UIC at TREC 2008 Blog Track

Lifeng Jia, Clement Yu, Wei Zhang

Department of Computer Science

University of Illinois at Chicago
851 S Morgan St
Chicago, IL 60607, USA

{ljia, yu, wzhang}@cs.uic.edu

ABSTRACT

Our opinion retrieval system has four steps. In the first step, documents which are deemed relevant by the system with respect to the query are retrieved, without taking into consideration whether the documents are opinionative or not. In the second step, the abbreviations of query concepts in documents are recognized. This helps in identifying whether an opinion is in the vicinity of a query concept (which can be an abbreviation) in a document. The third step of opinion identification is designed for recognizing query-relevant opinions within the documents. In the forth step, for each query, all retrieved opinionated documents are ranked by various methods which take into account IR scores, opinion scores and the number of concepts in query. For the polarity subtask, the opinionative documents are classified into positive, negative and mixed types by two classifiers. Since TREC 2008 does not require mixed documents, all documents which are deemed mixed by our system are discarded.

1. INTRODUCTION

The opinion retrieval task was introduced in the TREC 2006 Blog Track [1]. In this task, a query-relevant document must have query-relevant opinions, regardless of the orientation of the opinions. Our TREC 2008 opinion retrieval system is based on our TREC 2007 system [2]. We consider the opinion retrieval as a four-step procedure. The first step is an information retrieval (IR) component that retrieves documents relevant to the query topics according to concept similarity and term similarity. Concept (phrase) identification, query expansion and document filtering are applied to optimize retrieval effectiveness. Abbreviation identification is the second step, which is a new component in our 2008 system to improve opinion identification effectiveness. The third step is opinion identification component that finds the general opinionated texts in the documents. This is a text classification process. The chi-square test [3] is applied to the training data to select features to build a support vector machine (SVM) opinion classifier. This classifier tests all the sentences of a document. Each sentence receives either a subjective or objective label. A document is opinionated with respect to the query if it has at least one subjective sentence, which is close to query concepts in the document. The abbreviations of query concepts identified in the second step are utilized in this step. In the forth step, both the IR score and the opinionative score of each document is used for ranking.

TREC 2008 Blog Track also has a sub-task, the polarity task. It requires a system to identify the orientation (polarity) of the opinions in an opinionated query-relevant document. The possible labels are positive, negative and mixed. A SVM classifier is built using training data containing positive and negative opinions from review sites. This classifier classifies each sentence in an opinionative document to be either positive or negative. Then, a document's polarity is determined by the orientations of query-relevant opinionative sentences within it. A positive (negative) document should be dominated by positive (negative) opinions. A mixed document should contain sufficient amount of both positive and negative opinions. Since TREC 2008 does not allow the mixed document category, all documents which are deemed mixed by our system are discarded.

The paper is organized as follows. Section 2 describes the IR module and abbreviation identification module of our opinion retrieval system. Section 3 describes the opinion identification module. Section 4 explains the modification in the ranking module. The polarity classification system is described in Section 5. Section 6 summarizes the performance of our submitted runs. Conclusions are given in Section 7.

2. INFORMATION RETRIEVAL AND ABBREVIATION IDENTIFICATION

The information retrieval module has four components: concept identification, query expansion, concept based retrieval and document filter. The abbreviation identification component is a new component, which identifies abbreviations of query concepts in documents. It improves the effectiveness of determining whether an opinion is related to the given query.

2.1 Concept Identification

A concept in a query is a multi-word phrase or a single word that denotes an entity. Four types of concepts are defined: proper nouns, dictionary phrases, simple phrases and complex phrases. The proper nouns are the noun phrases referring to people, place, event, organization, or other particular things. A dictionary phrase is a phrase that has an entry in a dictionary such as Wikipedia and Wordnet, but is not a proper noun. A simple phrase is a 2-word phrase, which is grammatically valid but is not a dictionary entry, e.g. "small car". A complex phrase has 3 or more words but is neither a proper noun nor a dictionary phrase. We developed an algorithm that combines several tools to identify the concepts in a query. We use Minipar [4], WordNet [5], and Wikipedia [6] and Google for proper noun and dictionary phrase identification. Collins Parser is used to find the simple phrase and complex phrase. Web search engine (Google) is also used for identifying simple phrases within complex phrases. The details of the algorithm can be found in [7].

2.2 Query Expansion

Query expansion is another technique in this information retrieval component. Two types of expansions are obtained: concept expansion and term expansion. In concept expansion, query concepts are recognized, disambiguated, if necessary and their synonyms are added. For example, for the query "cheney hunting", there are many possible interpretations of "cheney", according to Wikipedia [6]. But, by using the query word "hunting", "cheney" is disambiguated to "dick cheney", based on a descriptive page in Wikipedia. As an example for concept expansion, consider the query "china one child law". "China" has the synonym "prc" (People's Republic of China), while "one child law" has the synonym "one child policy". Thus, the query becomes "china one child law" OR "china one child policy" OR "prc one child law" OR "prc one child policy". Term expansion is carried out by the pseudo-feedback process in which terms in the vicinities of query terms in the top retrieved documents are extracted [8]. We apply this technique to three different collections and take the union of the extracted terms. Specifically, the TREC documents and Web documents (via the use of Google) are employed. In addition, if a page in Wikipedia is found to represent a query concept and frequent words in that page are extracted. The union of terms extracted from these three sources is taken as the set of expanded query terms.

2.3 Concept-Based Information Retrieval

After concepts identification and query expansion, an original query will be expanded with a list of concepts and their synonyms (if exists) and additional words. In our information retrieval module, the query-document similarity consists of two parts: the concept similarity and the term similarity (concept-sim, term-sim). The concept-sim is computed based on the identified concepts in common between the query and the document. The term-sim is the usual term similarity between the document and the query using the Okapi formula [9]. Each query term that appears in the document contributes to the term similarity, irrespective of whether it occurs in a concept or not. The concept-sim has a higher priority than the term-sim, since we emphasize that the concept is more important than individual terms. Consider, for a given query, two documents d1 and d2 having similarities (x_1, y_1) and (x_2, y_2), respectively. d1 will be ranked higher than d2 if either (1) $x_1 > x_2$, or (2) $x_1 = x_2$ and $y_1 > y_2$. Note that if $x_i > 0$, then the individual terms which contribute to concept-sim will ensure that $y_i > 0$. The calculation of concept-sim is described in [10].

2.4 Document Filter

Spamming is very common on the Web. Opinion retrieval effectiveness will be improved if the spam documents are removed. Three simple filtering rules are adopted. The first rule removes any document that contains long sentences. Sentences in the blog documents are usually short. This is especially true for the comments, as more people tend to leave brief comments. One type of spamming documents is that they have long sequences of words.

Hundred of words form a sentence. These words do not present any meaningful information, but it is retrievable by many queries. So we discard a blog document if it contains a sentence of *T* or more words, where *T* is a threshold that is empirically set to 300 in the experiment. The second rule aims to remove pornographic documents. Some blog documents are embedded with pornographic words to attract search traffic. We identify a list of pornographic words. Given a blog document, all its words are scanned to match the words in the list. If the total number of the occurrences of the words in the list is above a threshold in the document, this document is considered pornographic spam, and is discarded. The third rule removes documents written in foreign languages. We count the frequencies of some common English words and foreign words (Spanish and Italian by now). If the English word frequency is smaller than a threshold, and the foreign word frequency is greater than the threshold, we consider the document as written in the foreign language, and then discard it.

2.5 Abbreviation Identification

The *NEAR* operator that will be presented in section 3.2 checks the query terms and an opinionative sentence to be within a window of 5 sentences in order to determine whether or not an opinionative sentence is query-relevant. Sometimes the query concepts can not be identified in the window because they are not written in exactly the same way as they appear in the original query. [11] uses Wikipedia to collect such abbreviations as the synonyms of the phrases in the query. But if an abbreviation is not widely known, it is not defined in the Wikipedia, and it is not added.

Example 1. Given a query "*Global Positioning System*" and an opinionated sentence "*The 'stop-and-go' feature is great but the GPS is controlled by a knob which is bad*", this sentence won't be considered as relevant to the query if the system does not know "GPS" is the abbreviation for the query phrase.

In order to find more abbreviations, an "in-document-abbreviation-recognition" method is implemented to extract abbreviations of a query concept from an individual query relevant document. This method works as follows: given a query, if a document has been retrieved by the information retrieval module, the strings in the format of "x (y)" are searched in this document, where x is a multiple term concept that has been recognized in the query, and y is an abbreviation of x. For example, the two underlined parts in the sentence "*…the Global Positioning System (GPS) becomes fully operational …*" stand for the x and y respectively. If such abbreviation y is found, and y has not been recognized as a synonym of x before in Wikipedia, y is considered as the synonym of the corresponding concept x in this document ONLY, but not in any other documents, because we think that the author of this document might just casually introduce this abbreviation to save the time of writing. This abbreviation may be informal, so it is better to be cautious in not using it outside of this document. In the document containing the sentence in example 1, if the term *GPS* is found as the synonym of the query, then the opinionated sentence in the example 1 will be considered as query-relevant. By recognizing the in-document abbreviations of the query concepts, the *NEAR* operator has a higher chance of finding query terms, so more query relevant sentences can be recognized, which may result in more accurate opinion similarity scores.

3. OPINION IDENTIFICATION

The documents retrieved from information retrieval module can be categorized into (1) no opinion, (2) opinionated but not relevant to the query, and (3) opinionated and relevant to the query. Opinion retrieval module is composed of an opinion detection component (a SVM classifier) and a component with the *NEAR* operator. The opinion detection module identifies the opinions in the documents obtained from the IR module. Only those documents having opinions will be kept. All the opinions in a document are detected by that component. The opinions can be either relevant or irrelevant to the query. The *NEAR* operator decides the relevance of opinions.

3.1 Opinion Detection Component

In TREC 2008, we collect query-relevant training data for all 150 queries and then pool them together to create a whole training data set. A support vector machine (SVM) classifier that uses unigrams (single words) and bigrams (two adjacent words) as features is adopted. The vectors are presented in a presence-of-feature form, i.e. only the presence or absence of each feature is recorded in the vector, but not the number of occurrences of the feature.

This classifier-feature setup had been shown to be among the best configurations by Pang et al. [12]. The SVM-Light [13] is utilized with its default settings as the SVM implementation.

3.1.1 Partially Query-Independent Training Data Collection

For each of 150 TREC 2008 queries, the query-related subjective training data is collected from review Web sites and general opinionative Web pages. Each concept in a query is submitted to Rateitall.com where all the topics are organized in a tree structure in the review site. Once an entry is found, the reviews are collected. The reviews from other sibling nodes of the entry node are also collected in order to get enough amounts of training data. The site epinions.com is added as a new data source to collect query-related reviews too. A small set of "opinion indication phrases", such as "I think", "I don't think", "I like" and "I don't like", are used together with the query to collect opinionative Web pages. Each such phrase is submitted to a search engine with the query. The top ranked documents are collected as query-related review documents. To obtain the objective training data, the query concepts are searched in Wikipedia. If there is an entry page, the whole page is collected as the objective training data. The titles of the query's sibling nodes from Rateitall.com are also searched in Wikipedia to collect more objective training data. The details of this training data collecting procedure can be found in [11]. The pool of 150 query-relevant data forms the training data, so it is called partially query-independent training data collection, because it is not totally query-independent.

In addition, a lot of data from numerous topics, which are unrelated to the 150 queries are collected from rateitall.com. This is referred to query independent data. Upon collecting the reviews, we also record the scores of these reviews. A review score of 0 stands for a most critical opinion, while 5 stands for the most favorable opinion. Reviews with scores of 0 or 1 compose a "negative" training set. Reviews with scores of 4 or 5 form a "positive" training set. Reviews with scores of 2 and 3 are discarded due to their mixed polarities. This positive-negative query-independent training set contains the reviews from over 10 thousand topics.

3.1.2 Feature Selection by Partially Query-Independent Training Data

The unigrams and bigrams are treated as the features to train the SVM classifier. The Pearson's chi-square test [3] is adopted to select the features. Yang [14] reported that chi-square test is an effective feature selection approach. To find out how dependent a feature f is with respect to the subjective set and the objective set, a null hypothesis is set that f is independent of the two categories (subjective and objective) with respect to its occurrences in the two sets. [15] had shown that more features yields higher retrieval effectiveness. So, in addition, we also got more features by first partitioning the query-independent subjective training data into a positive set and a negative set and then conduct chi-square feature selection on these two set. The final features are the union of features from query-dependent subjective and objective training sets and those from query-independent positive and negative ones.

3.1.3 The Establishment of SVM Opinion Classifier

We establish an opinion classifier by using the obtained features. All the subjective/objective training data is converted to a vector representation of the features. Then we use the support vector machine (SVM) [12] learning program to train a classifier by using the vector data. When using the classifier, a document is split into a list of sentences. Each sentence is converted to a vector of the features. The classifier takes the vector as the input, and outputs a label (subjective or objective) and an associated score. Subjective sentence gets a positive score while objective sentence gets a negative score. The score represents the confidence level of the classifier to this answer. Larger absolute score means higher confidence, while a score close to 0 means low confidence. We define that a document is subjective (opinionative) if it has at least one sentence labeled as subjective.

3.2 The *NEAR* Operator

When a document is identified to have at least an opinionative sentence, it needs a further analysis by the *NEAR* operator to determine whether an opinion within the document is related to query. In TREC 2008, the *NEAR* Operator is redefined to check whether there is sufficient evidence that the query terms are within a window of 5 sentences from an opinion. The new rules of searching the query terms in the text window are:

1) If the query consists of one or more proper nouns, at least one complete proper noun (or its abbreviation) must be found in the text window.

2) If the query consists of one or more dictionary concepts (phrases that can be found in a dictionary such as Wikipedia), at least one complete dictionary phrase (or its abbreviation) must be found in the text window.

3) If the query contains both a proper noun and a dictionary phrase, at least two original query terms must be found in the text window.

4) If the query contains two or more content words, and it does not contain multi-word proper noun or multi-word dictionary phrase, at least three original query terms or expanded query terms must be found in the text window.

4. OPINIONATIVE DOCUMENT RANKING

To rank opinionated relevant documents, we utilized a batch of methods which take into consideration of the IR score, the number of or the sum of SVM scores of opinionative query relevant sentences within opinionated relevant documents. For example, the total score of an opinionated relevant document is the weighted sum of its IR similarity scores and its opinion score (such as the number of query relevant opinionative sentences within it). The weights assigned to the two component scores are equal. The detailed information concerning the ranking methods can be referred in [11]. However, this assignment of equal weights may create problems for queries having multiple concepts. For example, the query "*tax break for hybrid automobiles*", documents about "*hybrid automobiles*" may contain substantial opinions but have nothing to do with tax breaks while documents about the entire query may have fewer opinions. Thus, our strategy is as follows. For a query having a single concept, the score of a document is not changed i.e. it is a weighted sum of its IR similarity score and its opinion score; the weight being equal for the two components. For a query having multiple concepts, the opinion score of a document having all query concepts will be emphasized over that of a document having fewer query concepts, because the latter document is relevant to some aspects of the query and not necessarily about the entire query.

5. OPINION POLARITY CLASSIFICATION

The opinion retrieval system distinguishes the subjective texts from the objective texts. But it does not distinguish the positive opinions from the negative ones within the subjective texts. To determine the polarities of opinionated documents, we propose a two-stage classification model. The proposed model takes the opinionated documents from the opinion retrieval system as input. In the first stage, this model categorizes every query-relevant opinionative sentence within a document as either positive or negative. In the second stage, this model adopts a second classifier to designate the document as positive, negative or mixed, according to the overall tone of opinions in the document. Fig. 1 shows the architecture of our polarity system. Since TREC 2008 allows positive and negative labels only, the mixed documents are discarded.

5.1 Sentence-Level Opinion Polarity Classification

The first classification stage aims to classify a query relevant opinionative sentence as either positive or negative. It is very similar to the case of classifying a sentence as either subjective or objective in the opinion detection module described in Section 3.1.3. Consequently, the SVM classifier is adopted here to determine the polarity of a query-relevant opinionative sentence. To train this classifier, query-independent positive and negative training data which is described in Section 3.1.1 are prepared for the Chi-square feature selection. This classifier takes a query-relevant opinionative sentence as input. It designates the sentence a positive or negative label, depending on a classification score. For a retrieved document from the opinion retrieval system, each query relevant opinionative sentence in an opinionated query-relevant is designated a polarity label and a confidence score. This information will be used in the second stage.
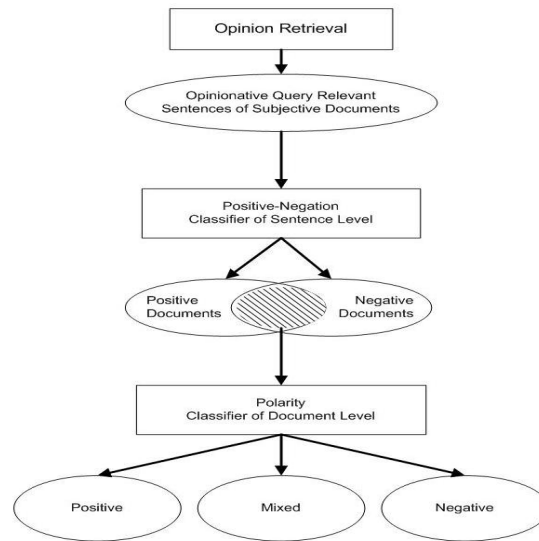
Figure 1. The architecture of the polarity classification system.

5.2 Document-Level Opinion Polarity Classification

The second stage of the proposed polarity classification model determines the overall opinion polarity of a document, based on the polarities of its query-relevant opinionative sentences. The document should be positive (or negative) if it only contains positive (or negative) query relevant opinions. It contains mixed opinions if both sufficient positive and sufficient negative opinions are found.

5.2.1 A Heuristic Rule Based Method

The polarity classification system [2] was developed based on the following intuition: a document is positive (negative) if it only contains positive (negative) relevant opinions. If the document contains both kinds of opinions, it needs further analysis. The opinion polarity of this document should be mixed if both the positive and the negative relevant opinions are approximately equal in strength in this document. If the positive (negative) relevant opinions are significantly stronger than the negative (positive) relevant opinions, the opinion polarity of this document should be positive (negative). In order to compare the positive opinions with the negative opinions in a document, [2] defined a set of features to measure the strength of the opinions. For example, a feature can be the number of sentences in a document that are classified to be positive relevant. More details concerning features and this heuristic rule based method can be found in [2].

5.2.2 Proposed Decision Tree Method

Although the above rule-based model achieved the highest classification accuracy in TREC 2007, the features in [2] may not be appropriately utilized. A machine learning method is proposed to improve the document-level opinion polarity classification accuracy. This method utilizes the query relevant opinionated documents, their polarity designations in the TREC official golden standard, and the positive/negative sentence information obtained from the first sentence-level classifier to train a secondary document-level classifier. Specifically, the feature set sketched in Section 5.2.1 is utilized. A vector is formed for each document whose polarity determined by our system is consistent to the gold standard vector. This forms the training set. For example, the TREC 2006 data is used to train a classifier to test the TREC 2007 queries. Then these vectors are fed into Quinlan's C4.5 decision tree program [16] to generate the classifier. The classifier will take a list of values of the features as the input and gives out a positive, negative or mixed label as the output. Similarly, we utilize the TREC 2007 data as training data to establish a classifier for the TREC 2006 queries. The data of TREC2006 and TREC2007 are unified as the training set to establish the classifier to test TREC2008.

6. EXPERIMENT RESULTS

For TREC2008, 5 baselines which are produced by participants of TREC are given to us for evaluation. Each baseline consists of at most 1000 documents for each query which are ranked in descending order of IR scores without considering whether they are opinionative or not. Therefore, we submit totally 21 opinion runs based on 6 baselines (5 baselines plusing our own baselines) where we applied our opinion identification technologies on. For those common 5 baselines, we designated their runids as uicopiblj(r), where i = 1 or 2, j = 1, 2, … , 5, r is an optional identifier representing re-ranking of baseline. For our own baseline, the runids are designated as uicopruni, where i = 1 or 2. The annotation of these runids is explained in the table below.

RunID	Description
uicop1blj	According to the baseline j, the opinion retrieval runs without the emphasis on documents containing all query concepts.
uicop1bljr	Documents in the baseline j which are deemed by our system to be without relevant opinions are attached at the bottom of uicop1blj according to descending order of the IR score. "r" here stands for the re-ranking of documents.
uicop2blj	According to the baseline j, the opinion retrieval runs with the emphasis on documents containing all query concepts.
uicop2bljr	Documents in the baseline j which are deemed by our system to be without relevant opinions are attached at the bottom of uicop2blj according to the IR score. "r" here stands for the re-ranking of documents.
uicoprun1	According to our own baseline, the opinion retrieval runs without the emphasis on documents containing all query concepts.
uicoprun2	According to our own baseline, the opinion retrieval runs with the emphasis on documents containing all query concepts.

Table 1. The annotation of opinion runids from UIC

For all 50 TREC 2008 queries, table 2 and table 3 show the MAP and R-Precision scores of each opinion run based on various baselines. All runs where the opinionated documents containing all query concepts are given higher priorities than the documents that contain fewer concepts perform slightly better than the runs without the emphasis on the multiple term concepts, because not all queries can benefits from this modification. Moreover, re-ranking runs outperform those corresponding runs because those documents which is not retrieved by our system but attached at the bottom of the ranking contribute to the performance enhancement.

	uicop1blj	uicop1bljr	uicop2blj	uicop2bljr
Baseline1	0.4303	0.4576	0.4314	N|A
Baseline2	0.3209	0.3457	0.3277	0.3525
Baseline3	0.4267	0.4483	0.4444	0.4663
Baseline4	0.4281	0.4529	0.4476	0.4726
Baseline5	0.3670	0.3866	0.3768	0.3965
	uicoprun1		uicoprun2	
Own Baseline	0.4461		0.4473	

Table 2. The MAP score of all opinion runs from UIC

	uicop1blj	uicop1bljr	uicop2blj	uicop2bljr
Baseline1	0.4837	0.4953	0.4839	N|A
Baseline2	0.3816	0.3902	0.3891	0.3977
Baseline3	0.4721	0.4752	0.4842	0.4874
Baseline4	0.4851	0.4897	0.5027	0.5072
Baseline5	0.4386	0.4454	0.4428	0.4497
	uicoprun1		uicoprun2	
Own Baseline	0.4822		0.4822	

Table 3. The R-Precision score of all opinion runs from UIC

In the polarity subtask, we submitted 10 polarity runs on the basis of 10 opinion runs. Table 4 and table 5 present the Map and R-Precision scores of positive and negative rankings respectively, according to 50 TREC 2008 queries only. We note that the polarity system does not perform as well as opinion retrieval system. One possible reason is our first feature-based classifier on the sentence level did not specially handle properly the sentences with occurrences of negation words, which might flip the orientation of opinion of a sentence. Another possible reason is that the computation of features of training data for the secondary classifier, for example, the number of positive relevant sentences. Because the golden standard only points out the polarity of document, but not provide further detail information, such as which sentences are relevant opinionative ones and how strong the opinion is, so we have to depend on the information from the classification results of the first-stage, which might not totally accurate.

RunID	Corresponding Opinion RunID	Positive Ranking	Negative Ranking
uicpo1bl1	uicop1bl1	0.1548	0.0576
uicpol1bl2	uicop1bl2	0.1094	0.0554
uicpol2bl2	uicop2bl2	0.1120	0.0536
uicpo1bl3	uicop1bl3	0.1442	0.0667
uicpol2bl3	uicop2bl3	0.1449	0.0651
uicpol1bl4	uicop1bl4	0.1542	0.0681
uicpol2bl4	uicop2bl4	0.1552	0.0655
uicpo1bl5	uicop1bl5	0.1072	0.0400
uicpol2bl5	uicop2bl5	0.1081	0.0423
uicpolrun1	uicoprun2	0.1627	0.0609

Table 4. The MAP score of positive and negative rankings

RunID	Corresponding Opinion RunID	Positive Ranking	Negative Ranking
uicpo11bl1	uicop1bl1	0.2221	0.1068

uicpol1bl2	uicop1bl2	0.1623	0.1063
uicpol2bl2	uicop2bl2	0.1692	0.1058
uicpo1bl3	uicop1bl3	0.2039	0.1155
uicpol2bl3	uicop2bl3	0.2059	0.1124
uicpol1bl4	uicop1bl4	0.2198	0.1386
uicpol2bl4	uicop2bl4	0.2218	0.1346
uicpo1bl5	uicop1bl5	0.1561	0.0974
uicpol2bl5	uicop2bl5	0.1577	0.0984
uicpolrun1	uicoprun2	0.2198	0.1065

Table 5. The R-Precision score of positive and negative rankings

7. CONCLUSIONS

In the opinion retrieval task of the TREC 2008 Blog Track, we develop a four-step algorithm to retrieve documents that have subjective content about a query topic. The system has the new features such as the new method of finding abbreviation of concepts, the new way of using the training data, and more emphasis over documents with all concepts than ones with fewer concepts. For the polarity classification task, we adopted a "split-and-merge" strategy to distinguish the three kinds of opinions. A SVM classifier is first designed to designate the orientation of opinion on the level of sentence. Then, a decision tree classifier is established to determine the polarity of opinionated document.

8. REFERENCES

[1] Iadh Ounis, Maarten de Rijke, Craig Macdonald, Gilad Mishne, Ian Soboroff. Overview of the TREC-2006 Blog Track. In proceedings of the 15th TREC. 2006.

[2] Wei Zhang and Clement Yu. UIC at TREC 2007 Blog Track. In TREC. 2007.

[3] Chernoff H, Lehmann E.L. The use of maximum likelihood estimates in χ^2 tests for goodness-of-fit. The Annals of Mathematical Statistics 1954; 25:579-586.

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

[5] http://wordnet.princeton.edu/

[6] http://en.wikipedia.or

[7] Wei Zhang, Shuang Liu, Clement Yu, Chaojing Sun, Fang Liu and Weiyi Meng. Recognition and Classification of Noun Phrases in Queries for Effective Retrieval. In proceedings of the 16th CIKM. 2007.

[8] Jinxi Xu and W Croft. Query Expansion Using Local and Global Document Analysis. In proceedings of SIGIR. 1996.

[9] S. Robertson, S. Walker Okapi/Keenbow at TREC-8, 1999.

[10] Shuang Liu, Fang Liu, Clement Yu, and Weiyi Meng. An Effective Approach to Document Retrieval via Utilizing WordNet and Recognizing Phrases. In Proceedings of the 27th Annual International ACM SIGIR Conference. 2004.

[11] Wei Zhang and Clement Yu. UIC at TREC 2006 Blog Track. In proceedings of the 15th TREC. 2006

[12] B Pang, L Lee and S Vaithyanathan. Thumbs up? Sentiment Classification Using Machine Learning Techniques. EMNLP'02, 2002.

[13] T Joachims. Making large-scale SVM learning practical. *Advances in Kernel Methods: Support Vector Learning*. 1999.

[14] Y Yang and J Pederson. A comparative study on feature selection in text categorization. In *Proc. of ICML.* 1997.

[15] Wei Zhang, Clement Yu and Weiyi Meng. "Opinion Retrieval from Blogs". In proceedings of CIKM. 2007.

[16] J. R. Quinlan. C4.5: Programs for Machine Learning. Morgan-Kaufman, 1993.