An opinion search system for consumer products

Qingliang Miao, Qiudan Li

Abstract—**With the rapid progress of e-commerce, many people like purchasing product on the e-commerce website, and giving their personal reviews to the product they purchased, so the number of customer reviews grows rapidly. Generally, a potential customer will browse product reviews before they purchase the product. However, retrieving opinions relevant to customer's desire still remains challenging. To provide efficient opinion information for customers, we propose an opinion search system for consumer products, which utilizes data mining and information retrieval technology. A ranking mechanism taking temporal dimension into account and a method for results visualization are developed in the system. Experimental results on a real-world data set show the system is feasible and effective.**

I. Introduction

In recent years, the rapid progress of Internet and Web technologies has promoted the development of electronic commerce. Generally, customers who purchase product on the e-commerce website are asked to give their personal review to the products they purchased, the customer reviews is a powerful source of opinion information. However, due to the large number of reviews, it is difficult for a potential customer to get efficient opinion information to make an informed decision on whether to purchase the product or not.

One way to overcome the above problem is to develop an opinion search system to provide efficient opinion information for potential customers' particular interest. With the help of the system, users can search for opinions on a particular object or feature of an object [7], e.g., customer opinions on a digital camera or the picture quality of a digital camera.

Fortunately, some researches on opinion mining and opinion search are conducted in recent years. Liu et al. [1][2][8] and Ana-Maria Popescu et al. [12] proposed some methods on extracting product features and opinions. S. Kim et al. [11], P. Turney [14] and V. Hatzivassiloglou et al. [15] proposed some methods to identify opinion semantic orientation. Jian Liu et al. [5] studied the problem of opinion searching, whose aim is to search the opinions about specific feature of specific product and locate them in multi-product reviews. Osamu Furuse et al. [13] implemented a search engine that can extract opinion sentences relevant to an open-domain query from Japanese blog pages.

Qingliang Miao is with the Key Laboratory of Complex System and Intelligence Science, Institute of Automation, Chinese Academy of Sciences, No.95 Zhongguancun East Road, Beijing, P.R.China (e-mail: qingliang.miao@ia.ac.cn).

Qiudan Li is with the Key Laboratory of Complex System and Intelligence Science, Institute of Automation, Chinese Academy of Sciences, No.95 Zhongguancun East Road, Beijing, P.R.China (e-mail: qiudan.li@ia.ac.cn).

Although several works have been done on opinion mining and opinion search, there are still some problems to be solved. For example, the search results should be ranked before returning to the user. In many cases, the search results are very long [7], it is hard for potential customers to read many of them in order to obtain an overview of the prevailing sentiments. Some forms of summary of opinions or results visualization are desirable. Opinions have a temporal dimension [7]. For example, the opinions of people on a particular object, e.g., a product or a topic, may change over time. Displaying the changing trend of sentiments along the time axis can be very useful in many applications.

This paper focuses on the ranking mechanism and search results visualization. Our contributions are as follows:

(1). We introduce a ranking mechanism, which computes the rank of a review sentence using three measures. More importantly we take temporal dimension information into account during the ranking process.

(2). We monitor customer reviews' changing trends with time, and visualize the changing trends.

(3). We generate visual comparison of positive and negative evaluation of a particular feature which potential customers are interested in.

The remainder of the paper is organized as follows. Section II describes related work, Section III introduces the basic terminology and gives the architecture of the opinion search system, Section IV expatiates the implementation methods, Section V gives an illustrated example, and Section VI reports the experimental process and the results of the study. Finally the conclusion is given in Section VII.

II. Related Work

Our work is closely related to Minqing Hu and Bing Liu's work in [1][2][8] on mining product features and determining the polarity of opinion sentences, their work is preformed in three steps: (1) mining the product features and opinions that have been commented on by customers. First, they used association rule mining to find all frequent item sets, after that they did feature pruning or compactness pruning, they extracted the nearby adjective words as opinion words at last; (2) identifying the opinion sentence in each review and deciding whether each opinion sentence is positive or negative, they used WordNet to predict the semantic orientations of opinion words; (3) summarizing the results; Our review sentences translation module is partly based on their work. In [14] P. Turney proposed a corpus-based approach PMI-IR. PMI-IR uses Pointwise Mutual Information (PMI) and Information Retrieval (IR) to measure the similarity of pairs of words or phrases. The semantic orientation of a given phrase is calculated by comparing its similarity to a positive reference word ("excellent") with its

978-1-4244-1821-3/08/$25.00©2008 IEEE

similarity to a negative reference word ("poor"). In [3][4] Philip S. Yu et al. studied the temporal dimension of search in the context of research publication. They argued that Page Rank and HITS algorithms miss the temporal dimension. They added temporal dimension to the PageRank algorithms, and proposed TimePageRank. In [5] Jian Liu et al. studied the problem of opinion searching. They used PMI to retain the domain-related phrases. For determining on which product a given feature semantically depends, they used machine learning method to build a classifier to predict Feature-Product Dependence. Their work included two steps: opinion indexing and opinion retrieving. Opinion indexing is to mark up all the informative opinions as opinion tuples and opinion retrieval is only to look up the opinion tuples. Our work differs from theirs in three main aspects: (1) Our focus is not on opinion indexing but on ranking the search results, our ranking mechanism can return more appropriate results to meet the user's information need. (2) Our work monitors customer reviews' changing trends with time. (3) Our system generates visual comparison of positive and negative evaluation of a particular feature which potential customers are interested in. In [6] Silvana Aciar et al. created an Informed Recommender system on consumer product reviews, although they focused on ranking but they did not take the temporal dimension into the rank algorithm. Also they did not monitor customer reviews' changing trends with time. In [13] Osamu Furuse's system identified positive, negative and neutral opinions and extracted only explicitly stated writer's opinions at the sentence-level.

III. TERMINOLOGY AND ARCHITECTURE

For clearness and disambiguation, here we introduce the important terms used in this paper. A product P has a set of reviews $R=\{r_1,r_2,...r_m\}$, each review r_i is a sequence of sentences $r_i=\{r_{i1},r_{i2},...r_{in}\}$. We represent each review of a product as a tuple including 4 elements [Title, Help, Date, R-Content], where Title is the title of the review, Help is the number of customers who find the review is helpful, Date is the date when the review is commented, R-Content is a set of sentences in customer reviews. We adopt the method in [5] to represent each sentence of a customer review as a tuple, which includes 3 elements [Feature, Sentiment, S-Content]. Feature includes a property, a part, a feature of product, a related concept, a property or a part of related concepts. Sentiment is the opinion polarity of the feature in the same sentence, S-Content is the content of the sentence. Fig. 1 shows the tuple's general structure. Metadata describes how and when and by whom a particular set of data was collected, and how the data is formatted. The metadata of a review contains Title, Help and Date.

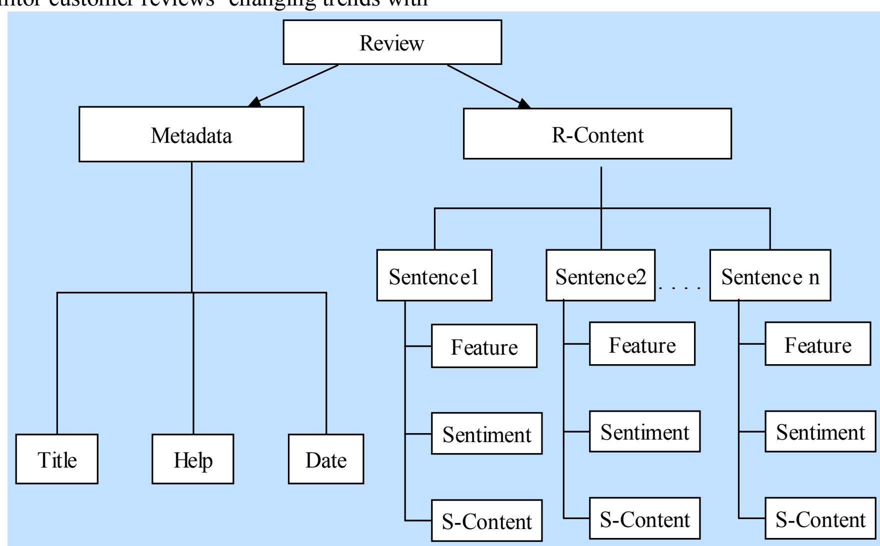

Fig. 1. The structure of the tuple used in the opinion search system

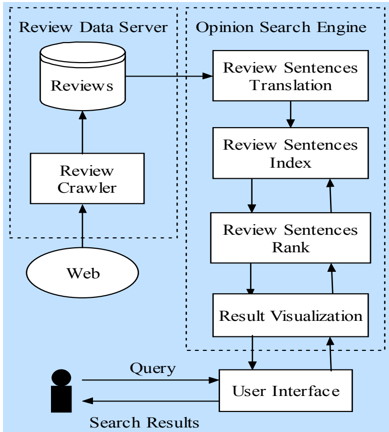

Fig. 2. Architecture of the opinion search system

Fig. 2 shows the architecture of the proposed opinion search system for customer reviews. The system consists of three parts: (1) Review Data Server; (2) Opinion Search Engine; (3) User Interface.

A. Review Data Server

The review data server collects review pages by periodically crawling the e-commerce websites such as Amazon.com. These review documents are then cleaned to remove HTML tags and stored in the Review Data Server.

B. Opinion Search Engine

Our opinion search engine receives review pages from the review data server, it consists of four main modules: review sentences translation, review sentences index, review sentences rank and results visualization. The review sentences translation module checks whether each sentence in the crawled review pages contains product feature and determine the sentiment of customer opinion on the feature. Review sentences are then translated to the tuple.

The review sentences index module indexes the review sentences and builds the index files off-line.

The opinion sentences rank module ranks opinion sentences relevant to the user's query phrases from the index files. This module is processed on-line.

The result visualization module gives the visualization of customer reviews' changing trends and the comparison of positive and negative evaluation of a particular feature which the user is interested in.

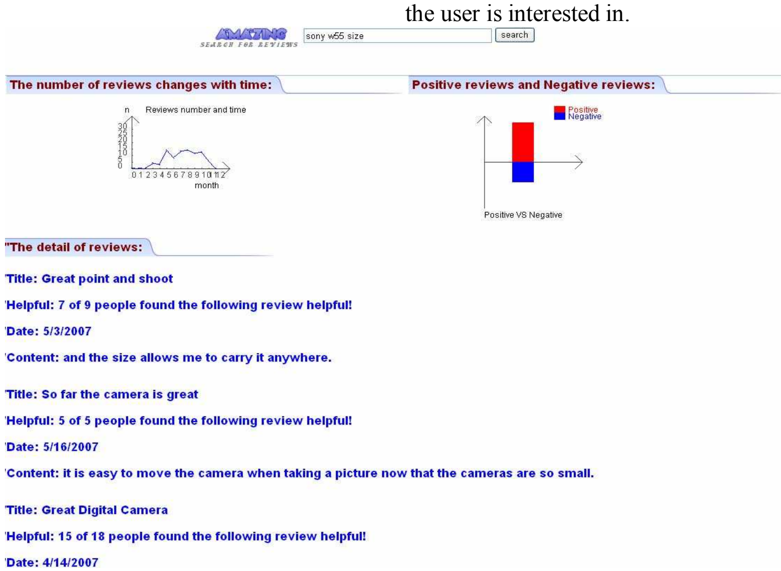

Fig. 3. User interface of the opinion search system

C. User Interface

Fig. 3 shows the user interface of the system. When a user inputs a particular object or features of an object, e.g., "Sony W55 size" in the query box and then clicks the search button, the system will return the results relevant to the query, which consists of the following three parts: the visualization of customer reviews' changing trends is on the top left corner of the page, the comparison of positive and negative evaluation of a particular feature is on the top right corner of the page, the customer review sentences which are relevant to the user's query is on the rest of the page.

IV. Implementation Methods

We implement our opinion search system as follows:

Step 1 : Review sentences translation.

To retrieve opinion sentences that are obvious and a user is glad to read, we only extract explicit features at the sentence level and discard the implicational features. First, the NLProcessor linguistic parser [16] is used to parse each review to sentences, synchronously generate the POS (part-of-speech) tag for each word [2]. Second, for the hypothesis that product features are usually nouns or noun phrases in review sentences [2], a file which only includes the identified nouns and noun phrases of the sentences is built.

Third, the association miner CBA [9], which is based on the Apriori algorithm in [10] is used to extract frequent features, such as size, picture quality, battery, price, et al. We build a feature set which contains all the features we have obtain.

After feature extraction, the sentences which do not contain the features in the feature set are discarded. For the sentences which contain features in the feature set, we extract opinion word about the feature in the sentence and then use the dictionary-based approach [2] for opinion polarity determination.

At last we translate the reviews to the tuple mentioned in section III.

Step 2 : Review sentences index.

In the Review Sentences Index module, Lucene is used to build our index files, which is an open source project of the Apache Software Foundation [17]. The metadata title, help and date are stored in the index files, and the R-Content are indexed in the index files.

Step 3 : Review sentences rank.

The opinion search system ranks the result sentences in respect to a user's query on the basis of some measures, they are: Opinion quality (OQ), Temporal dimension factor (TDF), Lucene rank (LR) and Final rank (FR). OQ gives the opinions' weighting value according to the number of people who find the review is helpful. TDF is a parameter in temporal dimension which is determined by the time interval between reviews' commented date and user's query date. LR is a relevant rank which is based on the vector space model, it evaluates the similarity between opinion sentence and query. FR provides a final rank of the product based on the valuation of each feature. In this section, we describe the calculation of these measures.

Rating the opinion quality (OQ)

Reviewers with different experiences and skill levels made different reviews. In general, if more people find the review is helpful, more important the review is. Therefore, rather than treating all opinions equally, we should give higher weight to the helpful reviews.

The opinion quality (OQ) is calculated as formula (1)

$$OQ_i = \frac{a_i}{\sum_{j=1}^{n} a_j} \tag{1}$$

Where OQ_i is the opinion quality of sentence i, a_i is the number of people who find the review is helpful, n is the total number of reviews of one product.

OQ_i is calculated from the values stored in the corresponding part of the index files. We calculate the OQ_i value for each sentence of a review.

Rating the temporal dimension factor (TDF)

In general people want to read the latest reviews because the latest reviews contain more new information than the old ones. So we take the temporal dimension into account in the ranking mechanism. Inspired by the idea of the paper [4], we choose to decay the temporal weights of each review exponentially, where the temporal dimension mentioned above is considered.

The temporal dimension factor (TDF) is calculated as formula (2)

$$\mathrm{TDF_i} = \exp\frac{(t_i - t)}{30*\beta} \tag{2}$$

Where $\mathrm{TDF_i}$ is the temporal dimension factor of sentence i, t_i is the date when review i was commented, t is the date user do the search and β is a constant. Note that the value of $(t_i - t)$ is negative. Because t and t_i are calculated by day, $(t_i - t)$ is divided by 30.

Lucene rank (LR)

Lucene rank is calculated as formula (3).

$$LR_i = \sum_{t \in q} tf(t) * idf(t) * b(t.field) * lN(t.field) \tag{3}$$

Where $tf(t)$ is term frequency, $idf(t)$ is the inversed document frequency, $b(t.field)$ is the boost factor of each field and $lN(t.field)$ is the standardized value of field.

Final rank (FR)

Based on the above measures, the final rank is calculated as formula (4)

$$FR_i = \alpha LR_i + (1-\alpha)(TDF_i + OQ_i) \tag{4}$$

Where α is a constant between zero and one.

Step 4 : Search results visualization.

In the result visualization module, we give a statistics curve to show customer reviews' changing trends with time at a review level and a histogram to show the comparison of positive and negative evaluation of a particular feature which the user is interested in.

In order to give product manufacturers the changing trends of customer reviews, we compute the number of reviews in the search results of each month and draw a statistical curve to show the changing trends.

For the purpose of giving a potential customer the comparison between the positive and negative evaluation, a histogram is presented by computing the number of sentence with positive and negative opinion separately in the search results. The part above x-axis presents the positive evaluation and the part below x-axis presents the negative evaluation. From the histogram, a potential customer can intuitively see the comparison between the positive and the negative evaluation.

V. AN ILLUSTRATED EXAMPLE

An example is presented to show how the system works, a digital camera is regarded as an object. The data is crawled from Amazon.com. First, we explain how the reviews are translated into the tuple.

A. Review Translation

Fig. 4 shows the review used in the illustrated example. We obtain the product features and sentiment by analyzing the reviews. Fig. 5 shows the analysis results. After the analyzing process, we translate the review to the tuple. Fig. 6 shows the translation of the review to the predefined tuple.

257 of 261 people found the following review helpful:
A very good choice for lots of people - easy to carry, easy to use
March 24, 2007
The Sony W55 is the follow up to the successful and popular W50. It's a very good compact camera, easy to carry and easy to use, and reasonably priced - so most people will be happy with it.
Not much has changed from the W50, which in this case is a good thing!
Same (good) lenses, same size sensor = the same (good) pictures, but not better.
The obvious change is of course the colors. Don't make too much fun of that! There are 4 DSC-W series cameras in our family, colors would have kept my mother-in-law from taking my sister's camera home with her. A little personalization is a good thing.

Fig. 4. A review from Amazon.com that we used in our example.

Helpful: 257 of 261
Title: A very good choice for lots of people - easy to carry, easy to use
Date: March 24, 2007
[Sony; P; The Sony W55 is the follow up to the successful and popular W50.]
[Sony; P; It's a very good compact camera,]
[Use; P; easy to carry and easy to use,]
[Price; P; and reasonably priced - so most people will be happy with it.]
[Lens; P; Same (good) lenses,]
[Picture; P; same size sensor = the same (good) pictures,]

Fig. 5. The result of Review sentences analysis

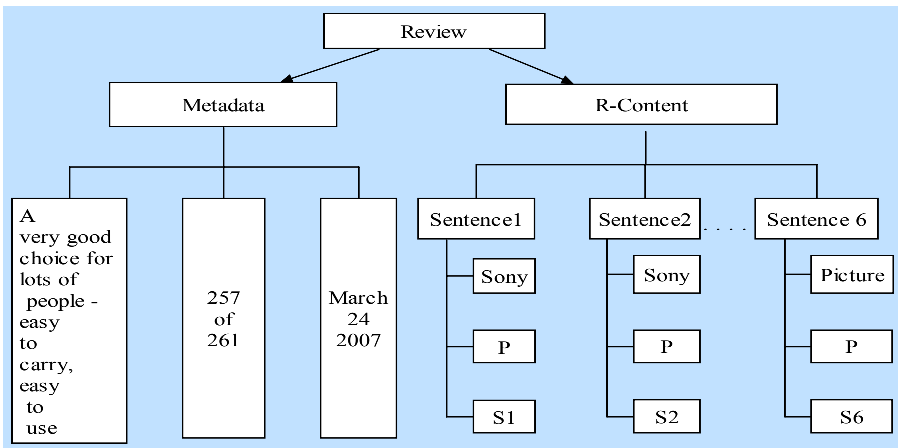

Fig. 6. Translated tuple from a customer review

B. *Review sentences index*

In this step the field Title = "A very good choice for lots of people - easy to carry, easy to use", Help = "257 of 261", Date = "March 24 2007" are stored in the index file, and the S-Content is indexed in the index file.

C. *Review sentence rank*

In this section, we provide the details of the ranking mechanism for a user query. For simplicity, we suppose that one product only has three reviews.

TABLE I
THREE EXAMPLE REVIEWS ON CAMERA SONY W55

	Review1	Review2	Review3
Help	257/ 261	16/17	15 /18
Date	3/24/07	4/10/07	5/15/07

Calculating OQ.

OQ is computed using formula (1). Table I presents the metadata Help in row 2. The OQ values for each consumer review in Table I are

$$OQ_1 = \frac{257}{257+16+15} = 0.89$$

$$OQ_2 = \frac{16}{257+16+15} = 0.06$$

$$OQ_3 = \frac{15}{257+16+15} = 0.05$$

From the Opinion Quality perspective, the most helpful review is Review1, Review1 has been read by most people and most of them believe it is helpful. So Review1 has better quality than Review2 and Review3.

Calculating TDF.

TDF is calculated using formula (2), we assign β =10, t=11/1/2007. Table I presents the metadata Date in row 3. The TDF values for each consumer review in Table I are:

$$TDF_1 = \exp\{-\frac{[30*(11-3)-24]}{30*10}\} = 0.49$$

$$TDF_2 = \exp\{-\frac{[30*(11-4)-10]}{30*10}\} = 0.51$$

$$TDF_3 = \exp\{-\frac{[30*(11-5)-15]}{30*10}\} = 0.58$$

Calculating LR.

LR is calculated using formula (3), LR represents the similarity between opinion sentence and query.

$$LR_i = 0.87$$

$$LR_2 = 0.91$$
$$LR_3 = 0.96$$

Calculating FR.

FR is computed using formula (4), we assign α =0.65. Note that if α is 1, the FR is equal to LR. FR gives the final rank as follows:

$$FR_1 = 0.65 * 0.87 + 0.35 * (0.89+0.49) = 1.05$$
$$FR_2 = 0.65 * 0.91 + 0.35 * (0.06+0.51) = 0.79$$
$$FR_3 = 0.65 * 0.96 + 0.35 * (0.05+0.58) = 0.84$$

Based on our ranking mechanism, Fig. 7 shows the ranking results. Fig. 8 shows the ranking results based on the Lucene's ranking mechanism.


"The detail of reviews:

'Title: A very good choice for lots of people - easy to carry, easy to use

'Helpful: 257 of 261 people found the following review helpful!

'Date: 3/24/2007

'Content: It is very small and fits easily in your pocket

'Title: Great Digital Camera

'Helpful: 15 of 18 people found the following review helpful!

'Date: 5/15/2007

'Content: This sexy little (yes its really small the size of a pack of cards)

'Title: Great Camera!!! Highly recommend!

'Helpful: 16 of 17 people found the following review helpful!

'Date: 4/10/2007

'Content: the camera is nice & small to just throw in my bag & go

Fig. 7. Search Results which are based on our ranking mechanism.


"The detail of reviews:

'Title: Great Digital Camera

'Helpful: 15 of 18 people found the following review helpful!

'Date: 5/15/2007

'Content: This sexy little (yes its really small the size of a pack of cards)

'Title: Great Camera!!! Highly recommend!

'Helpful: 16 of 17 people found the following review helpful!

'Date: 4/10/2007

'Content: the camera is nice & small to just throw in my bag & go

'Title: A very good choice for lots of people - easy to carry, easy to use

'Helpful: 257 of 261 people found the following review helpful!

'Date: 3/24/2007

'Content: It is very small and fits easily in your pocket

Fig. 8. Search Results which are based on Lucene's ranking mechanism.

From the above two ranking results, we can see that the most helpful review sentence gets the highest rank in our ranking mechanism, but it gets the lowest rank in Lucene rank. So our ranking mechanism can give more efficient opinion information.

Fig. 9 shows the customer reviews' changing trends with time. From the figure, product manufacturers can easily know the customers' concern about their products.

Fig. 10 shows the comparisons of positive and negative evaluation of a particular feature which the user is interested in. If the positive evaluation is preponderant, the feature of the product is likely good; otherwise the feature of the product may be bad.

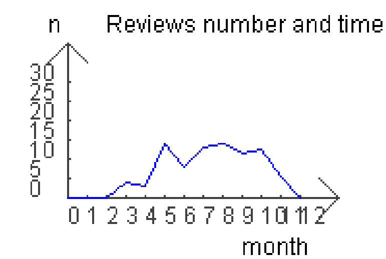

Fig. 9. The changing trend of reviews

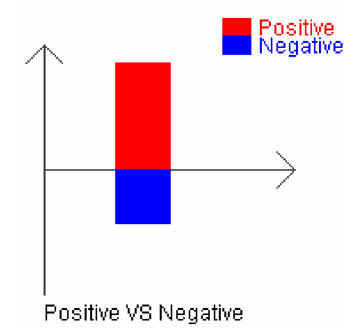

Fig. 10. The comparison of positive and negative evaluation of a particular feature.

VI. Experimental Evaluation

We conducted our experiments in the opinion search system using the customer reviews of two kinds of electronic products: 20 digital cameras and 20 cell phones. The reviews were collected from Amazon.com. For each camera and cell phone we crawled top 100 reviews. 8 participants issued their own queries and evaluated the search results under different α , each participant was asked to issue 20 queries.

Precision and recall are used as measures to evaluate the effects of the opinion search system.

TABLE II
PARAMETER DEFINITIONS OF PRECISION,RECALL

Value	Meaning
Ra	The relevant opinion results which has been retrieved by the system
R	All relevant opinion results about the query
A	All opinion results the system retrieved

$$Precision = \frac{Ra}{A}$$

$$\mathrm{Re}call = \frac{Ra}{R}$$

Where the value of Ra, R and A are defined in Table II. Table III shows the experimental results of our system.

TABLE III
EXPERIMENTAL RESULTS

α	Camera		Cell Phone	
	Recall	Precision	Recall	Precision
0.50	15.2%	81.5%	17.2%	80.2%
0.65	12.8%	85.3%	14.6&	82.5%
0.80	17.9%	78.8%	16.1%	79.3%

VII. CONCLUSION

Opinion search using customer reviews is still in its infancy. To the best of our knowledge, this is the first attempt to build an opinion search system including temporal dimension and search result visualization based on customer reviews. We have proposed a novel approach for retrieving opinion information from customer reviews. Our experimental results indicate that our opinion search system is promising in searching opinion from customer reviews.

In our future work, we plan to further improve and refine our techniques, i.e., extracting implicit opinion and opinions expressed with verbs. We will also address comparison search.

ACKNOWLEDGMENT

This research was supported by the project of The National High Technology Research and Development Program (863 program) of P.R.China under a research Grant No. 2006AA010106.

REFERENCES

[1] M. Hu and B. Liu. Mining Opinion Features in Customer Reviews. In Proc. of the 19th National Conf. on Artificial Intelligence (AAAI'04), pp. 755–760, 2004.

[2] M. Hu and B. Liu. Mining and Summarizing Customer Reviews. In Proc. of ACM SIGKDD Intl. Conf. on Knowledge Discovery and Data Mining (KDD'04), pp. 168–177, 2004.

[3] Philip S. Yu, Xin Li, and Bing Liu. "On the Temporal Dimension of Search", WWW-2004 poster paper.

[4] Philip S. Yu, Xin Li, Bing Liu. "Adding the Temporal Dimension to Search - A Case Study in Publication Search" To appear in IEEE/WIC/ACM International Conference on Web Intelligence (WI-05), September 19-22, 2005, Compiepne, France.

[5] Jian Liu, Gengfeng Wu, Jianxin Yao: Opinion Searching in Multi-Product Reviews. Proceedings of The Sixth IEEE International Conference on Computer and Information Technology (CIT'06).

[6] Silvana Aciar, Debbie Zhang, Simeon Simoff, John Debenham: Informed Recommender: Basing Recommendations on Consumer Product Reviews. Intelligent Systems, Volume 22, Issue 3, May-June 2007 Page(s):39 - 47

[7] Bing Liu, web data mining, Springer Berlin Heidelberg New York ISBN-10 3-540-37881-2.

[8] Bing Liu, Minqing Hu and Junsheng Cheng. "Opinion Observer: Analyzing and Comparing Opinions on the Web" To appear in Proceedings of the 14th international World Wide Web conference (WWW-2005), May 10-14, 2005, in Chiba, Japan.

[9] Liu, B., Hsu, W., Ma, Y. 1998. Integrating Classification and Association Rule Mining. KDD'98, 1998.

[10] Agrawal, R. & Srikant, R. 1994. Fast algorithm for mining association rules. VLDB'94, 1994.

[11] S. Kim and E. Hovy. Determining the Sentiment of Opinions. In Proc. of the Intl. Conf. on Computational Linguistics (COLING'04), 2004.

[12] A.-M. Popescu, and O. Etzioni. Extracting Product Features and Opinions from Reviews. In Proc. of Conference on Empirical Methods in Natural Language Processing (EMNLP'05), 2005.

[13] Osamu Furuse, Nobuaki Hiroshima, Setsuo Yamada, Ryoji Kataoka: Opinion Sentence Search Engine on Open-Domain Blog. IJCAI 2007: 2760-2765

[14] P. Turney. Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. In Proc. of the Meeting of the Association for Computational Linguistics (ACL'02), pp. 417–424, 2002.

[15] V. Hatzivassiloglou, and J. Wiebe. Effects of Adjective Orientation and Gradability on Sentence Subjectivity. In Proc. of the Intl. Conf. on Computational Linguistics (COLING'00), pp. 299–305. 2000.

[16] Bourigault, D. 1995. Lexter: A terminology extraction software for knowledge acquisition from texts. KAW'95.

[17] http://lucene.apache.org/