Expanding Domain Sentiment Lexicon through Double Propagation

Guang Qiu[*] [1], Bing Liu[**], Jiajun Bu[*] and Chun Chen[*]

[*] Zhejiang Key Laboratory of Service Robot
College of Computer Science
Zhejiang University
{qiuguang, bjj, chenc}@zju.edu.cn

[**] Department of Computer Science
University of Illinois at Chicago
liub@cs.uic.edu

Abstract

In most sentiment analysis applications, the sentiment lexicon plays a key role. However, it is hard, if not impossible, to collect and maintain a universal sentiment lexicon for all application domains because different words may be used in different domains. The main existing technique extracts such sentiment words from a large domain corpus based on different conjunctions and the idea of sentiment coherency in a sentence. In this paper, we propose a novel propagation approach that exploits the relations between sentiment words and topics or product features that the sentiment words modify, and also sentiment words and product features themselves to extract new sentiment words. As the method propagates information through both sentiment words and features, we call it *double propagation*. The extraction rules are designed based on relations described in dependency trees. A new method is also proposed to assign polarities to newly discovered sentiment words in a domain. Experimental results show that our approach is able to extract a large number of new sentiment words. The polarity assignment method is also effective.

1 Introduction

Sentiment analysis is an important problem in opinion mining and has attracted a great deal of attention [e.g., Hatzivassiloglou and McKeown, 1997; Pang *et al*., 2002; Turney, 2002; Wiebe, 2000; Yu and Hatzivassiloglou, 2003; Hu and Liu, 2004; Esuli and Sebastiani, 2005; Breck *et al.*, 2007]. The task is to predict the sentiment polarities (also known as semantic orientations) of opinions by analyzing sentiment words and expressions in sentences and documents.

Sentiment words are words that convey positive or negative sentiment polarities. A comprehensive sentiment lexicon is essential for sentiment analysis. However, as opinion expressions vary significantly among different domains, it is hard to maintain a universal sentiment lexicon to cover all domains. It is also well known that many such words are domain dependent [Turney, 2002].

Our work is closely related to [Kanayama and Nasukawa 2006], which extracts domain specific sentiment words in Japanese text. In their work, they exploit sentiment coherency within sentence and among sentences to extract sentiment candidates and then use a statistical method to determine whether a candidate is correct. However, their idea of selecting candidates restricts the extracted sentiment words only to contexts with known sentiment words (seeds), and the statistical estimation can be unreliable when the occurrences of candidates are infrequent with small corpora. The key difference between our work and theirs is that we exploit the relationships between sentiment words and product features (or topics) in extraction. This important information is not used in their work. We do not need any input features. Here product features (or features) mean product components and attributes [Liu, 2006]. Experimental results show that our approach, even without propagation, outperforms their method by 18% and 11% in precision and recall respectively. With propagation, our method improves even further.

The proposed method identifies domain specific sentiment words from relevant reviews using only some seed sentiment words (we currently focus on product domains). The key idea is that in reviews sentiment words are almost always associated with features. Thus, sentiment words can be recognized by identified features. Since feature extraction itself is also a challenging problem, we extract features using the same seed sentiment words in a similar way (no seed feature is needed from the user). The newly extracted sentiment words and features are utilized to extract new sentiment words and new features which are used again to extract more sentiment words and features. The propagation ends until no more sentiment words or features can be identified. As the process involves propagation through both sentiment words and features, we call the method *double propagation*. To our knowledge, no previous work on sentiment word extraction employed this approach. The extraction of sentiment words and features is performed using rules designed based on different relations between sentiment words and features, and also sentiment words and features themselves. Dependency grammar [Tesnière, 1959] is adopted to describe these rela-

[1] The research reported in this paper was conducted during his visit to the University of Illinois at Chicago's Department of Computer Science. The financial support was provided by China National Key Technology R&D Program (2008BAH26B00).

tions. A new method is also proposed to predict the polarities of new sentiment words. Experimental results show that the propagation is able to find a large number of new sentiment words and the prediction of polarity is also effective.

2 Related Work

Extensive work has been done on sentiment analysis at word, expression [Takamura *et al.*, 2007; Breck *et al.*, 2007], sentence [Yu and Hatzivassiloglou, 2003] and document [Pang *et al.*, 2002; Turney, 2002] levels. Due to the limited space, we only describe work at word level, which can be categorized as corpora-based approaches [Hatzivassiloglou and McKeown, 1997; Wiebe, 2000; Turney and Littman, 2003; Kanayama and Nasukawa, 2006; Kaji and Kitsuregawa, 2007] and dictionary-based approaches [Hu and Liu 2004; Kamps *et al.*, 2004; Esuli and Sebastiani, 2005; Takamura *et al.*, 2005]. Our work falls in the corpora-based category.

Hatzivassiloglou and McKeown [1997] did the first work on tackling the problem of determining the semantic orientation (or polarity) of words. Their method predicts the orientation of adjectives by analyzing pairs of adjectives extracted from a large document set. These pairs of adjectives are conjoined by *and*, *or*, *but*, *either-or*, or *neither-nor*. The underlying intuition is that the conjoining adjectives subject to linguistic constraints on the orientation of the adjectives involved. For example, *and* usually conjoins two adjectives of the same orientation while *but* conjoins two adjectives of opposite orientations. Our work differs from theirs in that they are unable to extract unpaired adjectives while we could extract through features.

Wiebe [2000] focused on the problem of subjectivity tagging and proposed an approach to finding subjective adjectives using the results of a method for clustering words according to their distributional similarity, seeded by a small number of simple adjectives extracted from a manually annotated corpus. The basic idea is that subjective words are similar in distribution as they share pragmatic usages. However, the approach is unable to predict sentiment orientations of the found subjective adjectives.

Turney and Littman [2003] adopt a different methodology which requires little linguistic knowledge. They first define two minimal sets of seed terms for positive and negative categories. Then they compute the point wise mutual information (PMI) of the target term with each seed term as a measure of their semantic association. Positive value means positive orientation and higher absolute value means stronger orientation. Their work requires additional Web access.

In [Kaji and Kitsuregawa, 2007], the authors propose to use language and layout structural clues of Web pages to extract sentiment sentences from Japanese HTML documents. The structural clues are set in advance. Adjectives/Adjective phrases in these sentences are treated as candidate sentiment phrases. The polarities of the candidates are given based on the computation of their chi-square and PMI values. In our work, we consider unstructured text and do not rely on the HTML layout evidence.

The work of Kanayama and Nasukawa [2006] first uses clause level context coherency to find candidate words from sentences that appear successively with sentences containing seed sentiment words. The intuition is that sentences appearing in contexts tend to have the same polarities; therefore if one of them contains sentiment words, the other successive sentences are likely to contain sentiment words too. The idea is an extension of that in [Hatzivassiloglou and McKeown 1997]. Verbs or adjectives in these sentences are extracted as candidates. Then they use a statistical estimation based method to determine whether the candidates are appropriate sentiment words. However, the idea of using context coherency to find candidates limits the recall if the occurrences of seed words in the data are infrequent or an unknown sentiment word has no known sentiment words in its context. Besides, the statistical estimation may be unreliable if the corpus is small. Our work extracts sentiment words with features and is not limited in successive sentences, so our approach is more flexible and has higher recall. Further, the relations used in our work impose intra-sentence constraints on each sentiment word extraction and are able to maintain good precision even in data of small size.

In dictionary-based approaches, Hu and Liu [2004] and Kim and Hovy [2004] found synonymous and antonyms of a set of seed sentiment words in WordNet. Kamps *et al*. [2004] use WordNet to construct a network by connecting pairs of synonymous words. The semantic orientation of a word is decided by its shortest paths to two seed words "*good*" and "*bad*" representing positive and negative orientations. Esuli and Sebastiani [2005] also used text classification to classify orientations. Their method determines the orientation of words based on glosses in an online glossary or dictionary. The classifier is trained on glosses of selected seed words and is then applied to classify gloss of an unknown word to categorize the word as positive or negative. The work of Takamura *et al.* [2005] exploits the gloss information from dictionaries as well. The authors constructed a lexical network by linking two words if one word appears in the gloss of the other word. The weights of links reflect if these two connected words are of the same orientation. The spin model is adopted to determine the orientation of the words. However, the dictionary-based methods are unable to find domain dependent sentiment words because entries in dictionaries are domain independent.

3 Sentiment Word Propagation and Polarity Assignment

The proposed approach first extracts some sentiment words and features using the seed sentiment lexicon. It then utilizes these sentiment words and features to find new sentiment words and features. The newly extracted sentiment words and features are used to extract more sentiment words and features in the same way. The process continues until no additional sentiment words can be added. The polarities of newly found sentiment words are predicted simultaneously. Note that the extractions are performed based on sentences.

3.1 Sentiment Word Extraction

From the above description, we can see that there are four extraction tasks during the propagation: (1) extract sentiment words using sentiment words; (2) extract features using sentiment words; (3) extract sentiment words using features; (4) extract features using features.

In this work, three types of relations are used to perform these extraction tasks: relations between sentiment words and sentiment words (for task 1), sentiment words and features (for tasks 2 and 3) and features and features (for task 4). Considering complex expressions in texts, we propose to describe the relations in a syntactic way rather than a distance-based way as in [Hu and Liu, 2004] (note that Hu and Liu's method only does feature extraction). In this work, we adopt the dependency grammar to describe these relations and employ the dependency parser Minipar[2] to parse sentences. Corresponding rules are designed based on these relations to extract sentiment words and features.

Relations of Sentiment Words and Features

After parsing, words in a sentence are linked to each other by certain relations. In dependency grammar, the relation between two words *A* and *B* can be described as *A*(or *B*) depends on *B*(or *A*). For example, in the simple sentence "*I love iPod*", both "*I*" and "*iPod*" depend on the verb "*love*" with the relations of ***subj*** and ***obj*** respectively. Here ***subj*** means that "*I*" is the subject of "*love*" while **o*bj*** means that "*iPod*" is the object of "*love*".

In most cases, sentences are much longer and more complex, thus the relations between words can be quite complex, e.g., *A* depends on *C* and *C* depends on *B*. An example is "*The newly released iPod is amazing*" in which "*newly*" depends on "*released*" which depends on "*iPod*" and "*iPod*" itself depends on "*is*". In this paper, we define two categories of relations to summarize all types of relations between two words, which are also illustrated in Figure 1. Arrows are used to represent dependencies.

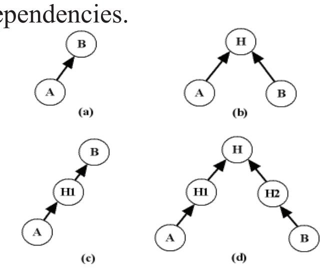

Fig. 1. Different relations between words *A* and *B*. (a) and (b) are two direct relations; (c) and (d) are two indirect relations.

Definition (Direct Relation (DR)): A *direct relation* means that one word depends on the other word directly or they both depend on a third word directly.

Some examples are shown in Figure 1 (a) and (b). In (a), *A* depends on *B* directly while they both directly depend on *H* in (b).

Definition (Indirect Relation (IDR)): An *indirect relation* means that one word depends on the other word through other words or they both depend on a third word indirectly.

Some examples are shown in Figure 1 (c) and (d). In (c), *A* depends on *B* through *H1*; in (d), *A* depends on *H* through *H1* while *B* depends on *H* through *H2*. In more complicated situations, there can be more than one *H1* or *H2*. **DR** can be regarded as a special case with no *H1* or *H2* in the dependency path. Note that in (d), there are cases that no *H1*(or *H2*) between *A*(or *B*) and *H*, but more than one *H2*(or *H1*) between *B*(or *A*) and *H*.

However, complex relations can make the algorithm vulnerable to parsing errors. Parsing is considerable more difficult and error prone with informal expressions used in the Web environment. Thus, in this work, we only utilize **DR**s. **IDR**s are more suitable for formal text such as news articles.

Extraction Rules based on Relations

Given two **DR**s between *A* and *B* (both *A* and *B* can be sentiment words or features), we define rules to capture specific relations as well as the word part-of-speech (POS) information. The Stanford POS tagger[3] is used to do the tagging. As there are four types of extraction tasks in our work, we define four types of rules, which are listed in Table 1.

Table 1. Rules for sentiment word and feature extraction. Column 2 is the observed relations between two words, column 3 shows the constraints on the observed relations and the final column is the result. The arrows mean dependency. For example, $S \rightarrow S\text{-}Dep \rightarrow F$ means *S* depends on *F* through a relation of *S-Dep*.

	Observations	Constraints	Outputs
R1$_1$	$S_{i(j)} \rightarrow S_{i(j)}\text{-}Dep \rightarrow S_{j(i)}$	$S_{j(i)} \in \{S\}$, $S_{i(j)}\text{-}Dep \in \{CONJ\}$, $POS(S_{i(j)}) \in \{JJ\}$	$s = S_{i(j)}$
R1$_2$	$S_i \rightarrow S_i\text{-}Dep \rightarrow H \leftarrow S_j\text{-}Dep \leftarrow S_j$	$S_i \in \{S\}$, $S_i\text{-}Dep == S_j\text{-}Dep$, $POS(S_j) \in \{JJ\}$	$s = S_j$
R2$_1$	$S \rightarrow S\text{-}Dep \rightarrow F$	$F \in \{F\}$, $S\text{-}Dep \in \{MR\}$, $POS(S) \in \{JJ\}$	$s = S$
R2$_2$	$S \rightarrow S\text{-}Dep \rightarrow H \leftarrow F\text{-}Dep \leftarrow F$	$F \in \{F\}$, $S/F\text{-}Dep \in \{MR\}$, $POS(S) \in \{JJ\}$	$s = S$
R3$_1$	$S \rightarrow S\text{-}Dep \rightarrow F$	$S \in \{S\}$, $S\text{-}Dep \in \{MR\}$, $POS(F) \in \{NN\}$	$f = F$
R3$_2$	$S \rightarrow S\text{-}Dep \rightarrow H \leftarrow F\text{-}Dep \leftarrow F$	$S \in \{S\}$, $S/F\text{-}Dep \in \{MR\}$, $POS(F) \in \{NN\}$	$f = F$
R4$_1$	$F_{i(j)} \rightarrow F_{i(j)}\text{-}Dep \rightarrow F_{j(i)}$	$F_{j(i)} \in \{F\}$, $F_{i(j)}\text{-}Dep \in \{CONJ\}$, $POS(F_{i(j)}) \in \{NN\}$	$f = F_{i(j)}$
R4$_2$	$F_i \rightarrow F_i\text{-}Dep \rightarrow H \leftarrow F_j\text{-}Dep \leftarrow F_j$	$F_i \in \{F\}$, $F_i\text{-}Dep == F_j\text{-}Dep$, $POS(F_j) \in \{NN\}$	$f = F_j$

In the table, *s*(or *f*) means the extracted sentiment word (or feature). {*S*}(or {*F*}) and *S*(or *F*)-*Dep* stand for the known sentiment words (or extracted features) and dependency relation of *S*(or *F*) respectively. *H* means any word. ***POS***(*S*(or *F*)) is the POS information of *S*(or *F*). {*JJ*} and {*NN*} are sets of POS tags of potential sentiment words and features respectively. In this work, we consider sentiment words to be adjectives as in most previous work on sentiment analysis, and features to be nouns/noun phrases. Therefore, {*JJ*} contains *JJ, JJR* (adjectives with the comparative ending) and *JJS* (adjectives with the superlative ending). {*NN*} con-

[2] http://www.cs.ualberta.ca/~lindek/minipar.htm

[3] http://nlp.stanford.edu/software/tagger.shtml

sists of *NN* and *NNS*, which stand for singular and plural nouns. However, there are cases that reviewers use pronouns to refer to some features already mentioned previously. Therefore, we also consider the pronouns as features. In the current work, we only use "*it*" and "*they*". Due to possible errors, we have not done any coreference resolution in this work. {*MR*} consists of dependency relations describing relations between sentiment words and features, such as ***mod*** which means that one word modifies the other word. Other *MR*s include ***subj***, ***obj***, ***pnmod***, etc. {*CONJ*} is the relation of conjunction and contains only ***conj***.

We use $\mathbf{R1_i}$ to extract sentiment words (s) using sentiment words (S_i), $\mathbf{R2_i}$ to extract sentiment words (s) using features (F), $\mathbf{R3_i}$ to extract features (f) using sentiment words (S) and $\mathbf{R4_i}$ to extract features (f) using extracted features (F_i).

3.2 Polarity Assignment

We now present our method for *polarity assignment* based on contextual evidences. The method consists of three rules that are integrated into the propagation algorithm. The basic intuition is that people often express their opinions in a consistent manner unless there are explicit *contrary words* such as "*but*" and "*however*". Before we describe our method, let us make some observations about sentiment words and features:

Observation 1 (**same polarity for same feature in a review**): A review is a document written by a single reviewer and is composed of a sequence of sentences. It is usually the case that the reviewer has the same sentiment or polarity on the same feature, although the feature may appear more than once in the review.

Observation 2 (**same polarity for same sentiment word in a domain corpus**). In our case, a domain corpus has a set of reviews reviewing the same product. It is usually the case that the same sentiment word has the same polarity.

Based on these observations, the propagation algorithm assigns polarities to both newly extracted features and sentiment words. The polarity of a feature is the identified sentiment polarity on the feature given in the review. The following rules are exploited to infer polarities for extracted sentiment words and features:

1. **Heterogeneous rule**: For sentiment words extracted by known features, and features extracted by known sentiment words, we assign them the same polarities as the known ones. Note that features convey no polarities and sentiment words are the only expressions that people use to show their attitudes towards features. Therefore, the polarities of features inherit those of associated sentiment words. We also consider whether there are negations such as "*not*" associated with the sentiment words (by examining every word in the surrounding 5 word windows), which change their polarities, i.e., opposite polarities are assigned to sentiment words (or features).

2. **Homogeneous rule**: For sentiment words extracted by known sentiment words and features extracted by known features, we assign them the same polarities as the known ones unless there are contrary words between them. The contrary words include "*but*", "*however*", "*although*", "*except*", etc. We also observe that these words can cancel the polarity change when they are used together or associated with negations. Therefore, we consider that the polarity only changes when there is an odd number of such contrary words and negations between the two sentiment words or features.

3. **Intra-review rule**: There are new sentiment words that are extracted by some features which are initially extracted in other reviews. These features should convey no polarities in current reviews because they do not conform to **Observation 1**. Hence, no polarities will be assigned to the sentiment words. **Observation 2** cannot be applied either if these sentiment words are found only in the current reviews. To assign polarities for these sentiment words, we make use of the overall review polarity to infer. We assume that the sentiment word takes the polarity of the review. The review polarity value is computed as the sum of polarity values of the contained known sentiment words (+1 for positive polarity and -1 for negative polarity). If the final sum is larger than 0, the review is positive and negative otherwise.

Note that, due to both observations, multiple polarities may be assigned to a sentiment word or feature. To resolve conflict, we sum the polarity values. A positive polarity is +1 and a negative polarity is -1. If the sum is larger than 0, the final polarity is positive, otherwise negative. This strategy reduces the probability of incorrect assignment.

4 Experiments and Discussions

We now present the experimental results. We use the customer review collection[4] as the testing data. The collection contains five review data sets: 2 on two digital cameras, 1 on a DVD player, 1 on an mp3 player and 1 on a cell phone. On average, each data set has 789 sentences and 63 reviews. To obtain the sentiment set of each data set for verification, we first use two initial positive and negative sentiment lists (which contain 654 and 1098 words respectively) to find the sentiment words. As there may be missing sentiment words, we exploit the pooling technique which is used in TREC conferences[5] to add missing ones. We manually check the newly extracted sentiment words by each approach (using the initial lists as seed words) and select the appropriate sentiment words to add to corresponding sentiment set of each data set. This strategy reduces efforts of human labeling.

For comparison, we implemented the approach in [Kanayama and Nasukawa, 2006] and only consider the adjectives as the candidates in our experiments (referred to as **KN06** hereafter) since our method only concerns adjective sentiment words. As propagation is not performed in **KN06**, we also implemented a non-propagation version of our approach, in which sentiment words are only extracted by seed words and features extracted by them. Furthermore, we experiment with the conditional random fields (**CRF**) technique for extraction [Lafferty *et al.*, 2001]. The well known toolkit CRF++[6] is employed (using the default parameter settings). We use 7 labels in the training phase, *product*

[4] http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html
[5] http://trec.nist.gov
[6] http://crfpp.sourceforge.net/

features (including "*it*" and "*they*"), *non-feature nouns*, *sentiment adjectives*, *non-sentiment adjectives*, *verbs*, *prepositions/conjunctions* and *others*. Note that **CRF** does not do polarity assignments.

Since **CRF** is a supervised learning method and our approaches and **KN06** are unsupervised, to achieve the same experiment setups, we use the sentiment words contained in the **CRF** training data as the seeds for **KN06** and our approaches. The test data is the same for all approaches. For each run, we use one data set for training **CRF** and the rest four for testing. The average results are reported below. All metrics (precision, recall and F-score) are computed on the newly extracted sentiment words. This is an important point because only the new extractions are meaningful. Using all the extracted words to compute precision and recall is not appropriate as they can include many words that are already in the seed list or the labeled training set in the case of **CRF**.

To examine the accuracy of each approach in extracting new sentiment words with different numbers of seeds, we divide the initial sentiment lexicon (totally 1752 positive and negative words together) into 10 parts, each with roughly the same number of words. We call these lists of sentiment words as *10p* lists. These 10 *10p* lists are combined to produce *20p*, *50p* and *80p* lists which mean containing 20%, 50% and 80% of the original set (1752) respectively. The actual seed list for each experiment is the intersection of the x% and those words appearing in the **CRF** training data file. For **CRF**, those sentences in the training data file that do not contain any sentiment words in this intersection are removed so as not to confuse **CRF**. The experiments using the four kinds of seed lists are performed separately.

4.1 Results of Sentiment Word Extraction

Figures 2, 3 and 4 show the average results of precision, recall and F-score of different approaches using different numbers of seed sentiment words. **Prop-dep** is our propagation approach and **noProp-dep** is the non-propagation version of our technique.

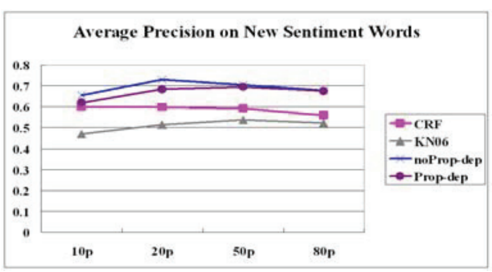

Fig. 2. Precisions of **CRF**, **KN06, noProp-dep** and **Prop-dep**

Observing from Figure 2, we can see that our approaches, both propagation and non-propagation versions, outperform others in all the four cases. It indicates that our designed rules are effective in extracting correct sentiment words. The precision of **CRF** is relatively low, which means **CRF** has difficulty in distinguishing ordinary adjectives from sentiment ones. **KN06** is reported to have around 60% precision in the Japanese test data, but it does not perform as well in our experiments. One reason could be that the statistical estimation of **KN06** measures word positive or negative occurrences compared to its total occurrences, which can introduce unreliability if words are infrequent when the corpus is small. Considering that the size of the testing data in our experiments is much smaller than theirs, the estimation thus can be unreliable. Many infrequent non-sentiment words are identified as sentiment words, which lowers the precision. In our technique, rules are applied in terms of single sentences, thus it is not sensitive to the size of the test data. We also notice that **noProp-dep** achieves better results than **Prop-dep**, which means that the propagation introduces some noises (but the recall is much higher as we will see below). Another observation is that in our approaches, the best performance is gained at *20p* rather than *80p*. This is because at *80p* most of the sentiment words are already known (in the seed list) and the number of remaining ones to be extracted is small and they are usually harder to identify.

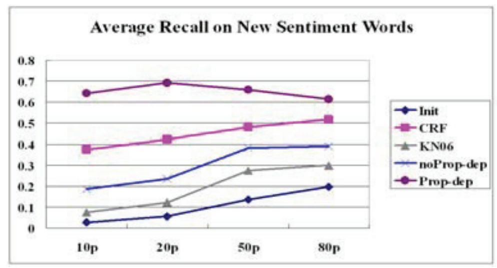

Fig. 3. Recalls of **Init**, **CRF**, **KN06, noProp-dep** and **Prop-dep**

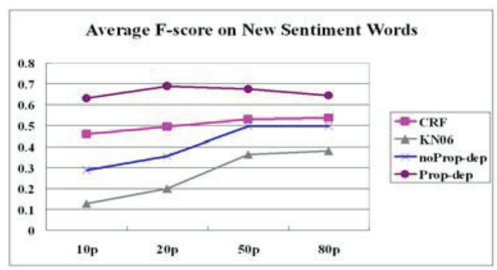

Fig. 4. F-scores of **CRF**, **KN06, noProp-dep** and **Prop-dep**

From Figure 3, we can see that our approach makes significant improvement over others in recall. Clearly, propagation is at work. In the best case (*20p*), the new sentiment words extracted by our approach could cover almost 70% of the whole sentiment set while the corresponding seed words only cover 5% (see the **Init** line). Thus our propagation method is quite powerful in generating a large number of new sentiment words. **CRF** is found to cover about 50% of the sentiment words in its best case (*80p*). Technically **CRF** captures only local patterns rather than long range patterns. Many dependency relationships are long range (i.e., there are many words between the sentiment word and the feature that it modifies), which explains the weaker performance of **CRF**. **KN06** performs poorly in finding new words, which we believe is due to its strategy in selecting candidates. The strategy only considers adjectives in successive sentences and does not use features or any dependency relationships. Such relationships clearly exist and are useful. We also notice the drop in recall for **Prop-dep** at *80p*, which can be explained in the same way as the drop in precision when a large number of seed sentiment words are used.

Figure 4 shows the F-score results. In all the four cases, our propagation approach (**Prop-dep**) achieves the highest F-score. We can thus draw the conclusion that our approach is superior to the existing methods.

4.2 Results of Polarity Assignment

Figure 5 shows the accuracy of polarity assignment of different approaches computed based on the newly discovered correct sentiment words by each approach.

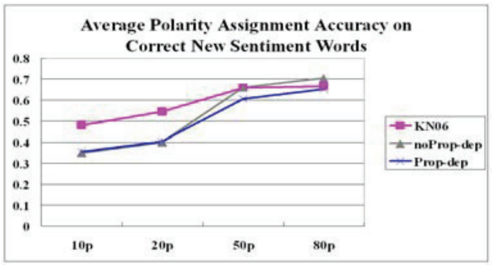

Fig. 5. Average polarity assignment accuracy on correct new sentiment words

From the results, we can see that **Prop-dep** performs worse than **KN06** but getting closer when the number of seeds increases and **noProp-dep** outperforms **KN06** from the case of *50p*. Considering our approach has a much higher recall, more than 30% higher at *80p* (Figure 3), this result is remarkable. At *10p*, *20p* and *50p*, the recall values of our methods are even higher than **KN06**. This means that our methods can extract dramatically more sentiment words, while also maintaining a good accuracy in polarity assignment, especially in the cases of *50p* and *80p*. We consider those two cases to be quite realistic for practical applications because there are already several existing sentiment lexicons compiled by researchers. Thus, in practice, there is no need to start with a very small number of seeds.

5 Conclusions

In this paper, we propose a domain sentiment word extraction approach based on the propagation of both known sentiment lexicon and extracted product features, which we call *double propagation*. The algorithm exploits dependency relations to capture the association between features and sentiment words and also sentiment words and features themselves. Several empirical rules are designed to extract sentiment words and features given known sentiment words. We also propose a new method to assign polarities to extracted sentiment words. Experimental results show that our approaches are effective.

References

[Breck *et al.*, 2007] Eric Breck, Yejin Choi and Claire Cardie. *Identifying expressions of opinion in context*. In Proc. of IJCAI'07, 2007.

[Esuli and Sebastiani, 2005] Andrea Esuli and Fabrizio Sebastiani. *Determining the Semantic Orientation of Terms through Gloss Classification*. In Proc. of CIKM'05, pp. 617-624, 2005.

[Hatzivassiloglou and McKeown, 1997] Vasileios Hatzivassiloglou. and Kathleen R. McKeown. *Predicting the semantic orientation of adjectives.* In Proc. of ACL'97, pp. 174-181, 1997.

[Hu and Liu, 2004] Minqing Hu and Bing Liu. *Mining and Summarizing Customer Reviews.* In Proc. of KDD'04, 2004.

[Kaji and Kitsuregawa, 2007] Nobuhiro Kaji and Masaru Kitsuregawa. *Building lexicon for Sentiment Analysis from Massive Collection of HTML Documents*. In Proc. of EMNLP'07, pp. 1075-1083, 2007.

[Kamps *et al.*, 2004] Jaap Kamps, Maarten Marx, Robert J. Mokken and Maarten de Rijke. *Using WordNet to measure semantic orientation of adjectives.* In Proc. of LREC'04, pp. 1115-1118, 2004.

[Kanayama and Nasukawa, 2006] Hiroshi Kanayama and Tetsuya Nasukawa. *Fully Automatic Lexicon Expansion for Domain-oriented Sentiment Analysis.* In Proc. of EMNLP'06, pp. 355-363, 2006.

[Kim and Hovy, 2004] Soo-Min Kim and Eduard Hovy. *Determining the Sentiment of Opinions.* In Proc. of COLING'04, pp.1367-1373, 2004.

[Lafferty *et al*., 2001] John Lafferty, Andrew McCallum and Fernando Pereira. *Conditional random fields: Probabilistic models for segmenting and labeling sequence data.* In Proc. of ICML'01, Williams, MA, 2001.

[Liu, 2006] Bing Liu. Web Data Mining. Springer, 2006.

[Pang *et al*., 2002] Bo Pang, Lillian Lee and Shivakumar Vaithyanathan. *Thumbs up? Sentiment classification using machine learning techniques.* In Proc. of EMNLP'02, pp. 79-86, 2002.

[Takamura *et al.*, 2005] Hiroya Takamura, Takashi Inui and Manabu Okumura. *Extracting semantic orientations of words using spin model*. In Proc. of ACL'05, pp. 133-140, 2005.

[Takamura *et al*., 2007] Hiroya Takamura, Takashi Inui and Manabu Okumura. *Extracting Semantic Orientations of Phrases from Dictionary.* In Proc. of NAACL HLT'07, pp. 292-299, 2007.

[Tesnière, 1959] Lucien Tesnière, éléments de syntaxe structurale, Klincksieck, Paris 1959.

[Turney, 2002] Peter D. Turney. *Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews.* In Proc. of ACL'02, pp. 417-424, 2002.

[Turney and Littman, 2003] Peter D. Turney and Michael L. Littman. *Measuring praise and criticism: Inference of semantic orientation from association.* ACM Transactions on Information Systems, 21(4):315-346, 2003.

[Wiebe, 2000] Janyce M. Wiebe. *Learning Subjective Adjective from Corpora.* In Proc. of AAAI'00, pp. 735-740, 2000.

[Yu and Hatzivassiloglou, 2003] Hong Yu and Vasileios Hatzivassiloglou. *Towards Answering Opinion Questions: Separating Facts from Opinions and Identifying the Polarity of Opinion Sentences.* In Proc. of EMNLP'03, pp. 129-136, 2003.