, and finally we compare our method to a supervised machine learning classification method.

Documents are extracted from the research engine BlogGooglesearch.com. We extract documents related to expressed opinions for the "cinema" domain. Seed words and applied requests are those already mentioned in section 3.1. For each seed word, we have limited the number of extracted documents by the search engine to 300. We then transform these documents, from HTML format to text format and we then use TreeTagger to keep only adjectives.

In order to study the best distance between seed words and adjectives to be learned, we have tested different values for the Window Size parameter from 1 to 3. Then, to extract correlation links between adjectives, we use the Apriori algorithm[2]. In conducted experiments, support value has been ranged from 1 to 3%. We get for each support value, two lists: one negative and one positive. As was stated in previous section, we discard from these lists adjectives being commun to both lists (for the same support value) and those which are correlated to only one seed word. To discard useless and frequent adjectives we used AcroDef$_{MI3}$ measure with a threshold value fixed experimentally to 0.005.

In order to test the quality of the learned adjectives, we use for the classification the Movie Review Data from NLP Group, Cornell University[3]. This database possesses 1000 positives and 1000 negatives opinions extracted from the Internet Movie Database[4]. We intentionally use a test corpora very different in nature from the training corpora (i.e. blogs), to show the stability of our method.

	Positives	Negatives	PL	NL
Seed List	**66,9%**	**30,4%**	7	7

Table 1: Classification of 1000 positive and negative documents with seed words

Table 1 shows classification results by considering only seed words (i.e. without applying the AMOD approach) on the negative and positive corpora. PL (resp. NL) correspond to the number of adjectives (in our case, this number corresponds to the number of seed words). Table 2 (resp. table 3), shows results obtained with learned adjectives using AMOD after classifying positive (resp. negative) documents. Column WS stands for the distances and column S corresponds to support values. The value 7 + 12 from the PL

[2]http://fimi.cs.helsinki.fi/fimi03/
[3]http://www.cs.cornell.edu/people/pabo/movie-review-data/
[4]http://www.imdb.com/

WS	S	Positive	PL	NL
1	1%	**67,2%**	7+12	7+20
	2%	60,3%	7+8	7+13
	3%	65,6%	7+6	7+1
2	1%	57,6%	7+13	7+35
	2%	56,8%	7+8	7+17
	3%	68,4%	7+4	7+4
3	1%	28,9%	7+11	7+48
	2%	59,3%	7+4	7+22
	3%	67,3%	7+5	7+11

Table 2: Classification of 1000 positive documents with learned adjectives

WS	S	Negative	PL	NL
1	1%	**39,2%**	7+12	7+20
	2%	46,5%	7+8	7+13
	3%	17,7%	7+6	7+1
2	1%	49,2%	7+13	7+35
	2%	49,8%	7+8	7+17
	3%	32,3%	7+4	7+4
3	1%	76,0%	7+11	7+48
	2%	46,7%	7+4	7+22
	3%	40,1%	7+5	7+11

Table 3: Classification of 1000 negative documents with learned adjectives

column at the first line indicates that we have 7 seed adjectives and 12 learned adjectives. As we see, our method allows, in case of a negative document, a much better classification result. For positive documents, the difference is less important but as illustrated in table 4, the learned adjectives appear in a very significant manner in the test documents. As expected if we compare the number of learned adjectives, the best results come with WS value of 1. This experiment confirm hypothesis on adjective proximity in opinion expression [15]. In table 2 and 3, we see that positive and nega-

positive seeds		**negative seeds**	
Adjective	Nb of occ.	Adjectives	Nb of occ.
Good	2147	Bad	1413
Nice	184	Wrong	212
Excellent	146	Poor	152
Superior	37	Nasty	38
Positive	29	Unfortunate	25
Correct	27	Negative	22
Fortunate	7	Inferior	10

Table 4: Occurrences of positive and negative seed adjectives for WS=1 and S=1%

tive learned adjective numbers may strongly vary according to support value. For example, if support value is 1% and WS=3, we get 11 learned positive adjectives and 48 negative ones. A thorough analyse of the results shows that most of the negative adjectives were frequent and useless adjectives. Results obtained by applying the AcroDef$_{MI3}$ measure as an adjective filter are plotted in tables 6 and 7, were we consider results obtained only with WS=1 and S=1%. The proportion of well classified documents with our approach ranges from 66.9% to 75.9% for positive adjectives and from 30.4% to 57.1% for negative adjectives. To enhance our method and extract the best discriminative adjectives, we

Learned positive adjectives			
Adjective	Nb of occ.	Adjective	Nb of occ.
Great	882	Hilarious	146
Funny	441	Happy	130
Perfect	244	Important	130
Beautiful	197	Amazing	117
Worth	164	Complete	101
Major	163	Helpful	52

Table 5: Occurrences of positive learned adjectives for WS=1 and S=1%

Learned negative adjectives			
Adjectives	Nb of occ.	Adjectives	Nb of occ.
Boring	200	Certain	88
Different	146	Dirty	33
Ridiculous	117	Social	33
Dull	113	Favorite	29
Silly	97	Huge	27
Expensive	95		

Table 6: Occurrences of negative learned adjectives for pour WS=1 et S=1%

WS	S	Positive	Negative	PL	NL
1	1%	**75,9%**	**57,1%**	7+11	7+11

Table 7: Classification of 1000 positive and negative documents with learned adjectives and AcroDef$_{MI3}$

have applied the following method:

- We enrich the seed word list with adjectives learned with the previous application of AMOD. We then get new seed word lists.
- Then, we apply the AMOD approach on the new lists to learn new adjectives.
- To evaluate the new lists, we apply the classification procedure on the test dataset.

This method is repeated until no more new adjectives are learned. Learned adjectives when applying for the first time

Learned positive adj.		**Learned negative adj.**	
Adjectives	Nb of occ.	Adjectives	Nb of occ.
Interesting	301	Commercial	198
comic	215	Dead	181
Wonderful	165	Terrible	113
Successful	105	Scary	110
Exciting	88	Sick	40

Table 8: Learned adjective occurrences with the first reinforcement for WS=1 and S=1%

this reinforcement method are showed in table 8. Learned adjectives considered as relevant and representative will thus enrich our adjective set. Obtained results for the classification are showed in table 9. The ratio of well attributed positive documents has been improved with the second reinforcement learning phase from 75.9 to **78.1%**.

WS	S	Positive	Negative	PL	NL
1	1%	**78,1%**	**54,9%**	7+16	7+16

Table 9: Classification of 1000 positive and negative documents with learned adjectives and AcroDef$_{MI3}$

Learned adjectives with the first reinforcement are then added to the previous seed word lists and the process is repeated. The second reinforcement phase produces new adjectives (C.f. Table 10).

Learned positive adj.	
Adjectives	Nb of occ.
special	282
entertaining	262
sweet	120

Learned negative adj.	
Adjectives	Nb of occ.
awful	109

Table 10: Learned adjective occurrences with the second reinforcement for WS=1 et S=1%

Table 11 shows that the classification result for positive documents has improved from 78.1% to **78.7%**, for the same dataset test. But results are slightly lower for negative documents. We may explain this by the too elementary classification procedure lying on adjective occurrence number. The learned adjective list shows that occurrence figures for positive learned adjectives is notably greater than those for learned negative adjectives. This significantly influences our classification results.

WS	S	Positive	Negative	PL	NL
1	1%	**78,7%**	**46,7%**	7+16	7+16

Table 11: Classification of 1000 positive and negative documents with learned adjectives and AcroDef$_{MI3}$

We have improved our classification method by adding the different forms of negation presented in previous section. Our results on 1000 positive texts classification have been enhanced from 78.7% to **82.6%** and from 46.7% to **52.4%** for the 1000 negative texts as shown in table 12.

A new application of the reinforcement learning phase does not produce any new adjectives. At the end of the process we obtain two relevant and discriminatory adjective lists (C.f. Table 13) for the *cinema* domain.

WS	S	Positive	Negative	PL	NL
1	1%	**82,6%**	**52,4%**	7+19	7+17

Table 12: Classification of 1000 positive and negative documents classification with learned adjectives, AcroDef$_{MI3}$ and negation

Positive adjective list	
Adjective	Adjective
Good	Great
Nice	Funny
Excellent	Perfect
Superior	Beautiful
Positive	Worth
Correct	Major
Fortunate	Interesting
Hilarious	Comic
Happy	Wonderful
Important	Successful
Amazing	Exciting
Complete	Entertaining
Special	Sweet

Negative adjective list	
Adjective	Adjective
Bad	Boring
Wrong	Different
Poor	Ridiculous
Nasty	Dull
Unfortunate	Silly
Negative	Expensive
Inferior	Huge
Certain	Dead
Dirty	Terrible
Social	Scary
Favorite	Sick
Awful	Commercial

Table 13: Adjective lists for WS=1 and S=1% for the domain "*cinema*"

In this experiment, we want to know how many documents are required to produce a stable and robust training set? We thus applied the AMOD training method several times. Each time we have increased by 50 the number of collected documents until we get a stability on the number of learned adjectives.

The figure 2 depicts the relationship between the size of

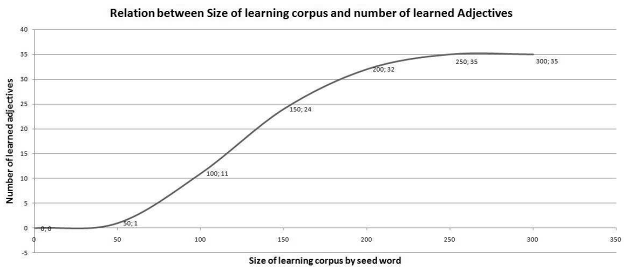

Figure 2: Relation between the size of training corpus and the number of learned adjectives

the corpus and the number of learned adjectives. As we can notice, above 2800 documents (i.e. 200 documents for each seed word) we do not learn much new adjectives.

Finally we conducted some experiments in order to compare the results obtained with a traditional classification method and with our approach. The classification method used for experiments is COPIVOTE [9]. This approach use a training corpus and a system of vote with several classifiers (SVM, ngrams, ...). Experiments have been done on the same datasets for learning and tests.

To compare our results, we used the well known FScore measure [10]. FScore is given by the following formula:

$$Fscore = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

Fscore is a compound between Recall and Precision, giving the same weight to each measure. *Precision* and *Recall* are defined as follows:

$$Recall_i = \frac{Nb\ documents\ rightly\ attributed\ to\ class\ i}{Nb\ documents\ of\ class\ i}$$

$$Precision_i = \frac{Nb\ documents\ rightly\ attributed\ to\ class\ i}{Nb\ documents\ attributed\ to\ class\ i}$$

Documents :	Positives	Negatives
FScore COPIVOTE :	60,5%	60,9%
FScore AMOD :	**71,73%**	**62,2%**

Table 14: Fscore classification results for 1000 negative and positive test documents with COPIVOTE and AMOD

Table 14 shows that our approach performs better for both positive case (**71,73%** vs. 60,5%) and negative case (**62,2%** vs. 60,9%). Generally the COPIVOTE method is very efficient for text classification (i.e. based on a voting system, the best classification method is selected), but is penalized by the large differences between test and training corpora.

In order to verify that our approach is suitable for other domains we performed some experiments with a totally different domain: "car". Positive and Negative corpora are obtained from BlogGooglesearch.com with the keyword "*car*". To validate acquired knowledge in training phase, we use in test phase 40 positive documents coming from *www.epinions.com*.
Applying the AMOD approach, with WS=1 and support = 1%, after AcroDef$_{IM3}$ filter and reinforcement training gives the results showed in table 15.
We get the following positive adjectives: good, nice, excellent, superior, positive, correct, fortunate, professional, popular, luxurious, secured, great, full, efficient, hard, fast, comfortable, powerful, fabulous, economical, quiet, strong, several, lovely, successful, amazing, maximum, first, active, beautiful, wonderful, practical.
And we get the following negative adjectives: bad, wrong, poor, nasty, unfortunate, negative, inferior, horrible, boring, unsecured, uncomfortable, expensive, ugly, luck, heavy, dangerous, weird.

Method	WS	S	Positive	PL	NL
Seed words only	1	1%	**57,5%**	7+0	7+0
with learned words	1	1%	**95%**	7+26	7+10

Table 15: 40 positive documents Classification with seed adjectives only and with learned adjectives, AcroDef$_{IM3}$ and negation filters

Compared to previous experiments the two training sets are similarly constituted from blogs. Our approach gives better results on similar data sets.

5. CONCLUSION

In this paper, we proposed a new approach for automatically extracting positive and negative adjectives in the context of the opinion mining. Experiments conducted on training sets (blogs vs. cinema reviews) show that with our approach we are able to extract relevant adjectives for a specific domain. Future works may be manifold. First, our method depend on good quality of documents extracted from blogs. We want to extend our training corpora method by applying text mining approaches on collected documents in order to minimize lower noisy texts. Second, in this work we focused on adjectives, we plan to extend the extraction task to other categories.

6. REFERENCES

[1] R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *VLDB'94*, 1994.

[2] A. Andreevskaia and S. Bergler. Semantic tag extraction from wordnet glosses. 2007.

[3] K. Church and P. Hanks. Word association norms, mutual information, and lexicography. In *Computational Linguistics*, volume 16, pages 22–29, 1990.

[4] D. Downey, M. Broadhead, and O. Etzioni. Locating complex named entities in web text. In *Proceedings of IJCAI'07*, pages 2733–2739, 2007.

[5] V. Hatzivassiloglou and K. McKeown. Predicting the semantic orientation of adjectives. In *In Proceedings of 35th Meeting of the Association for Computational Linguistics*, Madrid, Spain, 1997.

[6] M. Hu and B. Liu. Mining and summarizing customer reviews. In *In Proceedings of KDD'04, ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Seattle, WA, 2004.

[7] J. Kamps, M. Marx, R. J. Mokken, and M. Rijke. Using wordnet to measure semantic orientation of adjectives. In *In Proceedings of LREC 2004, the 4th International Conference on Language Resources and Evaluation*, pages 174–181, Lisbon, Portugal, 2004.

[8] G. Miller. Wordnet: A lexical database for english. In *Communications of the ACM*, 1995.

[9] M. Plantié, M. Roche, G. Dray, and P. Poncelet. Is a voting approach accurate for opinion mining? In *Proceedings of the 10th International Conference on Data Warehousing and Knowledge Discovery (DaWaK '08), Torino Italy*, 2008.

[10] V. Risbergen. Information retrieval, 2nd edition. In *Butterworths*, London, 1979.

[11] M. Roche and V. Prince. *AcroDef:* A Quality Measure for Discriminating Expansions of Ambiguous Acronyms. In *Proceedings of CONTEXT, Springer-Verlag, LNCS*, pages 411–424, 2007.

[12] H. Schmid. Treetagger. In *TC project at the Institute for Computational Linguistics of the University of Stuttgart*, 1994.

[13] P. Stone, D. Dunphy, M. Smith, and D. Ogilvie. The general inquirer: A computer approach to content analysis. Cambridge, MA, 1966. MIT Press.

[14] M. Taboada, C. Anthony, and K. Voll. Creating semantic orientation dictionaries. 2006.

[15] P. Turney. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *In Proceedings of 40th Meeting of the Association for Computational Linguistics*, pages 417–424, Paris, 2002.

[16] K. Voll and M. Taboada. Not all words are created equal: Extracting semantic orientation as a function of adjective relevance. 2007.