Topic-Related Polarity Classification of Blog Sentences

Michael Wiegand and Dietrich Klakow

Spoken Language Systems, Saarland University, Germany,
{Michael.Wiegand|Dietrich.Klakow}@lsv.uni-saarland.de

Abstract. Though polarity classification has been extensively explored at various text levels and domains, there has been only comparatively little work looking into topic-related polarity classification. This paper takes a detailed look at how sentences expressing a polar attitude towards a given topic can be retrieved from a blog collection. A cascade of independent text classifiers on top of a sentence-retrieval engine is a solution with limited effectiveness. We show that more sophisticated processing is necessary. In this context, we not only investigate the impact of a more precise detection and disambiguation of polar expressions beyond simple text classification but also inspect the usefulness of a joint analysis of topic terms and polar expressions. In particular, we examine whether any syntactic information is beneficial in this classification task.

1 Introduction

Though polarity classification has been extensively explored at various text levels and domains, there has only been comparatively little work examining topic-related polarity classification. This paper takes a detailed look at how sentences expressing a polar attitude towards a given topic can be retrieved from a blog collection. This means that for a specific topic, we do not only extract opinionated sentences but also distinguish between positive and negative polarity. For example, an appropriate positive opinion for a topic, such as *Mozart*, is Sentence (1) and an example of a negative opinion Sentence (2). Traditional *factoid* retrieval is inappropriate for this task, since arbitrary sentences regarding a specific topic are retrieved. On a query, such as {*topic:* ***Mozart****, target polarity:* ***positive***} a state-of-the-art method would probably highly rank Sentences (1)-(3), thus failing to single out mere factual statements, such as Sentence (3), and subjective statements with opposing polarity, such as Sentence (2).

(1) **positive statement:** My argument is that it is *pointless*$^-$ to ordinary mortals like you and me to discuss why Mozart was a ***genius***$^+$.
(2) **negative statement:** I have to say that I [don't ***like***$^+$]$^-$ Mozart.
(3) **neutral statement:** Wolfgang Amadeus Mozart's 250th birthday is coming up on the 27th of this month.

We show that though simple text classification can enhance the performance of factoid retrieval a more sophisticated approach is preferable. For example, Sentence (1) is ambiguous judged by the presence of polar expressions, i.e. words

containing a prior polarity, since there is both a positive and a negative polar expression, i.e. *pointless* and *genius*. Bag-of-words classifiers might therefore mislabel this sentence. A classification which jointly takes topic term and the polar expressions into account, on the other hand, results in a correct classification. For example, the closest polar expression, i.e. *genius*, is the expression which actually relates to the topic. Not only spatial distance but also syntactic information can resolve this ambiguity. In the current example, there is a direct syntactic relationship, i.e. a *subject-of* relationship, between the topic term and the polar expression relating to it. Usually syntactic relation features are more precise but also much sparser than proximity features.

Not only is it important to identify the polar expression within a sentence which actually relates to the polar expression but also to interpret a polar expression correctly in its context. In Sentence (2), the only polar expression has a positive prior polarity but since it is negated its contextual polarity is negative.

All these observations suggest that there are several sources of information to be considered which is why we examine features incorporating polarity information extracted from a large polarity lexicon, syntactic information from a dependency parse and surface-based proximity. In particular, we address the issue whether syntactic information is beneficial in this task.

2 Related Work

The main focus of existing work in sentiment analysis has been on plain polarity classification which is carried out either at document level [1], sentence level [2], or expression level [3]. There has also been quite some work on extracting and summarizing opinions regarding specific features of a particular product, one of the earliest works being Hu and Liu [4]. Unlike our paper, the task is usually confined to a very small domain. Moreover, the plethora of positively labeled data instances allow the effective usage of syntactic relation patterns.

Santos et al. [5] show that a Divergence From Randomness proximity model improves the retrieval of subjective documents. However, neither an evaluation on sentence level and nor an evaluation of polarity classification is conducted.

The work most closely related to this paper is Kessler and Nicolov [6] who examine the detection of targets of opinions by using syntactic information. Whereas Kessler and Nicolov [6] discuss how to detect whether two entities are in an opinion-target relationship – already knowing that there is such a relationship in the sentence to be processed – we do not conduct an explicit entity extraction but classify whether or not a sentence contains an opinion-target relationship. Unlike Kessler and Nicolov [6], we also restrict the opinion-bearing word to be of a specific polarity. Thus, we can use knowledge about polar expressions in order to predict an opinion-target relationship in a sentence. This change in focus raises the question whether for a sentence-level classification a similar amount of syntactic knowledge is necessary or whether sufficient information can be drawn from more surface-based features and lexical knowledge of prior polarity. Moreover, we believe that our results are more significant for realistic scenarios

like *opinion question answering*, since our settings are more similar to such a task than the ones presented by Kessler and Nicolov [6].

3 The Dataset

The dataset we use in this paper is a set of labeled sentences retrieved from relevant documents of the TREC Blog06 corpus [7] for TREC Blog 2007 topics [8]. The test collection contains 50 topics. For each topic we formulate two separate queries, one asking for positive opinions and another asking for negative opinions. In the final collection we only include queries for which there is at least one correct answer sentence. Thus, we arrive at 86 queries of which 45 ask for positive and 41 ask for negative opinions. The sentences have been retrieved by using a language model-based retrieval [9]. Each sentence from the retrieval output has been manually labeled. An annotator judged whether a sentence expresses an opinion with the target polarity towards a specific topic or not. Difficult cases have been labeled after discussion with another annotator. The annotation is strictly done at sentence level i.e. no information of surrounding context is taken into consideration. This means that each positively labeled sentence must contain some (human recognizable) form of a polar expression and a topic-related word. Our decision to restrict our experiments to sentence level is primarily to reduce the level of complexity. We are aware of the fact that we ignore inter-sentential relationships, however, Kessler and Nicolov [6] state that on their similar dataset 91% of the opinion-target relations are within the same sentence.

The proportion of relevant sentences containing at least one topic term is 97% which is fairly high. Although 71% of the relevant sentences contain a polar expression of the target polarity according to the polarity lexicon we use, in 50% of the sentences there is also at least one polar expression with opposing polarity. The joint occurrence of a polar expression matching with the target polarity and a topic term is no reliable indicator of a sentence being relevant, either. Only approximately 17% of these cases are correct. The entire dataset contains 25651 sentences of which only 1419 (i.e. 5.5%) are relevant indicating a fairly high class imbalance. This statistical analysis suggests that the extraction of correct sentences is fairly difficult.

4 Features

4.1 Sentence Retrieval, Topic Feature and Text Classifiers

Our simplest baseline consists of a cascade of a sentence-retrieval engine and two text classifiers, one to distinguish between objective and subjective content, and another to distinguish between positive and negative polarity. We employ stemming and only consider unigrams as features. The two text classifiers are run one after another on the ranked output. Rather than combining the scores of the classifiers with the retrieval score in order to re-rank the sentences, we

maintain the ranking of the sentence retrieval and delete all sentences being objective and not matching the target polarity. This method produces better results than combining the scores by some form of interpolation and does not require any parameter estimation. This hierarchical classification (subjectivity detection followed by polarity classification) is commonly used in opinion mining [3, 10].

We also consider a separate topic feature which counts the number of topic terms within a sentence since this feature scales up better with the other types of features we use for a learning-based ranker than the sentence retrieval score.

4.2 Polarity Features

For our polarity features, we mainly rely on the largest publicly available subjectivity lexicon [3]. We chose this lexicon since, unlike other resources, it does not only have part-of-speech labels attached to polar expressions, thus allowing a crude form of disambiguation[1], but also distinguishes between *weak* and *strong* expressions. As a basic polarity feature (`PolMatch`), we count the number of polar expressions within a candidate sentence which match the target polarity. Since this basic polarity feature is fairly coarse, we add further polarity features which have specific linguistic properties. We include a feature for strong polar expressions (`StrongPolMatch`) and a feature for polar expressions being modified by an intensifier (`IntensPolMatch`), such as *very*. We suspect that a strong polar expression, such as *excellent*, or an intensified polar expression, such as *very nice*$^+$, might be more indicative of a specific polarity than the occurrence of any plain polar expression. We use the list of intensifiers from Wilson et al. [3]. Furthermore, we distinguish polar expressions with regard to the most frequent part-of-speech types (`Pol`*`POS`*`Match`), being *nouns*, *verbs* and *adjectives*[2]. Some parts of speech, for instance adjectives, are more likely to carry polar information than others [1]. Table 1 lists all polarity features we use. It also includes some combined features of the features mentioned above, i.e. `StrongPol`*`POS`*`Match`, `IntensPol`*`POS`*`Match`, and `StrongIntensPol`*`POS`*`Match`.

4.3 Negation Modeling

A correct contextual disambiguation of polar expressions is important for topic-related sentence-level polarity classification since the instances to be classified are rather sparse in terms of polarity information. Therefore, we conduct negation modeling. Our negation module comprises three steps. In the first step, all potential negation expressions of a sentence are marked. In addition to common negation expressions, such as *not*, we also consider *polarity shifters*. Polarity shifters are weaker than ordinary negation expressions in the sense that they often only reverse a particular polarity type[3]. In the second step, all the potential negation expressions are disambiguated. All those cues which are not within a

[1] e.g. thus we can distinguish between the preposition *like* and the polar verb *like*

[2] We subsume *adverbs* by this type as well.

[3] e.g. the shifter *abate* only modifies negative polar expressions as in *abate the damage*

negation context, e.g. *not* in *not just*, are discarded. In the final step, the polarity of all polar expressions occurring within a window of five words[4] after a negation expression is reversed. We use the list of negation expressions, negation contexts and polarity shifters from Wilson et al. [3].

Table 1. List of polarity features.

Feature	**Abbreviation**
Number of polar expressions within sentence with matching polarity *(basic polarity feature)*	`PolMatch`
Number of strong polar expressions within sentence with matching polarity	`StrongPolMatch`
Number of intensified polar expressions within sentence with matching polarity	`IntensPolMatch`
Number of strong and intensified polar expressions within sentence with matching polarity	`StrongIntensPolMatch`
Number of polar nouns/verbs/adjectives within sentence with matching polarity	`Pol`*`POS`*`Match`
Number of strong polar nouns/verbs/adjectives within sentence with matching polarity	`StrongPol`*`POS`*`Match`
Number of intensified polar nouns/verbs/adjectives within sentence with matching polarity	`IntensPol`*`POS`*`Match`
Number of strong and intensified polar nouns/verbs/adjectives within sentence with matching polarity	`StrongIntensPol`*`POS`*`Match`

4.4 Spatial Distance

Textual proximity provides additional information to the previously mentioned features, as it takes the relation between polar expression and topic term into account. In Sentence (1), for example, the positive polar expression *genius* is closest to the topic term *Mozart*, which is an indication that the sentence describes a positive opinion towards the topic.

We encoded our distance feature as a binary feature with a threshold value[5]. This gave much better performance than encoding the explicit values in spite of attempts to scale this feature with the remaining ones. Since we do not have any development data, we had to determine the appropriate threshold values on our test data. The threshold value is set to 8.[6] Since all feature sets containing this distance feature supported the same threshold value, we have strong reasons to believe that the value chosen is fairly universal. We also experimented with a more straightforward distance feature which checks whether the closest polar

[4] This threshold value is taken from Wilson et al. [3].

[5] i.e. the feature is active if a polar expression and topic term are sufficiently close

[6] The threshold may appear quite high. However, given the fact that the average sentence length in this collection is at approx. 30 tokens and that there is a tendency of topic terms to be sentence initial or final, this value is fairly plausible.

expression to the topic term matches the target polarity. However, we did not measure any notable performance gain by this feature.

4.5 Syntactic Features from a Dependency Path

In addition to polarity and distance features we use a small set of syntactic features. By that we mean all those features that require the presence of a syntactic dependency parse. This set of features supplements both of the other feature types.

Similar to the polarity features are the two *prominence features* we use. Their purpose is to indicate the overall polarity of a sentence. On the news domain, they have already been shown to improve plain polarity classification [2]. Each polar expression can be characterized with its depth within the syntactic parse tree. Depth is defined as the number of edges from the node representing the polar expression to the root node. Usually, the deeper a node of a polar expression is, the less prominent it is within the sentence [2]. Similar to the distance feature, we define a binary feature (`LowDepth`) which is active if a polar expression has a sufficiently low depth. The threshold value is set to 5.[7] The main predicate (`MainPred`), too, is usually very indicative of the overall polarity of a sentence. Sentence (2) is a case where the main predicate coincides with the correct polarity.

The shortcoming of the prominence features is that they do not consider the relation of a polar expression to a mentioning of a topic but just focus on the overall polarity of a sentence. The overall polarity, however, does not need to coincide with the polarity towards a topic term as Sentence (4) illustrates:

(4) The strings $[screwed\ up]^{-}_{mainPred}$ the concert, in particular, my ***favorite***$^{+}$ scores by Mozart.
(overall polarity: negative, polarity towards Mozart: positive)

Moreover, textual proximity is sometimes misleading as to discover the correct relation between polar expression and topic term as illustrated by Sentence (5) where the polar expression with the shortest distance to the topic term is not the polar expression which relates to it.

(5) Mozart, it is $save^{+}$ to say, ***failed***$^{-}$ to bring music one step forward.

That is why we use a set of features describing the dependency relation path between polar expression and topic term. Unlike previous work [6], we do not focus on the relation labels on the path due to a heavy data-sparseness we experienced in initial experiments. Instead, we define features on the configuration of the path. The advantage of this is that these features are more general.

We use one feature that counts the number of paths with a direct dominance relationship (`ImmediateDom`), i.e. the paths between polar expressions and topic terms which are directly connected by one edge. All common relationships, such as *subject-verb*, *verb-object* or *modifier-noun* are subsumed by this feature. We

[7] The large value for the depth feature can be explained by the fact that MINIPAR uses auxiliary nodes in addition to the nodes representing the actual words.

Table 2. List of syntactic features.

Syntactic Prominence Features	
Feature	**Abbreviation**
Number of matching polar expressions with low depth within the syntactic parse tree	`LowDepth`
Is the main predicate of the sentence a matching polar expression?	`MainPred`
Syntactic Relation Features	
Feature	**Abbreviation**
Number of paths with an immediate dominance relationship between topic term and matching polar expression	`ImmediateDom`
Number of paths with a dominating relationship between topic term and matching polar expression	`Dom`
Number of paths where topic term dominates matching polar expression	`TopicDomPol`
Number of paths where topic term is dominated by matching polar expression	`PolDomTopic`
Number of paths between matching polar expression and topic term which are contained within the same event structure	`SameEvent`
Number of paths between matching polar expression and topic term which do not cross the root node	`NoCrossRoot`

also assume that, in general, any dominance relationship (`Dom`) is more indicative than other paths[8]. Furthermore, we distinguish between the two cases that the topic term dominates the polar expression (`TopicDomPol`) and the reverse relation (`PolDomTopic`).

Often a sentence contains more than one statement. A polar expression is less likely to refer to a topic term in case they appear in different statements. We account for this by two additional features. The first counts the number of paths within a sentence between polar expressions and topic terms which are within the same event structure (`SameEvent`). For this feature, we exclusively rely on the event-boundary annotation of a sentence by the dependency parser we use, i.e. Minipar [11]. Two nodes are within the same event structure, if the they have the same closest event-boundary node dominating them[9]. Additionally, we define a feature which counts the number of paths which do not cross the root node (`NoCrossRoot`). The root node typically connects different clauses of a sentence.

Table 2 summarizes all the different syntactic features we use. In order to familiarize yourself with the features, Figure 1 illustrates a sentence with two candidate paths and the feature updates associated with both paths.

5 Experiments

We report statistical significance on the basis of a paired t-test using 0.05 as the significance level on a 10 fold crossvalidation. For sentence retrieval, we used the language model-based retrieval engine from Shen et al. [9]. The text

[8] i.e. paths which go both up and down a tree

[9] We assume the dominance relationship to be reflexive.

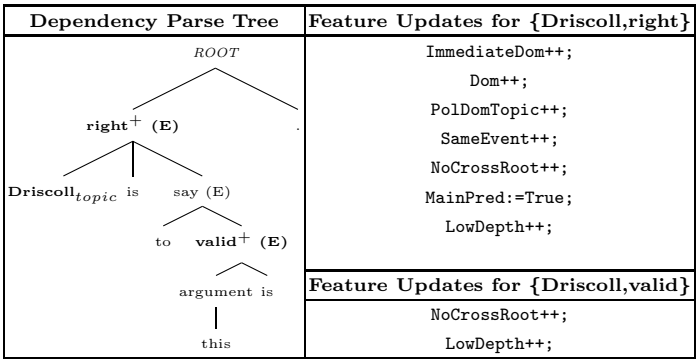

Fig. 1. Illustration of a (simplified) dependency parse tree of the sentence: *Driscoll is right to say this argument is valid* and corresponding updates for syntactic features. Target polarity: *positive*. Nodes which present an event boundary are marked with *(E)*. Note that the pair {*Driscoll,right*} expresses a genuine opinion-target relationship. Consequently, much more features fire.

classifiers were trained using SVMLight [12] in its standard configuration. The subjectivity classifier was trained on the dataset presented by Pang et al. [10]. The polarity classifier was trained on a labeled set of sentences we downloaded from *Rate-It-All*[10]. Both datasets are balanced. The former dataset comprises 5000 sentences and the latter of approximately 6800 sentences per class. Unlike the standard dataset for polarity classification [1], our dataset is not at document level but sentence level[11] and also comprises reviews from several domains and not exclusively the movie domain. Thus, we believe that this dataset is more suitable for our task since we use it for multi-domain sentence-level classification. We use the entire vocabulary of the data collection as our feature set. Feature selection did not result in a significant improvement on our test data.

For ranking we use *Yasmet*[12], a Maximum Entropy ranker. Maximum Entropy models are known to be most suitable for a ranking task [13]. We trained the ranker with 1000 iterations. This gave best performance on all feature sets. For part-of-speech tagging we employ the *C&C tagger*[13] and for dependency parsing Minipar [11]. We evaluate performance by measuring *mean average precision (MAP)*, *mean reciprocal rank (MRR)*, and *precision at rank 10 (Prec@10)*.

Due to the high coverage of topic terms within the set of positive labeled sentences (97%), we discard all instances not containing at least one topic term. This means that the topic feature counting the number of topic terms (see Section 4.1) is no longer an obligatory feature. In fact, we even found in our initial

[10] `http://www.rateitall.com`
[11] We only extracted reviews comprising one sentence.
[12] `http://www.fjoch.com/YASMET.html`
[13] `http://svn.ask.it.usyd.edu.au/trac/candc`

experiments that this gave much better performance than taking all data instances into account and always adding the topic feature.

5.1 Impact of Sentence Retrieval Combined with Text Classification

Table 3 displays the results of sentence retrieval with an opinion and a polarity filter. The results show that both text classifiers systematically increase performance of the retrieval. Only the increase in Prec@10 is marginal and slightly decreases when polarity classification is added to opinion classification.

Table 3. Performance of factoid sentence retrieval in combination with text classifiers.

Features	**MAP**	**MRR**	**Prec@10**
sentence retrieval	0.140	0.206	0.088
sentence retrieval + opinion classifier	0.179	0.247	0.118
sentence retrieval + opinion classifier + polarity classifier	0.220	0.267	0.114

5.2 Comparing Basic Polarity Feature and Text Classifiers

Table 4 compares the baseline using sentence retrieval and text classifiers with the basic polarity feature (i.e. `PolMatch`) using polarity information from the polarity lexicon. The polarity feature outperforms the baseline on all evaluation measures, most notably on MRR and Prec@10. We assume that the text classifiers suffer from a domain mismatch. The polarity lexicon is more likely to encode domain-independent knowledge. Unfortunately, combining the components from the baseline with the polarity feature is unsuccessful. Only the addition of the topic feature (which encodes information similar to the sentence retrieval) to the polarity feature results in a slight (but not significant) increase in MAP. Apparently, the precise amount of word overlap between topic and candidate sentence is less important than in factoid retrieval. Neither do the text classifiers contain any more additional useful information than the polarity feature.

Table 4. Performance text classifiers and basic polarity feature.

Features	**MAP**	**MRR**	**Prec@10**
sentence retrieval with text classifiers	0.220	0.267	0.114
basic polarity feature	0.236	**0.420**	**0.212**
basic polarity feature + topic	**0.239**	0.394	0.200
basic polarity feature + text classifiers	0.227	0.380	0.188
basic polarity feature + topic + text classifiers	0.222	0.390	0.179

5.3 Comparing Polarity Features and Syntactic Features

Table 5 displays the performance of various feature combinations of polarity and syntactic features. Each feature set is evaluated both without negation modeling (*plain*) and with negation modeling (*negation*). When syntactic features are added to the basic polarity feature, there is always an increase in performance. With regard to MAP the improvement is always significant. With regard to Prec@10, only the presence of the relation features results in a significant increase. When the syntactic features are added to all polarity features the increase in performance is similar. The best performing feature set (on average) is the set using all polarity scores and the syntactic relation features. It significantly outperforms the basic polarity feature on all evaluation measures. We, therefore, assume that the syntactic relation features are much more important than the syntactic prominence features. With the exception of very few feature sets, adding negation modeling increases performance as well. However, the improvement is never significant.

Table 5. Performance of polarity features and syntactic features. Each feature set is evaluated without negation modeling (*plain*) and with negation modeling (*negation*).

Features	**MAP**		**MRR**		**Prec@10**	
	plain	**negation**	**plain**	**negation**	**plain**	**negation**
basic polarity feature	0.236	0.245	0.420	0.441	0.212	0.215
basic pol. feat. + syntactic prominence feat.	0.258	0.266	0.477	0.473	0.214	0.216
basic pol. feat. + syntactic relation feat.	0.256	0.269	0.444	0.481	0.237	0.249
basic pol. feat. + all syntactic feat.	0.262	0.278	0.475	0.509	0.237	0.244
all polarity features	0.245	0.257	0.466	0.489	0.207	0.215
all pol. feat. + syntactic prominence feat.	0.261	0.269	0.477	0.474	0.210	0.222
all pol. feat. + syntactic relation feat.	**0.273**	0.281	**0.509**	0.518	**0.240**	**0.249**
all pol. feat. + all syntactic feat.	0.272	**0.284**	0.502	**0.526**	0.231	0.242

5.4 Impact of Distance Feature

Table 6 displays in detail what impact the addition of the distance feature has on the previously presented feature sets. On almost every feature set, there is an increase in performance when this feature is added. However, the degree of improvement varies. It is smallest on those feature sets which include the syntactic relation features. We, therefore, believe that these two feature types encode very much the same thing. Many of the syntactic relation features implicitly demand the topic word and polar expression to be close to each other. Therefore, when a syntactic relation feature fires, so does the distance feature. Unfortunately, our attempts to combine the syntactic relation features with the distance feature in a more effective way by applying feature selection remained unsuccessful. Moreover, we assume that the parsing accuracy upon which the syntactic features rely

is considerably degraded by the less structured sentences from the blog corpus. Table 6 even suggests that syntactic features are not actually required for this classification task since the best performing feature set only comprises all polarity features and the distance feature. The improvement gained by this feature set when compared to the basic polarity feature is larger than the sum of improvements gained when the two feature subsets are evaluated separately[14]. We assume that in the feature spaces representing the two separate feature sets the decision boundary is highly non-linear. The *combination* of the two sets provides the feature space with the best possible class separation, even though there are other feature subsets, such as the basic polarity feature & the syntactic features, which are *individually* more discriminative than the feature set comprising all polarity expressions or the feature set comprising the basic polarity feature & the distance feature.

Accounting for different types of polar expressions is important and, apparently, this is appropriately reflected by our set of different polarity features. Furthermore, polar expressions within the vicinity of a topic term seem to be crucial for a correct classification, as well. Obviously, defining vicinity by a fixed window size is more effective than relying on syntactic constraints.

Despite its lack of syntactic knowledge, the optimal feature set shows a considerable increase in performance when compared with the baseline ranker relying on text classification with an absolute improvement of 8.2% in MAP, 32.9% in MRR, and 14.3% in Prec@10. There is still in improvement by 6.6% in MAP, 17.6% in MRR, and 4.5% in Prec@10 when it is compared against the simplest ranker comprising one polarity feature (without negation modeling).

Table 6. Impact of distance feature. All feature sets – with the exception of *sentence retrieval with text classifiers* – include **negation** modeling.

Features	**MAP**		**MRR**		**Prec@10**	
		+distance		+distance		+distance
sentence retrieval with text classifiers	0.220	–	0.267	–	0.114	–
basic polarity feature	0.245	0.266	0.441	0.491	0.215	0.226
basic pol. feat. + syntactic prominence feat.	0.266	0.276	0.473	0.499	0.216	0.235
basic pol. feat. + syntactic relation feat.	0.269	0.270	0.481	0.498	0.249	0.253
basic pol. feat. + all syntactic feat.	0.278	0.271	0.509	0.521	0.244	0.256
all polarity features	0.257	**0.302**	0.489	**0.596**	0.215	**0.257**
all pol. feat. + syntactic prominence feat.	0.269	0.285	0.474	0.532	0.222	0.256
all pol. feat. + syntactic relation feat.	0.281	0.285	0.518	0.569	**0.249**	0.256
all pol. feat. + all syntactic feat.	**0.284**	0.281	**0.526**	0.555	0.242	0.252

[14] i.e. the improvement from the basic polarity feature to the optimal feature set is greater than the sum of improvements of the feature set comprising the basic polarity feature & the distance feature and the feature set comprising all polarity features

6 Conclusion

In this paper, we have evaluated different methods for topic-related polarity classification at sentence level. We have shown that a polarity classifier based on simple bag-of-words text classification produces fairly poor results. Better performance can be achieved by classifiers based on lexicon look-up. Obviously, the polarity information encoded in polarity lexicons is more domain independent. Optimal performance of this type of classifier can be achieved when a small set of lightweight linguistic polarity features is used in combination with a distance feature. Syntactic features are not necessary for this classification task.

Acknowledgements

The authors would like to thank Sabrina Wilske for interesting discussions and, in particular, Joo-Eun Feit and Saeedeh Momtazi for annotating the TREC Blog06 data. Michael Wiegand was funded by the German research council DFG through the International Research Training Group between Saarland University and University of Edinburgh.

References

1. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? Sentiment Classification Using Machine Learning Techniques. In: Proc. of EMNLP. (2002)
2. Wiegand, M., Klakow, D.: The Role of Knowledge-based Features in Polarity Classification at Sentence Level. In: Proc. of FLAIRS. (2009)
3. Wilson, T., Wiebe, J., Hoffmann, P.: Recognizing Contextual Polarity in Phrase-level Sentiment Analysis. In: Proc. of HLT/EMNLP. (2005)
4. Hu, M., Liu, B.: Mining and Summarizing Customer Reviews. In: Proc. of KDD. (2004)
5. Santos, R.L., He, B., Macdonald, C., Ounis, I.: Integrating Proximity to Subjective Sentences for Blog Opinion Retrieval. In: Proc. of ECIR. (2009)
6. Kessler, J.S., Nicolov, N.: Targeting Sentiment Expressions through Supervised Ranking of Linguistic Configurations. In: Proc. of ICWSM. (2009)
7. Macdonald, C., Ounis, I.: The TREC Blog06 Collection. Technical Report TR-2006-226 (2006)
8. Macdonald, C., Ounis, I., Soboroff, I.: Overview of the TREC-2007 Blog Track. In: Proc. of TREC. (2007)
9. Shen, D., Leidner, J.L., Merkel, A., Klakow, D.: The Alyssa System at TREC 2006: A Statistically-Inspired Question Answering System. In: Proc. of TREC. (2006)
10. Pang, B., Lee, L.: A Sentimental Education: Sentiment Analysis Using Subjectivity Summarization Based on Minimum Cuts. In: Proc. of ACL. (2004)
11. Lin, D.: Dependency-based Evaluation of MINIPAR. In: Proc. of the Workshop on the Evaluation of Parsing Systems. (1998)
12. Joachims, T.: Making Large-Scale SVM Learning Practical. In: Advances in Kernel Methods - Support Vector Learning. (1999)
13. Ravichandran, D., Hovy, E., Och, F.J.: Statistical QA - Classifier vs. Re-ranker: What's the Difference. In: Proc. of the ACL Workshop on Multilingual Summarization and Question Answering. (2003)