Movie Review Mining and Summarization [*]

Li Zhuang
Microsoft Research Asia
Department of Computer
Science and Technology,
Tsinghua University
Beijing, P.R.China
f-lzhuang@hotmail.com

Feng Jing
Microsoft Research Asia
Beijing, P.R.China
fengjing@microsoft.com

Xiao-Yan Zhu
Department of Computer
Science and Technology,
Tsinghua University
Beijing, P.R.China
zxy–dcs@tsinghua.edu.cn

ABSTRACT

With the flourish of the Web, online review is becoming a more and more useful and important information resource for people. As a result, automatic review mining and summarization has become a hot research topic recently. Different from traditional text summarization, review mining and summarization aims at extracting the features on which the reviewers express their opinions and determining whether the opinions are positive or negative. In this paper, we focus on a specific domain – movie review. A multi-knowledge based approach is proposed, which integrates WordNet, statistical analysis and movie knowledge. The experimental results show the effectiveness of the proposed approach in movie review mining and summarization.

Categories and Subject Descriptors

I.2.7 [**Artificial Intelligence**]: Natural Language Processing—*text analysis*; H.2.8 [**Database Management**]: Database Application—*data mining*

General Terms

Algorithms, Experimentation

Keywords

review mining, summarization

1. INTRODUCTION

With the emerging and developing of Web2.0 that emphasizes the participation of users, more and more Websites, such as Amazon (http://www.amazon.com) and IMDB (http://www.imdb.com), encourage people post reviews for the information they are interested in. These reviews are useful for both information promulgators and readers. For example, from the online reviews of political news or announcements, the government can perceive the influence of recent policies or events on common people, and take proper and timely actions based on the information. Through product reviews, on the one hand, manufacturers can gather feedbacks from their customers to further improve their products. On the other hand, people could objectively evaluate a product by viewing other people's opinions, which will possibly influence their decisions on whether to buy the product. However, many reviews are lengthy with only few sentences expressing the author's opinions. Therefore, it is hard for people to find or collect useful information they want. Moreover, for each information unit to be reviewed, such as a product, there may be many reviews. If only few reviews are read, the opinion will be biased. As a result, automatic review mining and summarization has become a hot research topic recently.

Most of the existing work on review mining and summarization is focused on product reviews. In this paper, we will focus on another domain – movie review. Different from product reviews, movie reviews have the following unique characteristic. When a person writes a movie review, he probably comments not only movie elements (e.g. screenplay, vision effects, music), but also movie-related people (e.g. director, screenwriter, actor). While in product reviews, few people will care the issues like who has designed or manufactured a product. Therefore, the commented features in movie review are much richer than those in product review. As a result, movie review mining is more challenging than product review mining.

In this paper, we decompose the problem of review mining and summarization into the following subtasks: 1) identifying feature words and opinion words in a sentence; 2) determining the class of feature word and the polarity of opinion word; 3) for each feature word, fist identifying the relevant opinion word(s), and then obtaining some valid feature-opinion pairs; 4) producing a summary using the discovered information. We propose a multi-knowledge based approach to perform these tasks. First, WordNet, movie casts and labeled training data were used to generate a keyword list for finding features and opinions. Then grammatical rules between feature words and opinion words were applied to identify the valid feature-opinion pairs. Finally, we reorganized the sentences according to the extracted feature-

[*] This work was done while the first author was visiting Microsoft Research Asia. Li Zhuang and Xiao-Yan Zhu are also with State Key Laboratory of Intelligent Technology and Systems.

CIKM'06, November 5–11, 2006, Arlington, Virginia, USA.
Copyright 2006 ACM 1-59593-433-2/06/0011 ...$5.00.

opinion pairs to generate the summary. Experimental results on the IMDB data set show the superiority of the proposed method over a well-known review mining algorithm [6].

The remainder of this paper is organized as follows. Section 2 describes some related work. Section 3 states the problem. Section 4 introduces the proposed approach. In Section 5, experimental results are provided and some typical errors are analysis. Finally, the conclusion and future work are presented in Section 6.

2. RELATED WORKS

Since review mining is a sub-topic of text sentiment analysis, it is related with work of subjective classification and sentiment classification. In the following of this section, we will first introduce existing work on review mining and summarization. Then, we will present work on subjective classification and sentiment classification and discuss their relationship with review mining.

2.1 Review mining and summarization

Different from traditional text summarization, review summarization aims at producing a sentiment summary, which consists of sentences from a document that capture the author's opinion. The summary may be either a single paragraph as in [1] or a structured sentence list as in [6]. The former is produced by selecting some sentences or a whole paragraph in which the author expresses his or her opinion(s). The latter is generated by the auto-mined features that the author comments on. Our work is more relevant to the latter method.

Existing works on review mining and summarization mainly focused on product reviews. As the pioneer work, Hu and Liu proposed a method that uses word attributes, including occurrence frequency, part-of-speech and synset in WordNet [6]. First, the product features were extracted. Then, the features were combined with their nearest opinion words, which are from a generated and semantic orientation labeled list containing only adjectives. Finally, a summary was produced by selecting and re-organizing the sentences according to the extracted features. To deal with the reviews in a special format, Liu et al expanded the opinion word list by adding some nouns [8]. Popescu and Etzioni proposed the OPINE system, which uses relaxation labeling for finding the semantic orientation of words [14]. In the Pulse system introduced by Gamon et al [4], a bootstrapping process was used to train a sentiment classifier. The features were extracted by labeling sentence clusters according to their key terms.

2.2 Subjective classification

The task of subjective classification is to distinguish sentences, paragraphs or documents that present opinions and evaluations from sentences that objectively present factual information. The earliest work was reported in [20], in which the author focused on finding high quality adjective features, using a method of word clustering. In 2003, Riloff et al investigated subjective nouns learned from un-annotated data using bootstrapping process [15], and they used the same approach to learn patterns for subjective expressions [16]. Yu and Hatzivassiloglou presented several unsupervised statistical techniques for detecting opinions at the sentence level, and then used the results with a Bayesian classifier to determine whether a document is subjective or not [22]. In 2005, Wiebe and Riloff developed an extraction pattern learner and a probabilistic subjectivity classifier using only un-annotated texts for training [21]. The performance of their approach rivaled that of previous supervised learning approaches.

The difference between subjective classification and review mining is two-folds. On the one hand, subjective classification does not need to determine the semantic orientations of those subjective sentences. On the other hand, subjective classification does not need to find features on which opinions have been expressed. While review mining need not only find features, but also determine the semantic orientations of opinions.

2.3 Sentiment classification

The task of sentiment classification is to determine the semantic orientations of words, sentences or documents. Most of the early work on this topic used words as the processing unit. In 1997, Hatzivassiloglou and McKeown investigated the semantic orientations of adjectives [5] by utilizing the linguistic constraints on the semantic orientations of adjectives in conjunctions. In 2002, Kamps and Marx proposed a WordNet (http://wordnet.princeton.edu) based approach [7], using semantic distance from a word to "good" and "bad" in WordNet as the classification criterion. Turney used pointwise mutual information (PMI) as the semantic distance between two words [18] so that the sentiment strength of a word can be measured easily. In [19], Turney et al further introduced the cosine distance in latent semantic analysis (LSA) space as the distance measure, which leads to better accuracy.

The earliest work of automatic sentiment classification at document level is [11]. The authors used several machine learning approaches with common text features to classify movie reviews from IMDB. In 2003, Dave et al designed a classifier based on information retrieval techniques for feature extraction and scoring [3]. In 2004, Mullen and Collier integrated PMI values, Osgood semantic factors [10] and some syntactic relations into the features of SVM [9]. Pang and Lee proposed another machine learning method based on subjectivity detection and minimum-cut in graph [12]. In 2005, Pang and Lee further developed their work to determine a reviewer's evaluation with respect to a multi-point scale [13]. In [2], the authors compared two kinds of approaches based on machine learning and semantic orientation systematically.

Sentiment classification is not involved in finding concrete features that are commented on yet. Therefore, its granularity of analysis is different to that of review mining and summarization.

3. PROBLEM STATEMENT

Let $R = r_1, r_2, ..., r_n$ be a set of reviews of a movie. Each review r_i consists of a set of sentences $< s_{i1}, s_{i2}, ..., s_{in} >$. The following describes some related definitions.

Definition (movie feature): A movie feature is a movie element (e.g. screenplay, music) or a movie-related people (e.g. director, actor) that has been commented on.

Since reviewers may use different words or phrases to describe the same movie feature, we manually define some classes for features. The feature classes are pre-defined according to the movie casts of IMDB. The classes are divided into two groups: ELEMENT and PEOPLE. The ELEMENT classes include OA (overall), ST (screenplay), CH (character design), VP (vision effects), MS (music and sound effects) and SE (special effects). The PEOPLE classes include PPR (producer), PDR (director), PSC (screenwriter), PAC (actor and actress), PMS (people in charge of music and sounds, including composer, singer, sound effects maker etc.) and PTC (people in charge of techniques of moviemaking, including cameraman, editor, set designer, special effects maker etc.). Each class contains words and phrases that describe similar movie elements or people in charge of similar kinds of work. For example, "story", "script" and "screenplay" belong to ST class; "actor", "actress" and "supporting cast" belong to PAC class.

Definition (relevant opinion of a feature): The relevant opinion of a feature is a set of words or phrases that expresses a positive (PRO) or negative (CON) opinion on the feature.

The polarity of a same opinion word may vary in different domain. For example, in product reviews, "predictable" is a word with neutral semantic orientation. While in movie reviews, "predictable" plot sounds negative to moviegoers.

Definition (feature-opinion pair): A feature-opinion pair consists of a feature and a relevant opinion. If both the feature and the opinion appear in sentence s, the pair is called an explicit feature-opinion pair in s. If the feature or the opinion does not appear in s, the pair is called an implicit feature-opinion pair in s.

For example, in sentence "The movie is excellent", the feature word is "movie" and the opinion word is "excellent". Therefore, the sentence contains an explicit feature-opinion pair "movie-excellent". While in sentence "When I watched this film, I hoped it ended as soon as possible", the reviewer means the film is very boring. However, no opinion word like "boring" appears in the sentence. We consider this sentence contains an implicit feature-opinion pair "film-boring".

The **task** of movie review mining and summarization is to find the feature-opinion pairs in each sentence first, and then identify the polarity (positive or negative) of the opinions, finally produce a structured sentence list according to the feature-opinion pairs as the summary, of which feature classes are used as the sub-headlines. In the next section, we will introduce our approach to perform the task.

4. MOVIE REVIEW MINING AND SUMMARIZATION

In this paper, we propose a multi-knowledge based movie review mining approach. The overview of the framework is shown in Figure 1. A keyword list is used to record information of features and opinions in movie review domain. Feature-opinion pairs are mined via some grammatical rules and the keyword list. More details of the proposed approach will be introduced in the following.

4.1 Keyword list generation

Considering that feature/opinion words vary obviously with different domains, it is necessary to build a keyword list to capture main feature/opinion words in movie reviews. We divide the keywords into two classes: features and opinions. The feature/opinion phrases with high frequency, such as "special effects", "well acted" etc., are also deemed as keywords.

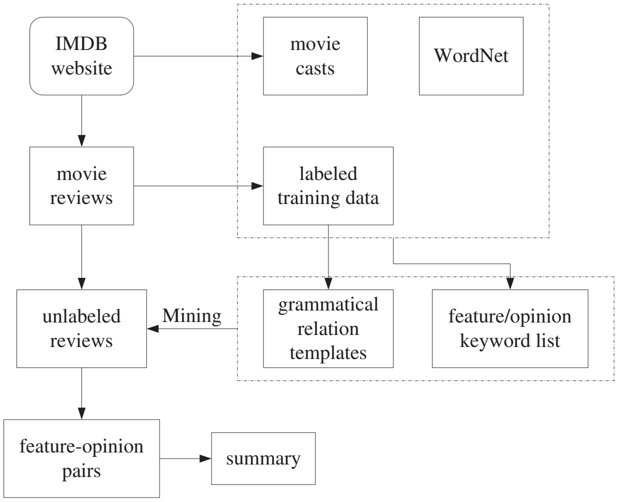

Figure 1: Architectural overview of our multi-knowledge based approach

In the following, we used statistical results on 1,100 manually labeled reviews to illustrate the characteristics of feature words and opinion words. In fact, keyword list generated from the training data was utilized in final experiments. Data we used will be introduced in Section 5.

4.1.1 Feature keywords

In [6], the authors indicated that when customers comment on product features, the words they use converge. Same conclusion could be drawn for movie reviews according to the statistical results on labeled data. For each feature class, if we remove the feature words with frequency lower than 1% of the total frequency of all feature words, the remaining words can still cover more than 90% feature occurrences. In addition, for most feature classes, the number of remaining words is less than 20. Table 1 shows the feature words of movie elements. The results indicate that we can use a few words to capture most features. Therefore, we save these remaining words as the main part of our feature word list. Because the feature words don't usually change, we don't add their synonymic words to expand the keyword list as for opinion words, which will be introduced in the next sub-section.

In movie reviews, some proper nouns, including movie names and people names, can also be features. Moreover, a name may be expressed in different forms, such as first name only, last name only, full name or abbreviation. To make name recognition easier, a cast library is built as a special part of the feature word list by downloading and saving full cast of each movie first and removing people names that are not mentioned in training data. By removing the redundant names, the size of the cast library can be reduced significantly. In addition, because movie fans are usually interested in a few important movie-related people (e.g. director, leading actor/actress, and a few famous composers or cameramen), the strategy will not lose the information of people who are often commented on, but preserve it well.

When mining a new review of a known movie, a few regular expressions are used to check the word sequences beginning with a capital letter. Table 2 shows the regular expres-

Table 1: Feature words of movie elements

Element class	Feature words
OA	film, movie
ST	story, plot, script, storyline, dialogue, screenplay, ending, line, scene, tale
CH	character, characterization, role
VP	scene, fight-scene, action-scene, action-sequence, set, battle-scene, picture, scenery, setting, visual-effects, color, background, image
MS	music, score, song, sound, soundtrack, theme
SE	special-effects, effect, CGI, SFX

sions for people name checking. If a sequence is matched by a regular expression, the cast library will give a person name list according to the same regular expression, so that the matched sequence has same format with each name in the list. If the sequence can be found in the given list, the corresponding name will be the recognition result.

4.1.2 Opinion keywords

The characteristic of opinion words is different to that of feature words. From the statistical results on labeled data, we can find 1093 words expressing positive opinion and 780 words expressing negative opinion. Among these words, only 553 (401) words for positive (negative) are labeled P (N) in GI lexicon [17], which describes semantic orientation of words in general cases. The number of opinion words indicates that people tend to use different words to express their opinions. The comparison with GI lexicon shows that movie review is domain specific. Therefore, for better generalization ability, instead of using all opinion words from statistical results of training data directly, the following steps were performed to generate the final opinion word list.

Firstly, from the opinion words coming from statistical results on training data, the first 100 positive/negative words with highest frequency are selected as seed words and put to the final opinion keyword list. Then, for each substantive in WordNet, we search it in WordNet for the synsets of its first two meanings. If one of the seed words is in the synsets, the substantive is added to the opinion word list, so that the list can deal with some unobserved words in training data. Finally, the opinion words with high frequency in training data but not in the generated list are added as domain specific words.

4.2 Mining explicit feature-opinion pairs

A sentence may contain more than one feature words and opinion words. Therefore, after finding a feature word and an opinion word in a sentence, we need to know whether they compose a valid feature-opinion pair or not. To solve this problem, we use dependency grammar graph to mine some relations between feature words and the corresponding opinion words in training data. The mined relations are then used to identify valid feature-opinion pairs in test data.

Figure 2 shows an example of dependency grammar graph, which is generated by Stanford Parser (http://www-nlp.stanford.edu/software/lex-parser.shtml), without distinguishing governing words and depending words. In training process, first a shortest path from the feature word to the opinion word is detected. Then the part-of-speech (of stemmed word) and relation sequence of the path is recorded. For example, in the sentence "This movie is a masterpiece", where "movie" and "masterpiece" have been labeled as feature and opinion respectively, the path "*movie (NN) - nsubj - is (VBZ) - dobj - masterpiece (NN)*" could be found and recorded as the sequence "*NN - nsubj - VB - dobj - NN*". If there is a negation word, such as "not", the shortest path from the negation word to a word in the feature-opinion path is recorded as the negation sequence, which is showed as the red dashed line in Figure 2. Finally, after removing the low frequency sequences, the remained ones are used as the templates of dependency relation between features and opinions. Table 3 shows four dependency relation templates with highest frequency.

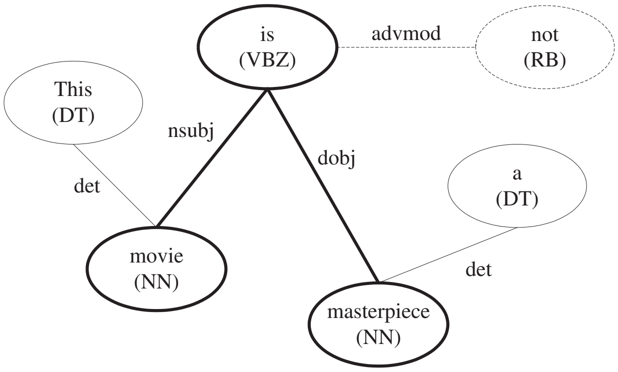

Figure 2: Dependency grammar graph

We use the keyword list and dependency relation templates together to mine explicit feature-opinion pairs. First, in a sentence, the keyword list is used to find all feature/opinion words, which are tagged with all of its possible class labels. Then, the dependency relation templates are used to detect the path between each feature word and each opinion word. For the feature-opinion pair that is matched by a grammatical template, whether there is a negation relation or not is checked. If there is a negation relation, the opinion class is transferred according to the simple rules: *not PRO* → *CON*, *not CON* → *PRO*.

4.3 Mining implicit feature-opinion pairs

Mining implicit feature-opinion pairs is a difficult problem. For example, from the sentence "When I watched this film, I hoped it ended as soon as possible", it is hard to mine the implicit opinion word "boring" automatically. In this paper, we only deal with two simple cases with opinion words appearing.

One case is for very short sentences (sentence length is not more than three) that appear at the beginning or ending of a review and contain obvious opinion words, e.g. "Great!", "A masterpiece." This kind of sentences usually expresses a sum-up opinion for the movie. Therefore, it is proper to

Table 2: Regular expressions for people name checking

No.	Regular expression	Meaning
1	[A-Z][a-z]+ [A-Z][a-z]+ [A-Z][a-z]+	First name + Middle name + Last name
2	[A-Z][a-z]+ [A-Z][a-z]+	First name + Last name
3	[A-Z][a-z]+	First name or Last name only
4	[A-Z][a-z]+ [A-Z][.] [A-Z][a-z]+	Abbreviation for middle name
5	[A-Z][.] [A-Z][.] [A-Z][a-z]+	Abbreviation for first and middle name
6	[A-Z][.] [A-Z][a-z]+	Abbreviation for first name, no middle name

Table 3: Examples of dependency relation templates

Dependency relation template	Feature word	Opinion word
NN - amod - JJ	NN	JJ
NN - nsubj - JJ	NN	JJ
NN - nsubj - VB - dobj - NN	The first NN	The last NN
VB - advmod - RB	VB	RB

Opinion words only for feature class OA:
entertaining, garbage, masterpiece, must-see, worth watching
Opinion words only for movie-related people
clever, masterful, talented, well-acted, well-directed

Figure 3: Some opinion words frequently used for only feature class OA or movie-related people

give an implicit feature word “film” or “movie” with the feature class “OA”. The other case is for a specific mapping from opinion word to feature word. For example, “must-see” is always used to describe a movie; “well-acted” is always used to describe an actor or actress. In order to deal with this case, we record the information of feature-opinion pairs where the opinion word is always used for one movie element or for movie-related people. Therefore, when detecting such an opinion word, the corresponding feature class can be decided, even without a feature word in the sentence. Figure 3 shows some opinion words frequently used for only feature class OA or movie-related people as examples.

4.4 Summary generation

After identifying all valid feature-opinion pairs, we generate the final summary according to the following steps. First, all the sentences that express opinions on a feature class are collected. Then, the semantic orientation of the relevant opinion in each sentence is identified. Finally, the organized sentence list is shown as the summary. The following is an example of the feature class OA.
Feature class: OA
PRO: 70
Sentence 1: The movie is excellent.
Sentence 2: This is the best film I have ever seen.
. . .
CON: 10
Sentence 1: I think the film is very boring.
Sentence 2: There is nothing good with the movie.
. . .

In fact, if movie-related people names are used as the subheadlines, the summary could be generated easily with the same steps. The following is such an example. For movie fans, this kind of summary probably interests them more.
Actress: Vivien Leigh
PRO: 18
Sentence 1: Vivien Leigh is the great lead.
Sentence 2: Vivien's performance is very good.
. . .
CON: 1
Sentence 1: Vivien Leigh is not perfect as many people considered.

5. EXPERIMENTS

As aforementioned in Section 2, Popescu's method outperforms Hu and Liu's method. However, Popescu's system OPINE is not easily available, which brings difficulty with adapting Popescu's method. Therefore, we adapted Hu and Liu's approach [6] and use it as the baseline. More specifically, on the one hand, the proposed keyword list was used to detect opinion words and determine their polarities. On the other hand, the proposed implicit feature-opinion mining strategy was utilized. Precision, recall and F-score are used as the performance measures and defined as

$$precision = \frac{N(correctly\ mined\ feature - opinion\ pairs)}{N(all\ mined\ feature - opinion\ pairs)} \tag{1}$$

$$recall = \frac{N(correctly\ mined\ feature - opinion\ pairs)}{N(all\ correct\ feature - opinion\ pairs)} \tag{2}$$

$$F - score = \frac{2 \times precision \times recall}{precision + recall} \tag{3}$$

where $N(*)$ denotes the number of $*$.

5.1 Data

We used the customer reviews of a few movies from IMDB as the data set. In order to avoid bias, the movies are selected according to two criteria. Firstly, the selected movies can cover as many different genres as possible. Secondly, the selected movies should be familiar to most movie fans. According to the above criterions, we selected 11 movies from the top 250 list of IMDB. The selected movies are *Gone with the Wind*, *The Wizard of OZ*, *Casablanca*, *The Godfather*, *The Shawshank Redemption*, *The Matrix*, *The Two Towers*

(The Lord of the Rings II), *American Beauty*, *Gladiator*, *Wo hu cang long*, and *Spirited Away*. For each movie, the first 100 reviews are downloaded. Since the reviews are sorted by the number of people who think them helpful, the top reviews are more informative. There are totally more than 16,000 sentences and more than 260,000 words in all the selected reviews.

Four movie fans were asked to label feature-opinion pairs, and give the classes of feature word and opinion word respectively. If a feature-opinion pair is given the same class label by at least three people, it is saved as the ground-truth result. The statistical results show that the consistency of at least three people is achieved in more than 80% sentences.

5.2 Experimental results

We randomly divided the data set into five equal-sized folds. Each fold contains 20 reviews of each movie. We used four folds (totally 880 reviews) as the training data and one fold as the test data, and performed five-fold cross-validation. Table 4 shows the average five-fold cross-validation results on the data.

From Table 4, three conclusions could be drawn. First, the precision of our approach is much higher than that of Hu and Liu's approach. One main reason is that, in Hu and Liu's approach, for each feature word, its nearest opinion word is used to construct the feature-opinion pair, which produces many invalid pairs due to the complexity of sentences in movie reviews. While our approach uses dependency relations to check the validity of a feature-opinion pair, which effectively improves the precision. Second, the average recall of our approach is lower than that of Hu and Liu's approach, which is due to two reasons: 1) Hu and Liu's approach identifies infrequent features, while our approach only depends on the keyword list that does not contain infrequent features; 2) Feature-opinion pairs with infrequent dependency relations cannot be detected by our approach because the infrequent relations are removed, while Hu and Liu's approach is not restricted by grammatical relations. The Last conclusion is that the average F-score of 11 movies of our approach is higher than that of Hu and Liu's approach by relative 8.40%.

Table 5 shows the average results of 11 movies for two feature classes - OA and PAC, as an example for detailed results. From it, same conclusions about precision and recall could be drawn.

Comparing with the product review mining results reported in [6] and [14], it can be found that both precision and recall of movie review mining are much lower than those of product review mining. This is not surprising, since movie reviews are known to be more difficult with sentiment mining. Movie reviews often contain many sentences with objective information about the plot, characters, directors or actors of the movie. Although these sentences are not used to express the author's opinions, they may contain many positive and negative terms. Therefore, there may be many confusing feature-opinion pairs in these sentences, which result in the low precision. In addition, movie reviews contain more literary descriptions than product reviews, which brings more implicit comments and results in the low recall.

5.3 Discussion

For further improvement, we checked the mining results manually and carefully. In the following, we will show a few examples to analyze some typical errors. For clarity, *Italic* and underline are used to denote feature word and opinion word, respectively.

Example 1:
Sentence: This is a good *picture*.
Error result: Feature class: **VP**
Right result: Feature class: **OA**

This error is due to the ambiguity of the word "picture". In most cases, "picture" means visual representation or image painted, drawn or photographed, which belongs to the feature class "VP" in our keyword list. However, in this sentence, it means movie.

Example 2:
Sentence: The *story* is simple.
Error result: Opinion class: **PRO**
Right result: Opinion class: **CON**

This error is due to the ambiguity of the word "simple", which has different semantic orientations in different cases. Sometimes, it means the object is easy to understand, where the semantic orientation is PRO. While sometimes it means the object is too naive, where the semantic orientation should be CON. In our approach, we just looked up the keyword list, and took the first found item as the result, which resulted in the error. However, from only one sentence, it is very difficult to identify the semantic orientation of words such as "simple", "complex" etc. To solve the problem, context information should be used.

Example 3:
Sentence: Is it a good *movie*?
Error result: Feature-Opinion pair: movie-good
Right result: NULL

This sentence is a question without answer. Therefore, we cannot decide the polarity of the opinion about the feature "movie" from only this sentence. However, the proposed algorithm cannot deal with it correctly, because the possible feature-opinion pair "movie-good" can be matched by the most frequently used dependency relation template "*JJ - amod - NN*", and "movie/good" is an obvious feature/opinion keyword. Same as example 2, context information should be used to solve the problem.

Example 4:
Sentence: This is a fantasic *movie*.
Error result: NULL
Right result: Opinion word: **fantastic**

Here the word "fantasic" is the mis-spelling of word "fantastic". In fact, there are many spelling errors in online movie reviews. In the test set, there exist errors such as "attative", "mavelous" and so on. It is easy for the human labelers to recognize and label these words. However, most of these unusual words will not be added to the keyword list. Therefore, this kind of errors will be almost unavoidable unless spelling correction is performed.

6. CONCLUSION AND FUTURE WORK

In this paper, a multi-knowledge based approach is proposed for movie review mining and summarization. The objective is to automatically generate a feature class-based summary for arbitrary online movie reviews. Experimental results show the effectiveness of the proposed approach. In addition, with the proposed approach, it is easy to generate a summary with movie-related people names as the sub-headlines, which probably interests many movie fans.

In the future work, we will further improve and refine our

Table 4: Results of feature-opinion pair mining

Movie	Hu and Liu's approach			The proposed approach		
	Precision	Recall	F-score	Precision	Recall	F-score
Gone with the Wind	0.462	0.651	0.551	0.556	0.564	0.560
The Wizard of OZ	0.475	0.705	0.568	0.589	0.648	0.618
Casablanca	0.431	0.661	0.522	0.452	0.521	0.484
The Godfather	0.400	0.654	0.496	0.476	0.619	0.538
The Shawshank Redemption	0.443	0.620	0.517	0.514	0.644	0.571
The Matrix	0.353	0.565	0.434	0.468	0.593	0.523
The Two Towers	0.338	0.583	0.428	0.404	0.577	0.476
American Beauty	0.375	0.576	0.454	0.393	0.527	0.450
Gladiator	0.405	0.619	0.489	0.505	0.632	0.562
Wo hu cang long	0.368	0.567	0.447	0.465	0.537	0.498
Spirited Away	0.388	0.583	0.466	0.493	0.567	0.527
Average	0.403	0.617	0.488	0.483	0.585	0.529

Table 5: Average results of pair mining for feature class OA and PAC

Feature class	Opinion class	Hu and Liu's approach			The proposed approach		
		Precision	Recall	F-score	Precision	Recall	F-score
OA	PRO	0.387	0.738	0.508	0.427	0.693	0.528
	CON	0.400	0.533	0.457	0.448	0.450	0.449
PAC	PRO	0.457	0.708	0.555	0.595	0.682	0.636
	CON	0.305	0.355	0.328	0.401	0.420	0.410

approach from two aspects as the analysis of errors indicated. Firstly, a spelling correction component will be added in the pre-processing of the reviews. Secondly, more context information will be considered to perform word sense disambiguation of feature word and opinion word. Furthermore, we will consider adding neutral semantic orientation to mine reviews more accurately.

7. ACKNOWLEDGEMENTS

The authors wish to express sincere gratitude to the anonymous referees and Dr. Hang Li for their constructive comments and helpful suggestions. They are also very thankful to Qiang Fu, Hao Hu, Cheng Lv, Qi-Wei Zhuo and Chang-Hu Wang for their efforts on data preparation. The first author and the third author are grateful to the financial support by the Natural Science Foundation of China (Grants No. 60572084 and 60321002).

8. ADDITIONAL AUTHORS

Additional authors: Lei Zhang (Microsoft Research Asia, email: `leizhang@microsoft.com`).

9. REFERENCES

[1] Philip Beineke, Trevor Hastie, Christopher Manning and Shivakumar Vaithyanathan. *An exploration of sentiment summarization*. In Proceedings of AAAI 2003, pp.12-15.

[2] Pimwadee Chaovalit and Lina Zhou. *Movie review mining: A comparison between supervised and unsupervised classification approaches*. In Proceedings of HICSS 2005, vol.4.

[3] Kushal Dave, Steve Lawrence and David M. Pennock. *Mining the peanut gallery: Opinion extraction and semantic classification of product reviews*. In Proceedings of WWW 2005, pp.519-528.

[4] Michael Gamon, Anthony Aue, Simon Corston-Oliver and Eric Ringger. 2005. *Pulse: Mining customer opinions from free text*. In Proceedings of IDA 2005, pp.121-132.

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

[6] Minqing Hu and Bing Liu. *Mining and summarizing customer reviews*. In Proceedings of ACM-KDD 2004, pp.168-177.

[7] J. Kamps and M. Marx. 2002. *Words with attitude*. In Proc. of the First International Conference on Global WordNet, pp.332-341.

[8] Bing Liu, Minqing Hu and Junsheng Cheng. *Opinion Observer: Analyzing and comparing opinions on the web*. In Proceedings of WWW 2005, pp.342-351.

[9] Tony Mullen and Nigel Collier. *Sentiment analysis using support vector machines with diverse information sources*. In Proceedings of EMNLP 2004, pp.412-418.

[10] Charles E. Osgood, George J. Succi and Percy H.Tannenbaum. 1957. *The Measurement of Meaning*. University of Illinois.

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

[12] Bo Pang and Lillian Lee. *A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts*. In Proceedings of ACL 2004, pp.271-278.

[13] Bo Pang and Lillian Lee. *Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales*. In Proceedings of ACL 2005, pp.115-124.

[14] Ana-Maria Popescu and Oren Etzioni. *Extracting product features and opinions from reviews*. In Proceedings of EMNLP 2005, pp.339-346.

[15] Ellen Riloff, Janyce Webie and Theresa Wilson. *Learning subjective nouns using extraction pattern bootstrapping*. In Proceedings of CoNLL 2003, pp.25-32.

[16] Ellen Riloff and Janyce Wiebe. *Learning extraction patterns for subjective expressions*. In Proceedings of EMNLP 2003, pp.105-112.

[17] Philip J. Stone, Dexter C. Dunphy, Marshall S. Smith and Daniel M. Ogilvie. 1966. *The General Inquirer: A Computer Approach to Content Analysis*. MIT Press, Cambridge, MA.

[18] Peter D. Turney. *Thumbs up or thumbs down: Semantic orientation applied to unsupervised classification of reviews*. In Proceedings of ACL 2002, pp.417-424.

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

[20] Janyce Wiebe. *Learning subjective adjectives from corpora*. In Proceedings of AAAI 2000, pp.735-740.

[21] Janyce Wiebe and Ellen Riloff. *Creating subjective and objective sentence classifiers from un-annotated texts*. In Proceedings of CICLing 2005, pp.486-497.

[22] Hong Yu and Vasileios Hatzivassiloglou. *Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences*. In Proceedings of EMNLP 2003, pp.129-136.