Answering Opinion Questions with Random Walks on Graphs

Fangtao Li, Yang Tang, Minlie Huang, and Xiaoyan Zhu
State Key Laboratory on Intelligent Technology and Systems
Tsinghua National Laboratory for Information Science and Technology
Department of Computer Sci. and Tech., Tsinghua University, Beijing 100084, China
{fangtao06,tangyang9}@gmail.com,{aihuang,zxy-dcs}@tsinghua.edu.cn

Abstract

Opinion Question Answering (*Opinion QA*), which aims to find the authors' sentimental opinions on a specific target, is more challenging than traditional fact-based question answering problems. To extract the opinion oriented answers, we need to consider both topic relevance and opinion sentiment issues. Current solutions to this problem are mostly ad-hoc combinations of question topic information and opinion information. In this paper, we propose an *Opinion PageRank* model and an *Opinion HITS* model to fully explore the information from different relations among questions and answers, answers and answers, and topics and opinions. By fully exploiting these relations, the experiment results show that our proposed algorithms outperform several state of the art baselines on benchmark data set. A gain of over 10% in F scores is achieved as compared to many other systems.

1 Introduction

Question Answering (QA), which aims to provide answers to human-generated questions automatically, is an important research area in natural language processing (NLP) and much progress has been made on this topic in previous years. However, the objective of most state-of-the-art QA systems is to find answers to factual questions, such as "What is the longest river in the United States?" and "Who is Andrew Carnegie?" In fact, rather than factual information, people would also like to know about others' opinions, thoughts and feelings toward some specific objects, people and events. Some examples of these questions are: "How is Bush's decision not to ratify the Kyoto Protocol looked upon by Japan and other US allies?"(Stoyanov et al., 2005) and "Why do people like Subway Sandwiches?" from TAC 2008 (Dang, 2008). Systems designed to deal with such questions are called *opinion QA systems*. Researchers (Stoyanov et al., 2005) have found that opinion questions have very different characteristics when compared with fact-based questions: opinion questions are often much longer, more likely to represent partial answers rather than complete answers and vary much more widely. These features make opinion QA a harder problem to tackle than fact-based QA. Also as shown in (Stoyanov et al., 2005), directly applying previous systems designed for fact-based QA onto opinion QA tasks would not achieve good performances.

Similar to other complex QA tasks (Chen et al., 2006; Cui et al., 2007), the problem of opinion QA can be viewed as a sentence ranking problem. The Opinion QA task needs to consider not only the topic relevance of a sentence (to identify whether this sentence matches the topic of the question) but also the sentiment of a sentence (to identify the opinion polarity of a sentence). Current solutions to opinion QA tasks are generally in ad hoc styles: the topic score and the opinion score are usually separately calculated and then combined via a linear combination (Varma et al., 2008) or just filter out the candidate without matching the question sentiment (Stoyanov et al., 2005). However, topic and opinion are not independent in reality. The opinion words are closely associated with their contexts. Another problem is that existing algorithms compute the score for each answer candidate individually, in other words, they do not consider the relations between answer candidates. The quality of a answer candidate is not only determined by the relevance to the question, but also by other candidates. For example, the good answer may be mentioned by many candidates.

In this paper, we propose two models to address the above limitations of previous sentence

Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP, pages 737–745,
Suntec, Singapore, 2-7 August 2009.

ranking models. We incorporate both the topic relevance information and the opinion sentiment information into our sentence ranking procedure. Meanwhile, our sentence ranking models could naturally consider the relationships between different answer candidates. More specifically, our first model, called Opinion PageRank, incorporates opinion sentiment information into the graph model as a condition. The second model, called Opinion HITS model, considers the sentences as authorities and both question topic information and opinion sentiment information as hubs. The experiment results on the TAC QA data set demonstrate the effectiveness of the proposed Random Walk based methods. Our proposed method performs better than the best method in the TAC 2008 competition.

The rest of this paper is organized as follows: Section 2 introduces some related works. We will discuss our proposed models in Section 3. In Section 4, we present an overview of our opinion QA system. The experiment results are shown in Section 5. Finally, Section 6 concludes this paper and provides possible directions for future work.

2 Related Work

Few previous studies have been done on opinion QA. To our best knowledge, (Stoyanov et al., 2005) first created an opinion QA corpus OpQA. They find that opinion QA is a more challenging task than factual question answering, and they point out that traditional fact-based QA approaches may have difficulty on opinion QA tasks if unchanged. (Somasundaran et al., 2007) argues that making finer grained distinction of subjective types (sentiment and arguing) further improves the QA system. For non-English opinion QA, (Ku et al., 2007) creates a Chinese opinion QA corpus. They classify opinion questions into six types and construct three components to retrieve opinion answers. Relevant answers are further processed by focus detection, opinion scope identification and polarity detection. Some works on opinion mining are motivated by opinion question answering. (Yu and Hatzivassiloglou, 2003) discusses a necessary component for an opinion question answering system: separating opinions from fact at both the document and sentence level. (Soo-Min and Hovy, 2005) addresses another important component of opinion question answering: finding opinion holders.

More recently, TAC 2008 QA track (evolved from TREC) focuses on finding answers to opinion questions (Dang, 2008). Opinion questions retrieve sentences or passages as answers which are relevant for both question topic and question sentiment. Most TAC participants employ a strategy of calculating two types of scores for answer candidates, which are the topic score measure and the opinion score measure (the opinion information expressed in the answer candidate). However, most approaches simply combined these two scores by a weighted sum, or removed candidates that didn't match the polarity of questions, in order to extract the opinion answers.

Algorithms based on Markov Random Walk have been proposed to solve different kinds of ranking problems, most of which are inspired by the PageRank algorithm (Page et al., 1998) and the HITS algorithm (Kleinberg, 1999). These two algorithms were initially applied to the task of Web search and some of their variants have been proved successful in a number of applications, including fact-based QA and text summarization (Erkan and Radev, 2004; Mihalcea and Tarau, 2004; Otterbacher et al., 2005; Wan and Yang, 2008). Generally, such models would first construct a directed or undirected graph to represent the relationship between sentences and then certain graph-based ranking methods are applied on the graph to compute the ranking score for each sentence. Sentences with high scores are then added into the answer set or the summary. However, to the best of our knowledge, all previous Markov Random Walk-based sentence ranking models only make use of topic relevance information, i.e. whether this sentence is relevant to the fact we are looking for, thus they are limited to fact-based QA tasks. To solve the opinion QA problems, we need to consider both topic and sentiment in a non-trivial manner.

3 Our Models for Opinion Sentence Ranking

In this section, we formulate the opinion question answering problem as a topic and sentiment based sentence ranking task. In order to naturally integrate the topic and opinion information into the graph based sentence ranking framework, we propose two random walk based models for solving the problem, i.e. an Opinion PageRank model and an Opinion HITS model.

3.1 Opinion PageRank Model

In order to rank sentence for opinion question answering, two aspects should be taken into account. First, the answer candidate is relevant to the question topic; second, the answer candidate is suitable for question sentiment.

Considering Question Topic: We first introduce how to incorporate the question topic into the Markov Random Walk model, which is similar as the Topic-sensitive LexRank (Otterbacher et al., 2005). Given the set $V_s = \{v_i\}$ containing all the sentences to be ranked, we construct a graph where each node represents a sentence and each edge weight between sentence v_i and sentence v_j is induced from sentence similarity measure as follows: $p(i \rightarrow j) = \frac{f(i \rightarrow j)}{\sum_{k=1}^{|V_s|} f(i \rightarrow k)}$, where $f(i \rightarrow j)$ represents the similarity between sentence v_i and sentence v_j, here is cosine similarity (Baeza-Yates and Ribeiro-Neto, 1999). We define $f(i \rightarrow i) = 0$ to avoid self transition. Note that $p(i \rightarrow j)$ is usually not equal to $p(j \rightarrow i)$. We also compute the similarity $rel(v_i|q)$ of a sentence v_i to the question topic q using the cosine measure. This relevance score is then normalized as follows to make the sum of all relevance values of the sentences equal to 1: $rel'(v_i|q) = \frac{rel(v_i|q)}{\sum_{k=1}^{|V_s|} rel(v_k|q)}$.

The saliency score $Score(v_i)$ for sentence v_i can be calculated by mixing topic relevance score and scores of all other sentences linked with it as follows: $\text{Score}(v_i) = \mu \sum_{j \neq i} \text{Score}(v_j) \cdot p(j \rightarrow i) + (1-\mu) rel'(v_i|q)$, where μ is the damping factor as in the PageRank algorithm.

The matrix form is: $\tilde{p} = \mu \tilde{M}^T \tilde{p} + (1-\mu)\vec{\alpha}$, where $\tilde{p} = [Score(v_i)]_{|V_s| \times 1}$ is the vector of saliency scores for the sentences; $\tilde{M} = [p(i \rightarrow j)]_{|V_s| \times |V_s|}$ is the graph with each entry corresponding to the transition probability; $\vec{\alpha} = [rel'(v_i|q)]_{|V_s| \times 1}$ is the vector containing the relevance scores of all the sentences to the question. The above process can be considered as a Markov chain by taking the sentences as the states and the corresponding transition matrix is given by $A' = \mu \tilde{M}^T + (1-\mu)\vec{e}\vec{\alpha}^T$.

Considering Topics and Sentiments Together: In order to incorporate the opinion information and topic information for opinion sentence ranking in an unified framework, we propose an Opinion PageRank model (Figure 1) based on a two-layer link graph (Liu and Ma, 2005; Wan and Yang, 2008). In our opinion PageRank model, the first layer contains all the sentiment words from a lexicon to represent the opinion information, and the second layer denotes the sentence relationship in the topic sensitive Markov Random Walk model discussed above. The dashed lines between these two layers indicate the conditional influence between the opinion information and the sentences to be ranked.

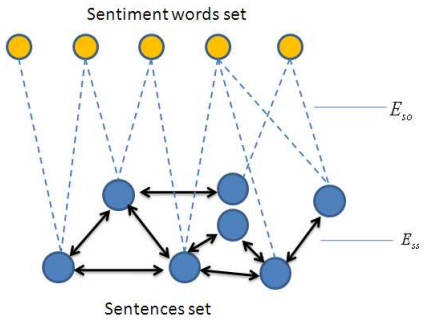

Figure 1: Opinion PageRank

Formally, the new representation for the two-layer graph is denoted as $G^* = \langle V_s, V_o, E_{ss}, E_{so} \rangle$, where $V_s = \{v_i\}$ is the set of sentences and $V_o = \{o_j\}$ is the set of sentiment words representing the opinion information; $E_{ss} = \{e_{ij}|v_i, v_j \in V_s\}$ corresponds to all links between sentences and $E_{so} = \{e_{ij}|v_i \in V_s, o_j \in V_o\}$ corresponds to the opinion correlation between a sentence and the sentiment words. For further discussions, we let $\pi(o_j) \in [0,1]$ denote the sentiment strength of word o_j, and let $\omega(v_i, o_j) \in [0,1]$ denote the strength of the correlation between sentence v_i and word o_j. We incorporate the two factors into the transition probability from v_i to v_j and the new transition probability $p(i \rightarrow j|\text{Op}(v_i), \text{Op}(v_j))$ is defined as $\frac{f(i \rightarrow j|\text{Op}(v_i), \text{Op}(v_j))}{\sum_{k=1}^{|V_s|} f(i \rightarrow k|\text{Op}(v_i), \text{Op}(v_k))}$ when $\sum f \neq 0$, and defined as 0 otherwise, where $\text{Op}(v_i)$ is denoted as the opinion information of sentence v_i, and $f(i \rightarrow j|\text{Op}(v_i), \text{Op}(v_j))$ is the new similarity score between two sentences v_i and v_j, conditioned on the opinion information expressed by the sentiment words they contain. We propose to compute the conditional similarity score by linearly combining the scores conditioned on the source opinion (i.e. $f(i \rightarrow j|\text{Op}(v_i))$) and the destination opinion (i.e. $f(i \rightarrow j|\text{Op}(v_j))$) as follows:

$$
\begin{aligned}
& f(i \rightarrow j|\text{Op}(v_i), \text{Op}(v_j)) \\
& = \lambda \cdot f(i \rightarrow j|\text{Op}(v_i)) + (1-\lambda) \cdot f(i \rightarrow j|\text{Op}(v_j)) \\
& = \lambda \cdot \sum_{o_k \in \text{Op}(v_i)} f(i \rightarrow j) \cdot \pi(o_k) \cdot \omega(o_k, v_i) \\
& + (1-\lambda) \cdot \sum_{o_{k'} \in \text{Op}(v_j))} (i \rightarrow j) \cdot \pi(o_{k'}) \cdot \omega(o_{k'}, v_j) \quad (1)
\end{aligned}
$$

where $\lambda \in [0,1]$ is the combination weight controlling the relative contributions from the source

opinion and the destination opinion. In this study, for simplicity, we define $\pi(o_j)$ as 1, if o_j exists in the sentiment lexicon, otherwise 0. And $\omega(v_i, o_j)$ is described as an indicative function. In other words, if word o_j appears in the sentence v_i, $\omega(v_i, o_j)$ is equal to 1. Otherwise, its value is 0. Then the new row-normalized matrix $\tilde{M}^*$ is defined as follows: $\tilde{M}^*_{ij} = p(i \rightarrow j|\mathrm{Op}(i), \mathrm{Op}j)$.

The final sentence score for Opinion PageRank model is then denoted by: $\mathrm{Score}(v_i) = \mu \cdot \sum_{j \neq i} \mathrm{Score}(v_j) \cdot \tilde{M}^*_{ji} + (1-\mu) \cdot rel'(s_i|q)$.

The matrix form is: $\tilde{p} = \mu \tilde{M}^{*T} \tilde{p} + (1-\mu) \cdot \vec{\alpha}$. The final transition matrix is then denoted as: $A^* = \mu \tilde{M}^{*T} + (1-\mu)\vec{e}\vec{\alpha}^T$ and the sentence scores are obtained by the principle eigenvector of the new transition matrix A^*.

3.2 Opinion HITS Model

The word's sentiment score is fixed in Opinion PageRank. This may encounter problem when the sentiment score definition is not suitable for the specific question. We propose another opinion sentence ranking model based on the popular graph ranking algorithm HITS (Kleinberg, 1999). This model can dynamically learn the word sentiment score towards a specific question. HITS algorithm distinguishes the hubs and authorities in the objects. A hub object has links to many authorities, and an authority object has high-quality content and there are many hubs linking to it. The hub scores and authority scores are computed in a recursive way. Our proposed opinion HITS algorithm contains three layers. The upper level contains all the sentiment words from a lexicon, which represent their opinion information. The lower level contains all the words, which represent their topic information. The middle level contains all the opinion sentences to be ranked. We consider both the opinion layer and topic layer as hubs and the sentences as authorities. Figure 2 gives the bipartite graph representation, where the upper opinion layer is merged with lower topic layer together as the hubs, and the middle sentence layer is considered as the authority.

Formally, the representation for the bipartite graph is denoted as $G^{\#} = \langle V_s, V_o, V_t, E_{so}, E_{st} \rangle$, where $V_s = \{v_i\}$ is the set of sentences. $V_o = \{o_j\}$ is the set of all the sentiment words representing opinion information, $V_t = \{t_j\}$ is the set of all the words representing topic information. $E_{so} = \{e_{ij}|v_i \in V_s, o_j \in V_o\}$ corresponds to the

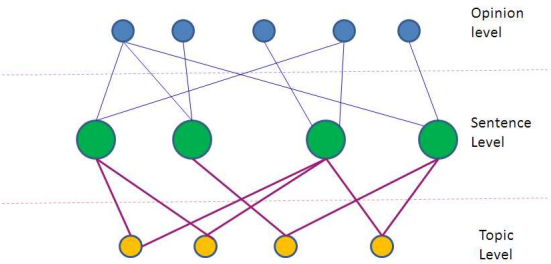

Figure 2: Opinion HITS model

correlations between sentence and opinion words. Each edge e_{ij} is associated with a weight ow_{ij} denoting the strength of the relationship between the sentence v_i and the opinion word o_j. The weight ow_{ij} is 1 if the sentence v_i contains word o_j, otherwise 0. E_{st} denotes the relationship between sentence and topic word. Its weight tw_{ij} is calculated by $tf \cdot idf$ (Otterbacher et al., 2005).

We define two matrixes $O = (O_{ij})_{|V_s| \times |V_o|}$ and $T = (T_{ij})_{|V_s| \times |V_t|}$ as follows, for $O_{ij} = ow_{ij}$, and if sentence i contains word j, therefore ow_{ij} is assigned 1, otherwise ow_{ij} is 0. $T_{ij} = tw_{ij} = tf_j \cdot idf_j$ (Otterbacher et al., 2005).

Our new opinion HITS model is different from the basic HITS algorithm in two aspects. First, we consider the topic relevance when computing the sentence authority score based on the topic hub level as follows: $\mathrm{Auth}_{\mathrm{sen}}(v_i) \propto \sum_{tw_{ij}>0} tw_{ij} \cdot \mathrm{topic\_score}(j) \cdot \mathrm{hub}_{\mathrm{topic}}(j)$, where $\mathrm{topic\_score}(j)$ is empirically defined as 1, if the word j is in the topic set (we will discuss in next section), and 0.1 otherwise.

Second, in our opinion HITS model, there are two aspects to boost the sentence authority score: we simultaneously consider both topic information and opinion information as hubs.

The final scores for authority sentence, hub topic and hub opinion in our opinion HITS model are defined as:

$$\mathrm{Auth}^{(n+1)}_{\mathrm{sen}}(v_i) = \gamma \cdot \sum_{tw_{ij}>0} tw_{ij} \cdot \mathrm{topic\_score}(j) \cdot \mathrm{Hub}^{(n)}_{\mathrm{topic}}(t_j) + (1-\gamma) \cdot \sum_{ow_{ij}>0} ow_{ij} \cdot \mathrm{Hub}^{(n)}_{\mathrm{opinion}}(o_j) \quad (2)$$

$$\mathrm{Hub}^{(n+1)}_{\mathrm{topic}}(t_i) = \sum_{tw_{ki}>0} tw_{ki} \cdot \mathrm{Auth}^{(n)}_{\mathrm{sen}}(v_i) \quad (3)$$

$$\mathrm{Hub}^{(n+1)}_{\mathrm{opinion}}(o_i) = \sum_{ow_{ki}>0} ow_{ki} \cdot \mathrm{Auth}^{(n)}_{\mathrm{sen}}(v_i) \quad (4)$$

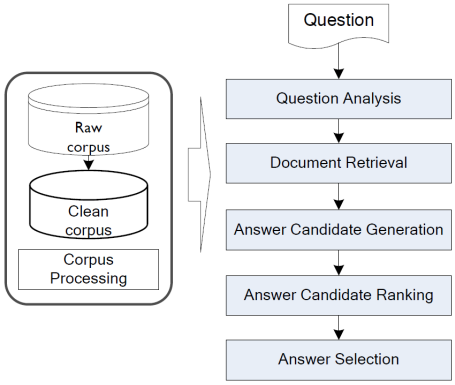

Figure 3: Opinion Question Answering System

The matrix form is:

$$a^{(n+1)} = \gamma \cdot T \cdot e \cdot t_s^T \cdot I \cdot h_t^{(n)} + (1-\gamma) \cdot O \cdot h_o^{(n)} \quad (5)$$

$$h_t^{(n+1)} = T^T \cdot a^{(n)} \quad (6)$$

$$h_o^{(n+1)} = O^T \cdot a^{(n)} \quad (7)$$

where e is a $|V_t| \times 1$ vector with all elements equal to 1 and I is a $|V_t| \times |V_t|$ identity matrix, $t_s = [\text{topic\_score}(j)]_{|V_t| \times 1}$ is the score vector for topic words, $a^{(n)} = [\text{Auth}_{\text{sen}}^{(n)}(v_i)]_{|V_s| \times 1}$ is the vector authority scores for the sentence in the n^{th} iteration, and the same as $h_t^{(n)} = [\text{Hub}_{\text{topic}}^{(n)}(t_j)]_{|V_t| \times 1}$, $h_o^{(n)} = [\text{Hub}_{\text{opinion}}^{(n)}(t_j)]_{|V_o| \times 1}$. In order to guarantee the convergence of the iterative form, authority score and hub score are normalized after each iteration.

For computation of the final scores, the initial scores of all nodes, including sentences, topic words and opinion words, are set to 1 and the above iterative steps are used to compute the new scores until convergence. Usually the convergence of the iteration algorithm is achieved when the difference between the scores computed at two successive iterations for any nodes falls below a given threshold (10e-6 in this study). We use the authority scores as the saliency scores in the Opinion HITS model. The sentences are then ranked by their saliency scores.

4 System Description

In this section, we introduce the opinion question answering system based on the proposed graph methods. Figure 3 shows five main modules:

Question Analysis: It mainly includes two components. 1).Sentiment Classification: We classify all opinion questions into two categories: positive type or negative type. We extract several types of features, including a set of pattern features, and then design a classifier to identify sentiment polarity for each question (similar as (Yu and Hatzivassiloglou, 2003)). 2).Topic Set Expansion: The opinion question asks opinions about a particular target. Semantic role labeling based (Carreras and Marquez, 2005) and rule based techniques can be employed to extract this target as topic word. We also expand the topic word with several external knowledge bases: Since all the entity synonyms are redirected into the same page in Wikipedia (Rodrigo et al., 2007), we collect these redirection synonym words to expand topic set. We also collect some related lists as topic words. For example, given question "What reasons did people give for liking Ed Norton's movies?", we collect all the Norton's movies from IMDB as this question's topic words.

Document Retrieval: The PRISE search engine, supported by NIST (Dang, 2008), is employed to retrieve the documents with topic word.

Answer Candidate Extraction: We split retrieved documents into sentences, and extract sentences containing topic words. In order to improve recall, we carry out the following process to handle the problem of coreference resolution: We classify the topic word into four categories: male, female, group and other. Several pronouns are defined for each category, such as "he", "him", "his" for male category. If a sentence is determined to contain the topic word, and its next sentence contains the corresponding pronouns, then the next sentence is also extracted as an answer candidate, similar as (Chen et al., 2006).

Answer Ranking: The answer candidates are ranked by our proposed Opinion PageRank method or Opinion HITS method.

Answer Selection by Removing Redundancy: We incrementally add the top ranked sentence into the answer set, if its cosine similarity with every extracted answer doesn't exceed a predefined threshold, until the number of selected sentence (here is 40) is reached.

5 Experiments

5.1 Experiment Step

5.1.1 Dataset

We employ the dataset from the TAC 2008 QA track. The task contains a total of 87 squishy

opinion questions.[1] These questions have simple forms, and can be easily divided into positive type or negative type, for example "Why do people like Mythbusters?" and "What were the specific actions or reasons given for a negative attitude towards Mahmoud Ahmadinejad?". The initial topic word for each question (called target in TAC) is also provided. Since our work in this paper focuses on sentence ranking for opinion QA, these characteristics of TAC data make it easy to process question analysis. Answers for all questions must be retrieved from the TREC Blog06 collection (Craig Macdonald and Iadh Ounis, 2006). The collection is a large sample of the blog sphere, crawled over an eleven-week period from December 6, 2005 until February 21, 2006. We retrieve the top 50 documents for each question.

5.1.2 Evaluation Metrics

We adopt the evaluation metrics used in the TAC squishy opinion QA task (Dang, 2008). The TAC assessors create a list of acceptable information nuggets for each question. Each nugget will be assigned a normalized weight based on the number of assessors who judged it to be vital. We use these nuggets and corresponding weights to assess our approach. Three human assessors complete the evaluation process. Every question is scored using nugget recall (NR) and an approximation to nugget precision (NP) based on length. The final score will be calculated using F measure with TAC official value $\beta = 3$ (Dang, 2008). This means recall is 3 times as important as precision:

$$F(\beta = 3) = \frac{(3^2 + 1) \cdot NP \cdot NR}{3^2 \cdot NP + NR}$$

where NP is the sum of weights of nuggets returned in response over the total sum of weights of all nuggets in nugget list, and $NP = 1 - (\text{length} - \text{allowance})/(\text{length})$ if length is no less than allowance and 0 otherwise. Here $\text{allowance} = 100 \times (\sharp\text{nuggets returned})$ and length equals to the number of non-white characters in strings. We will use average F Score to evaluate the performance for each system.

5.1.3 Baseline

The baseline combines the topic score and opinion score with a linear weight for each answer candidate, similar to the previous ad-hoc algorithms:

$$\text{final\_score} = (1 - \alpha) \times \text{opinion\_score} + \alpha \times \text{topic\_score} \quad (8)$$

The topic score is computed by the cosine similarity between question topic words and answer candidate. The opinion score is calculated using the number of opinion words normalized by the total number of words in candidate sentence.

5.2 Performance Evaluation

5.2.1 Performance on Sentimental Lexicons

	Lexicon Name	Neg Size	Pos Size	Description
1	HowNet	2700	2009	English translation of positive/negative Chinese words
2	Senti-WordNet	4800	2290	Words with a positive or negative score above 0.6
3	Intersec-tion	640	518	Words appeared in both 1 and 2
4	Union	6860	3781	Words appeared in 1 or 2
5	All	10228	10228	All words appeared in 1 or 2 without distinguishing pos or neg

Table 1: Sentiment lexicon description

For lexicon-based opinion analysis, the selection of opinion thesaurus plays an important role in the final performance. HowNet[2] is a knowledge database of the Chinese language, and provides an online word list with tags of positive and negative polarity. We use the English translation of those sentiment words as the sentimental lexicon. SentiWordNet (Esuli and Sebastiani, 2006) is another popular lexical resource for opinion mining. Table 1 shows the detail information of our used sentiment lexicons. In our models, the positive opinion words are used only for positive questions, and negative opinion words just for negative questions. We initially set parameter λ in Opinion PageRank as 0 as (Liu and Ma, 2005), and other parameters simply as 0.5, including μ in Opinion PageRank, γ in Opinion HITS, and α in baseline. The experiment results are shown in Figure 4.

We can make three conclusions from Figure 4: 1. Opinion PageRank and Opinion HITS are both effective. The best results of Opinion PageRank and Opinion HITS respectively achieve around 35.4% (0.199 vs 0.145), and 34.7% (0.195 vs 0.145) improvements in terms of F score over the best baseline result. We believe this is because our proposed models not only incorporate the topic information and opinion information, but also con-

[1] 3 questions were dropped from the evaluation due to no correct answers found in the corpus

[2] http://www.keenage.com/zhiwang/e_zhiwang.html

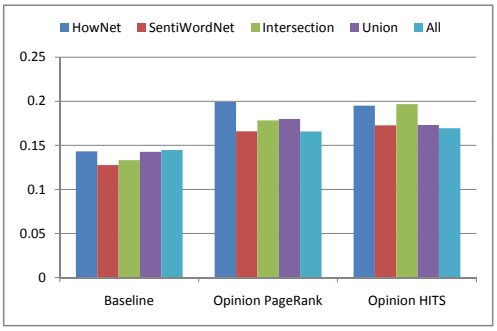

Figure 4: Sentiment Lexicon Performance

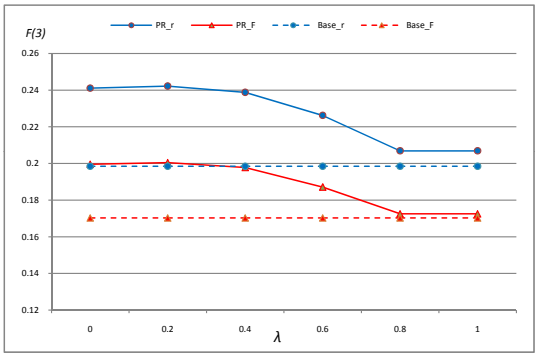

Figure 5: Opinion PageRank Performance with varying parameter λ ($\mu = 0.5$)

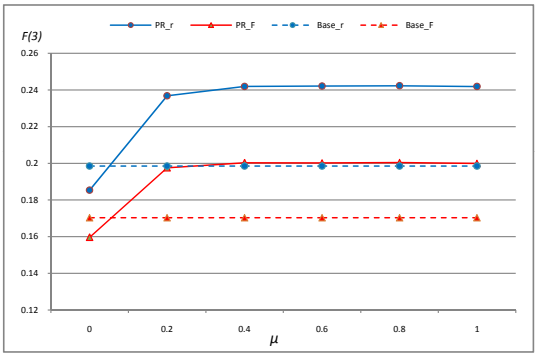

Figure 6: Opinion PageRank Performance with varying parameter μ ($\lambda = 0.2$)

sider the relationship between different answers. The experiment results demonstrate the effectiveness of these relations. 2. Opinion PageRank and Opinion HITS are comparable. Among five sentimental lexicons, Opinion PageRank achieves the best results when using HowNet and Union lexicons, and Opinion HITS achieves the best results using the other three lexicons. This may be because when the sentiment lexicon is defined appropriately for the specific question set, the opinion PageRank model performs better. While when the sentiment lexicon is not suitable for these questions, the opinion HITS model may dynamically learn a temporal sentiment lexicon and can yield a satisfied performance. 3. Hownet achieves the best overall performance among five sentiment lexicons. In HowNet, English translations of the Chinese sentiment words are annotated by non-native speakers; hence most of them are common and popular terms, which maybe more suitable for the Blog environment (Zhang and Ye, 2008). We will use HowNet as the sentiment thesaurus in the following experiments.

In baseline, the parameter α shows the relative contributions for topic score and opinion score. We vary α from 0 to 1 with an interval of 0.1, and find that the best baseline result 0.170 is achieved when α=0.1. This is because the topic information has been considered during candidate extraction, the system considering more opinion information (lower α) achieves better. We will use this best result as baseline score in following experiments. Since F(3) score is more related with recall, F score and recall will be demonstrated. In the next two sections, we will present the performances of the parameters in each model. For simplicity, we denote Opinion PageRank as PR, Opinion HITS as HITS, baseline as Base, Recall as r, F score as F.

5.2.2 Opinion PageRank Performance

In Opinion PageRank model, the value λ combines the source opinion and the destination opinion. Figure 5 shows the experiment results on parameter λ. When we consider lower λ, the system performs better. This demonstrates that the destination opinion score contributes more than source opinion score in this task.

The value of μ is a trade-off between answer reinforcement relation and topic relation to calculate the scores of each node. For lower value of μ, we give more importance to the relevance to the question than the similarity with other sentences. The experiment results are shown in Figure 6. The best result is achieved when $\mu = 0.8$. This figure also shows the importance of reinforcement between answer candidates. If we don't consider the sentence similarity($\mu = 0$), the performance drops significantly.

5.2.3 Opinion HITS Performance

The parameter γ combines the opinion hub score and topic hub score in the Opinion HITS model. The higher γ is, the more contribution is given

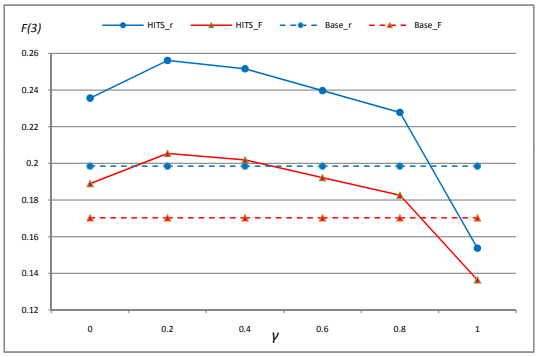

Figure 7: Opinion HITS Performance with varying parameter γ

to topic hub level, while the less contribution is given to opinion hub level. The experiment results are shown in Figure 7. Similar to baseline parameter α, since the answer candidates are extracted based on topic information, the systems considering opinion information heavily (α=0.1 in baseline, γ=0.2) perform best.

Opinion HITS model ranks the sentences by authority scores. It can also rank the popular opinion words and popular topic words from the topic hub layer and opinion hub layer, towards a specific question. Take the question 1024.3 "What reasons do people give for liking Zillow?" as an example, its topic word is "Zillow", and its sentiment polarity is positive. Based on the final hub scores, the top 10 topic words and opinion words are shown as Table 2.

Opinion Words	real, like, accurate, rich, right, interesting, better, easily, free, good
Topic Words	zillow, estate, home, house, data, value, site, information, market, worth

Table 2: Question-specific popular topic words and opinion words generated by Opinion HITS

Zillow is a real estate site for users to see the value of houses or homes. People like it because it is easily used, accurate and sometimes free. From the Table 2, we can see that the top topic words are the most related with question topic, and the top opinion words are question-specific sentiment words, such as "accurate", "easily", "free", not just general opinion words, like "great", "excellent" and "good".

5.2.4 Comparisons with TAC Systems

We are also interested in the performance comparison with the systems in TAC QA 2008. From Table 3, we can see Opinion PageRank and Opinion HITS respectively achieve around 10% improvement compared with the best result in TAC 2008, which demonstrates that our algorithm is indeed performing much better than the state-of-the-art opinion QA methods.

System	Precision	Recall	F(3)
OpPageRank	0.109	0.242	0.200
OpHITS	0.102	0.256	**0.205**
System 1	0.079	0.235	0.186
System 2	0.053	0.262	0.173
System 3	0.109	0.216	0.172

Table 3: Comparison results with TAC 2008 Three Top Ranked Systems (system 1-3 demonstrate top 3 systems in TAC)

6 Conclusion and Future Works

In this paper, we proposed two graph based sentence ranking methods for opinion question answering. Our models, called Opinion PageRank and Opinion HITS, could naturally incorporate topic relevance information and the opinion sentiment information. Furthermore, the relationships between different answer candidates can be considered. We demonstrate the usefulness of these relations through our experiments. The experiment results also show that our proposed methods outperform TAC 2008 QA Task top ranked systems by about 10% in terms of F score.

Our random walk based graph methods integrate topic information and sentiment information in a unified framework. They are not limited to the sentence ranking for opinion question answering. They can be used in general opinion document search. Moreover, these models can be more generalized to the ranking task with two types of influencing factors.

Acknowledgments: Special thanks to Derek Hao Hu and Qiang Yang for their valuable comments and great help on paper preparation. We also thank Hongning Wang, Min Zhang, Xiaojun Wan and the anonymous reviewers for their useful comments, and thank Hoa Trang Dang for providing the TAC evaluation results. The work was supported by 973 project in China(2007CB311003), NSFC project(60803075), Microsoft joint project "Opinion Summarization toward Opinion Search", and a grant from the International Development Research Center, Canada.

References

Ricardo Baeza-Yates and Berthier Ribeiro-Neto. 1999. *Modern Information Retrieval*. Addison Wesley, May.

Xavier Carreras and Lluis Marquez. 2005. Introduction to the conll-2005 shared task: Semantic role labeling.

Yi Chen, Ming Zhou, and Shilong Wang. 2006. Reranking answers for definitional qa using language modeling. In *ACL-CoLing*, pages 1081–1088.

Hang Cui, Min-Yen Kan, and Tat-Seng Chua. 2007. Soft pattern matching models for definitional question answering. *ACM Trans. Inf. Syst.*, 25(2):8.

Hoa Trang Dang. 2008. Overview of the tac 2008 opinion question answering and summarization tasks (draft). In *TAC*.

Günes Erkan and Dragomir R. Radev. 2004. Lexpagerank: Prestige in multi-document text summarization. In *EMNLP*.

Andrea Esuli and Fabrizio Sebastiani. 2006. Sentiwordnet: A publicly available lexical resource for opinion mining. In *LREC*.

Jon M. Kleinberg. 1999. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632.

Lun-Wei Ku, Yu-Ting Liang, and Hsin-Hsi Chen. 2007. Question analysis and answer passage retrieval for opinion question answering systems. In *ROCLING*.

Tie-Yan Liu and Wei-Ying Ma. 2005. Webpage importance analysis using conditional markov random walk. In *Web Intelligence*, pages 515–521.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into texts. In *EMNLP*.

Jahna Otterbacher, Günes Erkan, and Dragomir R. Radev. 2005. Using random walks for question-focused sentence retrieval. In *HLT/EMNLP*.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1998. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford University.

Swapna Somasundaran, Theresa Wilson, Janyce Wiebe, and Veselin Stoyanov. 2007. Qa with attitude: Exploiting opinion type analysis for improving question answering in online discussions and the news. In *ICWSM*.

Kim Soo-Min and Eduard Hovy. 2005. Identifying opinion holders for question answering in opinion texts. In *AAAI 2005 Workshop*.

Veselin Stoyanov, Claire Cardie, and Janyce Wiebe. 2005. Multi-perspective question answering using the opqa corpus. In *HLT/EMNLP*.

Vasudeva Varma, Prasad Pingali, Rahul Katragadda, and et al. 2008. Iiit hyderabad at tac 2008. In *Text Analysis Conference*.

X. Wan and J Yang. 2008. Multi-document summarization using cluster-based link analysis. In *SIGIR*, pages 299–306.

Hong Yu and Vasileios Hatzivassiloglou. 2003. Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In *EMNLP*.

Min Zhang and Xingyao Ye. 2008. A generation model to unify topic relevance and lexicon-based sentiment for opinion retrieval. In *SIGIR*, pages 411–418.