Simultaneous Ranking and Clustering of Sentences: A Reinforcement Approach to Multi-Document Summarization

[1]Xiaoyan Cai, [1]Wenjie Li, [1]You Ouyang, [2]Hong Yan

[1]Department of Computing, The Hong Kong Polytechnic University
{csxcai,cswjli,csyouyang}@comp.polyu.edu.hk

[2]Department of Logistics and Maritime Studies, The Hong Kong Polytechnic University
lgthyan@polyu.edu.hk

Abstract

Multi-document summarization aims to produce a concise summary that contains salient information from a set of source documents. In this field, sentence ranking has hitherto been the issue of most concern. Since documents often cover a number of topic themes with each theme represented by a cluster of highly related sentences, sentence clustering was recently explored in the literature in order to provide more informative summaries. Existing cluster-based ranking approaches applied clustering and ranking in isolation. As a result, the ranking performance will be inevitably influenced by the clustering result. In this paper, we propose a reinforcement approach that tightly integrates ranking and clustering by mutually and simultaneously updating each other so that the performance of both can be improved. Experimental results on the DUC datasets demonstrate its effectiveness and robustness.

1 Introduction

Automatic multi-document summarization has drawn increasing attention in the past with the rapid growth of the Internet and information explosion. It aims to condense the original text into its essential content and to assist in filtering and selection of necessary information. So far extractive summarization that directly extracts sentences from documents to compose summaries is still the mainstream in this field. Under this framework, sentence ranking is the issue of most concern.

Though traditional feature-based ranking approaches and graph-based approaches employed quite different techniques to rank sentences, they have at least one point in common, i.e., all of them focused on sentences only, but ignored the information beyond the sentence level (referring to Figure 1(a)). Actually, in a given document set, there usually exist a number of themes (or topics) with each theme represented by a cluster of highly related sentences (Harabagiu and Lacatusu, 2005; Hardy et al., 2002). These theme clusters are of different size and especially different importance to assist users in understanding the content in the whole document set. The cluster level information is supposed to have foreseeable influence on sentence ranking.

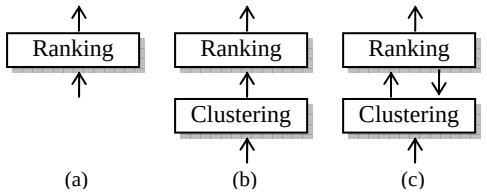

Figure 1. Ranking vs. Clustering

In order to enhance the performance of summarization, recently cluster-based ranking approaches were explored in the literature (Wan and Yang, 2006; Sun et al, 2007; Wang et al, 2008a,b; Qazvinian and Radev, 2008). Normally these approaches applied a clustering algorithm to obtain the theme clusters first and then ranked the sentences within each cluster or by exploring the interaction between sentences and obtained clusters (referring to Figure 1(b)). In other words, clustering and ranking are regarded as two independent processes in these approaches although the cluster-level information has been incorporated into the sentence ranking process. As a result,

Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010), pages 134–142,
Beijing, August 2010

the ranking performance is inevitably influenced by the clustering result.

To help alleviate this problem, we argue in this paper that the quality of ranking and clustering can be both improved when the two processes are mutually enhanced (referring to Figure 1(c)). Based on it, we propose a reinforcement approach that updates ranking and clustering interactively and iteratively to multi-document summarization. The main contributions of the paper are three-fold: (1) Three different ranking functions are defined in a bi-type document graph constructed from the given document set, namely global, within-cluster and conditional rankings, respectively. (2) A reinforcement approach is proposed to tightly integrate ranking and clustering of sentences by exploring term rank distributions over the clusters. (3) Thorough experimental studies are conducted to verify the effectiveness and robustness of the proposed approach.

The rest of this paper is organized as follows. Section 2 reviews related work in cluster-based ranking. Section 3 defines ranking functions and explains reinforced ranking and clustering process and its application in multi-document summarization. Section 4 presents experiments and evaluations. Section 5 concludes the paper.

2 Related Work

Clustering has become an increasingly important topic with the explosion of information available via the Internet. It is an important tool in text mining and knowledge discovery. Its ability to automatically group similar textual objects together enables one to discover hidden similarity and key concepts, as well as to summarize a large amount of text into a small number of groups (Karypis et al., 2000).

To summarize a scientific paper, Qazvinian and Radev (2008) presented two sentence selection strategies based on the clusters which were generated by a hierarchical agglomeration algorithm applied in the citation summary network. One was called C-RR, which started with the largest cluster and extracted the first sentence from each cluster in the order they appeared until the summary length limit was reached. The other was called C-LexRank, which was similar to C-RR but adopted LexRank to rank the sentences within each cluster and chose the most salient one.

Meanwhile, Wan and Yang (2008) proposed two models to incorporate the cluster-level information into the process of sentence ranking for generic summarization. While the Cluster-based Conditional Markov Random Walk model (ClusterCMRW) incorporated the cluster-level information into the text graph and manipulated clusters and sentences equally, the Cluster-based HITS model (ClusterHITS) treated clusters and sentences as hubs and authorities in the HITS algorithm.

Besides, Wang et al. (2008) proposed a language model to simultaneously cluster and summarize documents. Nonnegative factorization was performed on the term-document matrix using the term-sentence matrix as the base so that the document-topic and sentence-topic matrices could be constructed, from which the document clusters and the corresponding summary sentences were generated simultaneously.

3 A Reinforcement Approach to Multi-document Summarization

3.1 Document Bi-type Graph

First of all, let's introduce the sentence-term bi-type graph model for a set of given documents *D*, based on which the algorithm of reinforced ranking and clustering is developed. Let $G =< V, E, W >$, where V is the set of vertices that consists of the sentence set $S = \{s_1, s_2, \ldots, s_n\}$ and the term set $T = \{t_1, t_2, \ldots, t_m\}$, i.e., $V = S \bigcup T$, E is the set of edges that connect the vertices, i.e., $E = \{< v_i, v_j > | v_i, v_j \in V\}$. W is the adjacency matrix in which the element w_{ij} represents the weight of the edge connecting v_i and v_j. Formally, W can be decomposed into four blocks, i.e., W_{SS}, W_{ST}, W_{TS} and W_{TT}, each representing a sub-graph of the textual objects indicated by the subscripts. W can be written as

$$W = \begin{pmatrix} W_{SS} & W_{ST} \\ W_{TS} & W_{TT} \end{pmatrix},$$

where $W_{ST}(i, j)$ is the number of times the term t_j appears in the sentence s_i. $W_{SS}(i,j)$ is

the number of common terms in the sentences s_i and s_j. W_{TS} is equal to W_{ST}^T as the relationships between terms and sentences are symmetric. For simplification, in this study we assume there is no direct relationships between terms, i.e., $W_{TT} = 0$. In the future, we will explore effective ways to integrate term semantic relationships into the model.

3.2 Basic Ranking Functions

Recall that our ultimate goal is sentence ranking. As an indispensable part of the approach, the basic ranking functions need to be defined first.

3.2.1 Global Ranking (*without Clustering*)

Let $r(s_i)$ (i=1, 2, …, n) and $r(t_j)$ (j=1, 2, …, m) denote the ranking scores of the sentence s_i and the term t_j in the whole document set, respectively. Based on the assumptions that

> "*Highly ranked terms appear in highly ranked sentences, while highly ranked sentences contain highly ranked terms. Moreover, a sentence is ranked higher if it contains many terms that appear in many other highly ranked sentences.*"

we define

$$r(s_i) = \lambda \cdot \sum_{j=1}^{m} W_{ST}(i, j) \cdot r(t_j) + (1-\lambda) \cdot \sum_{j=1}^{n} W_{SS}(i, j) \cdot r(s_j) \quad (1)$$

and

$$r(t_j) = \sum_{i=1}^{n} W_{TS}(j, i) \cdot r(s_i) . \quad (2)$$

For calculation purpose, $r(s_i)$ and $r(t_j)$ are normalized by

$$r(s_i) \leftarrow \frac{r(s_i)}{\sum_{i'=1}^{n} r(s_{i'})} \text{ and } r(t_j) \leftarrow \frac{r(t_j)}{\sum_{j'=1}^{m} r(t_{j'})} .$$

Equations (1) and (2) can be rewritten using the matrix form, i.e.,

$$\begin{cases} r(S) = \lambda \cdot \dfrac{W_{ST} \cdot r(T)}{\| W_{ST} \cdot r(T) \|} + (1-\lambda) \cdot \dfrac{W_{SS} \cdot r(S)}{\| W_{SS} \cdot r(S) \|} \\ r(T) = \dfrac{W_{TS} \cdot r(S)}{\| W_{TS} \cdot r(S) \|} \end{cases} . \quad (3)$$

We call $r(S)$ and $r(T)$ the "**global ranking functions**", because at this moment sentence clustering is not yet involved and all the sentences/terms in the whole document set are ranked together.

Theorem: The solution to $r(S)$ and $r(T)$ given by Equation (3) is the primary eigenvector of $\lambda \cdot W_{ST} \cdot W_{TS} + (1-\lambda) \cdot W_{SS}$ and $\lambda \cdot W_{TS} (I - (1-\lambda) \cdot W_{SS})^{-1} \cdot W_{ST}$, respectively.

Proof: Combine Equations (1) and (2), we get

$$r(S) = \lambda \cdot \frac{W_{ST} \cdot \dfrac{W_{TS} \cdot r(S)}{\| W_{TS} \cdot r(S) \|}}{\| W_{ST} \cdot \dfrac{W_{TS} \cdot r(S)}{\| W_{TS} \cdot r(S) \|} \|} + (1-\lambda) \cdot \frac{W_{SS} \cdot r(S)}{\| W_{SS} \cdot r(S) \|}$$

$$= \lambda \cdot \frac{W_{ST} \cdot W_{TS} \cdot r(S)}{\| W_{ST} \cdot W_{TS} \cdot r(S) \|} + (1-\lambda) \cdot \frac{W_{SS} \cdot r(S)}{\| W_{SS} \cdot r(S) \|}$$

As the iterative process is a power method, it is guaranteed that $r(S)$ converges to the primary eigenvector of $\lambda \cdot W_{ST} \cdot W_{TS} + (1-\lambda) \cdot W_{SS}$. Similarly, $r(T)$ is guaranteed to converge to the primary eigenvector of $\lambda \cdot W_{TS} (I - (1-\lambda) \cdot W_{SS})^{-1} \cdot W_{ST}$. ■

3.2.2 Local Ranking (*within Clusters*)

Assume now K theme clusters have been generated by certain clustering algorithm, denoted as $C = \{C_1, C_2, \ldots, C_K\}$ where C_k (k=1, 2, …, K) represents a cluster of highly related sentences S_{C_k} ($\in C_k$) which contain the terms T_{C_k} ($\in C_k$). The sentences and terms within the cluster C_k form a cluster bi-type graph with the adjacency matrix W_{C_k}. Let $r_{C_k}(S_{C_k})$ and $r_{C_k}(T_{C_k})$ denote the ranking scores of S_{C_k} and T_{C_k} within C_k. They are calculated by an equation similar to Equation (3) by replacing the document level adjacency matrix W with the cluster level adjacency matrix W_{C_k}. We call $r_{C_k}(S_{C_k})$ and $r_{C_k}(T_{C_k})$ the "**within-cluster ranking functions**" with respect to the cluster C_k. They are the local ranking functions, in contrast to $r(S)$ and $r(T)$ that rank all the sentences and terms in the whole document set D. We believe that it will benefit sentence overall ranking when knowing more details about the ranking results at the finer granularity of theme clusters, instead of at the coarse granularity of the whole document set.

3.2.3 Conditional Ranking (*across Clusters*)

To facilitate the discovery of rank distributions of terms and sentences over all the theme clusters, we further define two "**conditional ranking functions**" $r(S \mid C_k)$ and $r(T \mid C_k)$. These rank distributions are necessary for the parameter estimation during the reinforcement process introduced later. The conditional ranking score of the term t_j on the cluster C_k, i.e., $r(T \mid C_k)$ is directly derived from T_{C_k}, i.e., $r(t_j \mid C_k) = r_{C_k}(t_j)$ if $t_j \in C_k$, and $r(t_j \mid C_k) = 0$ otherwise. It is further normalized as

$$r(t_j \mid C_k) = \frac{r(t_j \mid C_k)}{\sum_{j=1}^{m} r(t_j \mid C_k)}. \quad (4)$$

Then the conditional ranking score of the sentence s_i on the cluster C_k is deduced from the terms that are included in s_i, i.e.,

$$r(s_i \mid C_k) = \frac{\sum_{j=1}^{m} W_{ST}(i,j) \cdot r(t_j \mid C_k)}{\sum_{i=1}^{n}\sum_{j=1}^{m} W_{ST}(i,j) \cdot r(t_j \mid C_k)}. \quad (5)$$

Equation (5) can be interpreted as that the conditional rank of s_i on C_k is higher if many terms in s_i are ranked higher in C_k. Now we have sentence and term conditional ranks over all the theme clusters and are ready to introduce the reinforcement process.

3.3 Reinforcement between Within-Cluster Ranking and Clustering

The conditional ranks of the term t_j across the K theme clusters can be viewed as a rank distribution. Then the rank distribution of the sentence s_i can be considered as a mixture model over K conditional rank distributions of the terms contained in the sentence s_i. And the sentence s_i can be represented as a K-dimensional vector in the new measure space, in which the vectors can be used to guide the sentence clustering update. Next, we will explain the mixture model of sentence and use EM algorithm (Bilmes, 1997) to get the component coefficients of the model. Then, we will present the similarity measure between sentence and cluster, which is used to adjust the clusters that the sentences belong to and in turn modify within-cluster ranking for the sentences in the updated clusters.

3.3.1 Sentence Mixture Model

For each sentence s_i, we assume that it follows the distribution $r(T \mid s_i)$ to generate the relationship between the sentence s_i and the term set T. This distribution can be considered as a mixture model over K component distributions, i.e. the term conditional rank distributions across K theme clusters. We use $\gamma_{i,k}$ to denote the probability that s_i belongs to C_k, then $r(T \mid s_i)$ can be modeled as:

$$r(T \mid s_i) = \sum_{k=1}^{K} \gamma_{i,k} \cdot r(T \mid C_k) \text{ and } \sum_{k=1}^{K} \gamma_{i,k} = 1. \quad (6)$$

$\gamma_{i,k}$ can be explained as $p(C_k \mid s_i)$ and calculated by the Bayesian equation $p(C_k \mid s_i) \propto p(s_i \mid C_k) \cdot p(C_k)$, where $p(s_i \mid C_k)$ is assumed to be $r(s_i \mid C_k)$ obtained from the conditional rank of s_i on C_k as introduced before and $p(C_k)$ is the prior probability.

3.3.2 Parameter Estimation

We use EM algorithm to estimate the component coefficients $\gamma_{i,k}$ along with $\{p(C_k)\}$. A hidden variable C_z, $z \in \{1,2,\ldots,K\}$ is used to denote the cluster label that a sentence term pair (s_i, t_j) are from. In addition, we make the independent assumption that the probability of s_i belonging to C_k and the probability of t_j belonging to C_k are independent, i.e., $p(s_i, t_j \mid C_k) = p(s_i \mid C_k) \cdot p(t_j \mid C_k)$, where $p(s_i, t_j \mid C_k)$ is the probability of s_i and t_j both belonging to C_k. Similarly, $p(t_j \mid C_k)$ is assumed to be $r(t_j \mid C_k)$.

Let Θ be the parameter matrix, which is a $n \times K$ matrix $\Theta_{n \times K} = \{\gamma_{i,k}\}$ $(i = 1,\ldots,n;\ k = 1,\ldots,K)$. The best Θ is estimated from the relationships observed in the document bi-type graph, i.e., W_{ST} and W_{SS}. The likelihood of generating all the relationships under the parameter Θ can be calculated as:

$$L'(\Theta \mid W_{ST}, W_{SS}) = p(W_{ST} \mid \Theta) \cdot p(W_{SS} \mid \Theta)$$
$$= \prod_{i=1}^{n}\prod_{j=1}^{m} p(s_i, t_j \mid \Theta)^{W_{ST}(i,j)} \cdot \prod_{i=1}^{n}\prod_{j=1}^{n} p(s_i, s_j \mid \Theta)^{W_{SS}(i,j)}$$

where $p(s_i, t_j \mid \Theta)$ is the probability that s_i and t_j both belong to the same cluster, given the current parameter. As $p(s_i, s_j \mid \Theta)$ does not contain variables from Θ, we only need to consider maximizing the first part of the likelihood in order to get the best estimation of Θ. Let $L(\Theta \mid W_{ST})$ be the first part of likelihood.

Taking into account the hidden variable C_z, the complete log-likelihood can be written as

$$\log L(\Theta \mid W_{ST}, C_Z) = \log \prod_{i=1}^{n}\prod_{j=1}^{m}\left(p(s_i, t_j, C_z \mid \Theta)\right)^{W_{ST}(i,j)}$$
$$= \log \prod_{i=1}^{n}\prod_{j=1}^{m}\left(p(s_i, t_j \mid C_z, \Theta)\cdot p(C_z \mid \Theta)\right)^{W_{ST}(i,j)} .$$
$$= \sum_{i=1}^{n}\sum_{j=1}^{m} W_{ST}(i,j)\cdot \log\left(p_Z(s_i, t_j)\cdot p(C_z \mid \Theta)\right)$$

In the E-step, given the initial parameter Θ^0, which is set to $\gamma_{i,k}^0 = 1/K$ for all *i* and *k*, the expectation of log-likelihood under the current distribution of C_Z is:

$$Q(\Theta, \Theta^0) = E_{f(C_Z \mid W_{ST}, \Theta^0)}(\log L(\Theta \mid W_{ST}, C_Z)$$
$$= \sum_{k=1}^{K}\sum_{i=1}^{n}\sum_{j=1}^{m} W_{ST}(i,j)\cdot \log(p_k(s_i, t_j))\cdot p(C_z = C_k \mid s_i, t_j, \Theta^0) +$$
$$\sum_{i=1}^{n}\sum_{k=1}^{K}\sum_{j=1}^{m} W_{ST}(i,j)\cdot \log(p(C_z = C_k \mid \Theta))\cdot p(C_z = C_k \mid s_i, t_j, \Theta^0)$$

The conditional distribution in the above equation, i.e., $p(C_z = C_k \mid s_i, t_j, \Theta^0)$, can be calculated using the Bayesian rule as follows:

$$p(C_z = C_k \mid s_i, t_j, \Theta^0)$$
$$\propto p(s_i, t_j \mid C_z = C_k, \Theta^0) p(C_z = C_k \mid \Theta^0). \quad (7)$$
$$\propto p^0(s_i \mid C_k) p^0(t_j \mid C_k) p^0(C_z = C_k)$$

In the M-Step, we first get the estimation of $p(C_z = C_k)$ by maximizing the expectation $Q(\Theta, \Theta^0)$. By introducing a Lagrange multiplier λ, we get the equation below.

$$\frac{\partial}{\partial p(C_z = C_k)}[Q(\Theta, \Theta^0) + \lambda(\sum_{k=1}^{K} p(C_z = C_k) - 1)] = 0 \Rightarrow$$
$$\sum_{i=1}^{n}\sum_{j=1}^{m} W_{ST}(i,j)\frac{1}{p(C_z = C_k)} p(C_z = C_k \mid s_i, t_j, \Theta^0) + \lambda = 0$$

Thus, the estimation of $p(C_z = C_k)$ given previous Θ^0 is

$$p(C_z = C_k) = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m} W_{ST}(i,j) p(C_z = C_k \mid s_i, t_j, \Theta^0)}{\sum_{i=1}^{n}\sum_{j=1}^{m} W_{ST}(i,j)} . \quad (8)$$

Then, the parameters $\gamma_{i,k}$ can be calculated with the Bayesian rule as

$$\gamma_{i,k} = \frac{p(s_i \mid C_k) p(C_z = C_k)}{\sum_{l=1}^{K} p(s_i \mid C_l) p(C_z = C_l)} . \quad (9)$$

By setting $\Theta^0 = \Theta$, the whole process can be repeated. The updating rules provided in Equations (7)-(9) are applied at each iteration. Finally Θ will converge to a local maximum. A similar estimation process has been adopted in (Sun et al., 2009), which was used to estimate the component coefficients for author-conference networks.

3.3.3 Similarity Measure

After we get the estimations of the component coefficients $\gamma_{i,k}$ for s_i, s_i will be represented as a *K* dimensional vector $\vec{s}_i = (\gamma_{i,1}, \gamma_{i,2}, \ldots, \gamma_{i,K})$. The center of each cluster can thus be calculated accordingly, which is the mean of $\vec{s}_i$ for all s_i in the same cluster, i.e.,

$$\overrightarrow{Center}_{C_k} = \frac{\sum_{s_i \in C_k} \vec{s}_i}{|C_k|},$$

where $|C_k|$ is the size of C_k.

Then the similarity between each sentence and each cluster can be calculated as the cosine similarity between them, i.e.,

$$sim(s_i, C_k) = \frac{\sum_{l=1}^{K} \vec{s}_i(l)\overrightarrow{Center}_{C_k}(l)}{\sqrt{\sum_{l=1}^{K} \vec{s}_i(l))^2}\sqrt{\sum_{l=1}^{K} \overrightarrow{Center}_{C_k}(l))^2}} . \quad (10)$$

Finally, each sentence is re-assigned to a cluster that is the most similar to the sentence. Based on the updated clusters, within-cluster ranking is updated accordingly, which triggers the next round of clustering refinement. It is expected that the quality of clusters should be improved during this iterative update process since the similar sentences under new attributes will be grouped together, and meanwhile the quality of ranking will be improved along with the better clusters and

thus offers better attributes for further clustering.

3.4 Ensemble Ranking

The overall sentence ranking function f is defined as the ensemble of all the sentence conditional ranking scores on the K clusters.

$$f(s_i) = \sum_{k=1}^{K} \alpha_k \cdot r(s_i \mid C_k), \qquad (11)$$

where α_k is a coefficient evaluating the importance of C_k. It can be formulated as the normalized cosine similarity between a theme cluster and the whole document set for generic summarization, or between a theme cluster and a given query for query-based summarization. $\alpha_k \in [0,1]$ and $\sum_{k=1}^{K} \alpha_k = 1$.

Figure 2 below summarizes the whole process that determines the overall sentence ensemble ranking scores.

Input: The bi-type document graph $G =< S \bigcup T, E, W >$, ranking functions, the cluster number K, $\varepsilon = 1$, $Tre = 0.001$, $IterNum = 10$.

Output: sentence final ensemble ranking vector $f(S)$.

1. $t \leftarrow 0$;
2. Get the initial partition for S, i.e. C_k^t, $k = 1,2,\ldots K$, calculate cluster centers $\overrightarrow{Center}_{C_k^t}$ accordingly.
3. **For** (t=1; t<IterNum && $\varepsilon > Tre$; t++)
4. Calculate the within-cluster ranking $r_{C_k}(T_{C_k})$, $r_{C_k}(S_{C_k})$ and the conditional ranking $r(s_i \mid C_k)$;
5. Get new attribute $\vec{s}_i$ for each sentence s_i, and new attribute $\overrightarrow{Center}_{C_k^t}$ for each cluster C_k^t;
6. **For** each sentence s_i in S
7. **For** k=1 to K
8. Calculate similarity value $sim(s_i, C_k^t)$
9. **End For**
10. Assign s_i to $C_{k_0}^{t+1}$, $k_0 = \arg\max_k sim(s_i, C_k^t)$
11. **End For**
12. $\varepsilon = \max_k | \overrightarrow{Center}_{C_k^{t+1}} - \overrightarrow{Center}_{C_k^t} |$
13. $t \leftarrow t+1$
14. **End For**
15. For each sentence s_i in S
16. **For** k=1 to K
17. $f(s_i) = \sum_{k=1}^{K} \alpha_k \cdot r(s_i \mid C_k)$
18. **End For**
19. **End For**

Figure 2. The Overall Sentence Ranking Algorithm

3.5 Summary Generation

In multi-document summarization, the number of documents to be summarized can be very large. This makes information redundancy appears to be more serious in multi-document summarization than in single-document summarization. Redundancy control is necessary. We apply a simple yet effective way to choose summary sentences. Each time, we compare the current candidate sentence to the sentences already included in the summary. Only the sentence that is not too similar to any sentence in the summary (i.e., the cosine similarity between them is lower than a threshold) is selected into the summary. The iteration is repeated until the length of the sentences in the summary reaches the length limitation. In this paper, the threshold is set to 0.7 as always in our past work.

4 Experiments and Evaluations

We conduct the experiments on the DUC 2004 generic multi-document summarization dataset and the DUC 2006 query-based multi-document summarization dataset. According to task definitions, systems are required to produce a concise summary for each document set (without or with a given query description) and the length of summaries is limited to 665 bytes in DUC 2004 and 250 words in DUC 2006.

A well-recognized automatic evaluation toolkit ROUGE (Lin and Hovy, 2003) is used in evaluation. It measures summary quality by counting overlapping units between system-generated summaries and human-written reference summaries. We report two common ROUGE scores in this paper, namely ROUGE-1 and ROUGE-2, which base on Uni-gram match and Bi-gram match, respectively. Documents and queries are pre-processed by segmenting sentences and splitting words. Stop words are removed and the remaining words are stemmed using Porter stemmer.

4.1 Evaluation of Performance

In order to evaluate the performance of reinforced clustering and ranking approach, we compare it with the other three ranking approaches: (1) Global-Rank, which does not apply clustering and simply relies on the

sentence global ranking scores to select summary sentences; (2) Local-Rank, which clusters sentences first and then rank sentences within each cluster. A summary is generated in the same way as presented in (Qazvinian and Radev, 2008). The clusters are ordered by decreasing size; (3) Cluster-HITS, which also clusters sentences first, but then regards clusters as hubs and sentences as authorities in the HITS algorithm and uses the obtained authority scores to rank and select sentences. The classical clustering algorithm K-means is used where necessary. For query-based summarization, the additional query-relevance (i.e. the cosine similarity between sentences and query) is involved to re-rank the candidate sentences chosen by the ranking approaches for generic summarization.

Note that K-means requires a predefined cluster number *K*. To avoid exhaustive search for a proper cluster number for each document set, we employ the spectra approach introduced in (Li et al., 2007) to predict the number of the expected clusters. Based on the sentence similarity matrix using the normalized 1-norm, for its eigenvalues λ_i (i=1,2, …, n), the ratio $\alpha_i = \lambda_{i+1} / \lambda_2 (\lambda \geq 1)$ is defined. If $\alpha_i - \alpha_{i+1} > 0.05$ and α_i is still close to 1, then set K=i+1. Tables 1 and 2 below compare the performance of the four approaches on DUC 2004 and 2006 according to the calculated *K*.

DUC 2004	ROUGE-1	ROUGE-2
Reinforced	0.37082	0.08351
Cluster-HITS	0.36463	0.07632
Local-Rank	0.36294	0.07351
Global-Rank	0.35729	0.06893

Table 1. Results on the DUC 2004 dataset

DUC 2006	ROUGE-1	ROUGE-2
Reinforced	0.39531	0.08957
Cluster-HITS	0.38315	0.08632
Local-Rank	0.38104	0.08841
Global-Rank	0.37478	0.08531

Table 2. Results on the DUC 2006 dataset

It is not surprised to find that "Global-Rank" shows the poorest performance, when it utilizes the sentence level information only whereas the other three approaches all integrate the additional cluster level information in various ways. In addition, as results illustrate, the performance of "Cluster-HITS" is better than the performance of "Local-Rank". This can be mainly credited to the ability of "Cluster-HITS" to consider not only the cluster-level information, but also the sentence-to-cluster relationships, which are ignored in "Local-Rank". It is happy to see that the proposed reinforcement approach, which simultaneously updates clustering and ranking of sentences, consistently outperforms the other three approaches.

4.2 Analysis of Cluster Quality

Our original intention to propose the reinforcement approach is to hope to generate more accurate clusters and ranking results by mutually refining within-cluster ranking and clustering. In order to check and monitor the variation trend of the cluster quality during the iterations, we define the following measure

$$quan = \sum_{k=1}^{K} \left(\frac{\min_{s_i \in C_k} sim(s_i, C_k)}{\sum_{l=1, l \neq k}^{K} \min_{s_i \in C_k, s_j \in C_l} sim(s_i, s_j)} \right), \quad (12)$$

where $\min_{s_i \in C_k} sim(s_i, C_k)$ denotes the distance between the cluster center and the border sentence in a cluster that is the farthest away from the center. The larger it is, the more compact the cluster is. $\min_{s_i \in C_k, s_j \in C_l} sim(s_i, s_j)$, on the other hand, denotes the distance between the most distant pair of sentences, one from each cluster. The smaller it is, the more separated the two clusters are. The distance is measured by cosine similarity. As a whole, the larger *quan* means the better cluster quality. Figure 3 below plots the values of *quan* in each iteration on the DUC 2004 and 2006 datasets. Note that the algorithm converges in less than 6 rounds and 5 rounds on the DUC 2004 and 2006 datasets, respectively. The curves clearly show the increasment of *quan* and thus the improved cluster quality.

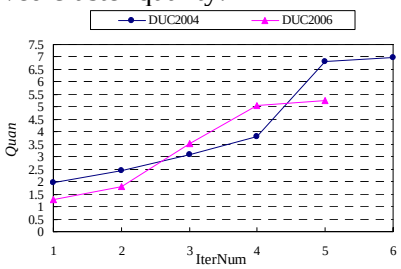

Figure 3. Cluster Quality on DUC 2004 and 2006

While *quan* directly evaluate the quality of the generated clusters, we are also quite interested in whether the improved clusters quality can further enhance the quality of sentence ranking and thus consequently raise the performance of summarization. Therefore, we evaluate the ROUGEs in each iteration as well. Figure 4 below illustrates the changes of ROUGE-1 and ROUGE-2 result on the DUC 2004 and 2006 datasets, respectively. Now, we have come to the positive conclusion.

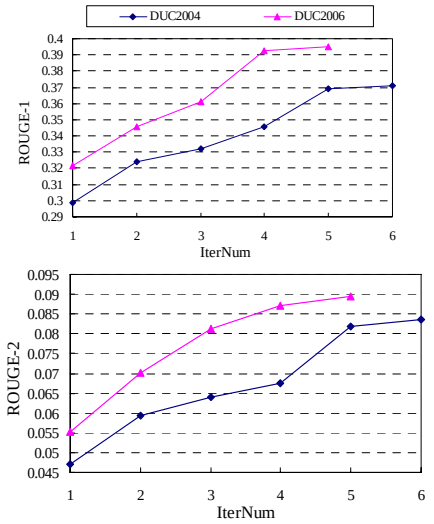

Figure 4. ROUGEs on DUC 2004 and 2006

4.3 Impact of Cluster Numbers

In previous experiments, the cluster number is predicted through the eigenvalues of 1-norm normalized sentence similarity matrix. This number is just the estimated number. The actual number is hard to predict accurately. To further examine how the cluster number influences summarization, we conduct the following additional experiments by varying the cluster number. Given a document set, we let S denote the sentence set in the document set, and set K in the following way:

$$K = \varepsilon \times |S|, \qquad (13)$$

where $\varepsilon \in (0,1)$ is a ratio controlling the expected cluster number. The larger ε is, the more clusters will be produced. ε ranges from 0.1 to 0.9 in the experiments. Due to page limitation, we only provide the ROUGE-1 and ROUGE-2 results of the proposed approach, "Cluster-HITS" and "Local-Rank" on the DUC 2004 dataset in Figure 5. The similar curves are also observed on the 2006 dataset.

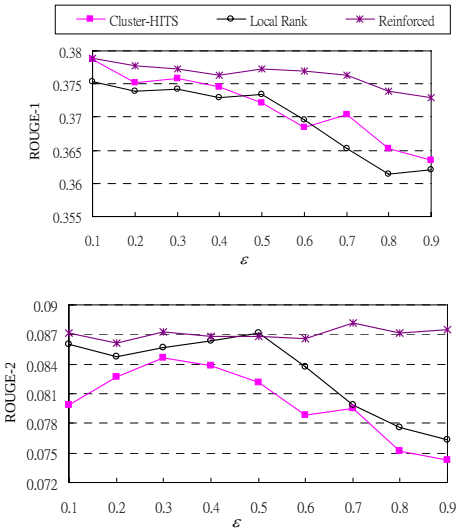

Figure 5. ROUGEs vs. ε on DUC 2004

It is shown that (1) the proposed approach outperforms "Cluster-HITS" and "Local-Rank" in almost all the cases no matter how the cluster number is set; (2) the performances of "Cluster-HITS" and "Local-Rank" are more sensitive to the cluster number and a large number of clusters appears to deteriorate the performances of both. This is reasonable. Actually when ε getting close to 1, "Local-Rank" approaches to "Global-Rank". These results demonstrate the robustness of the proposed approach.

5 Conclusion

In this paper, we present a reinforcement approach that tightly integrates ranking and clustering together by mutually and simultaneously updating each other. Experimental results demonstrate the effectiveness and the robustness of the proposed approach. In the future, we will explore how to integrate term semantic relationships to further improve the performance of summarization.

Acknowledgement

The work described in this paper was supported by an internal grant from the Hong Kong Polytechnic University (G-YG80).

References

J. Bilmes. 1997. *A Gentle Tutorial on the em Algorithm and Its Application to Parameter Wstimation for Gaussian Mixture and Hidden Markov Models*. Technical Report ICSI-TR-97-02, University of Berkeley.

Brin, S., and Page, L. 1998. *The Anatomy of a Large-scale Hypertextual Web Search Engine*. In Proceedings of WWW1998..

Harabagiu S. and Lacatusu F. 2005. *Topic Themes for Multi-Document Summarization*. In Proceedings of SIGIR2005.

Hardy H., Shimizu N., Strzalkowski T., Ting L., Wise G. B., and Zhang X. 2002. *Cross-Document Summarization by Concept Classification*. In Proceedings of SIGIR2002.

Jon M. Kleinberg. 1999. *Authoritative Sources in a Hyperlinked Environment*. In Proceedings of the 9th ACM-SIAM Symposium on Discrete Algorithms.

Karypis, George, Vipin Kumar and Michael Steinbach. 2000. *A Comparison of Document Clustering Techniques*. KDD workshop on Text Mining.

Lin, C. Y. and Hovy, E. 2000. *The Automated Acquisition of Topic Signature for Text Summarization*. In Proceedings of COLING2000.

Li W.Y., Ng W.K., Liu Y. and Ong K.L. 2007. *Enhancing the Effectiveness of Clustering with Spectra Analysis*. IEEE Transactions on Knowledge and Data Engineering (TKDE). 19(7): 887-902.

Li, F., Tang, Y., Huang, M., Zhu, X. 2009. *Answering Opinion Questions with Random Walks on Graphs*. In Proceedings of ACL2009.

Otterbacher J., Erkan G. and Radev D. 2005. *Using RandomWalks for Question-focused Sentence Retrieval*. In Proceedings of HLT/EMNLP 2005.

Qazvinian V. and Radev D. R. 2008. *Scientific paper summarization using citation summary networks*. In Proceedings of COLING2008.

Sun P., Lee J.H., Kim D.H., and Ahn C.M. 2007. *Multi-Document Using Weighted Similarity Between Topic and Clustering-Based Non-negative Semantic Feature*. APWeb/WAIM 2007.

Sun Y., Han J., Zhao P., Yin Z., Cheng H., and Wu T. 2009. *Rankclus: Integrating Clustering with Ranking for Heterogenous Information Network Analysis*. In Proceedings of EDBT 2009.

Wang D.D., Li T., Zhu S.H., Ding Chris. 2008a *Multi-Document Summarization via Sentence-Level Semantic Analysis and Symmetric Matrix Factorization*. In Proceedings of SIGIR2008.

Wang D.D., Zhu S.H., Li T., Chi Y., and Gong Y.H. 2008b. *Integrating Clustering and Multi-Document Summarization to Improve Document Understanding*. In Proceedings of CIKM 2008.

Wan X. and Yang J. 2006. *Improved Affinity Graph based Multi-Document Summarization*. In Proceedings of HLT-NAACL2006.

Zha H. 2002. *Generic Summarization and Key Phrase Extraction using Mutual Reinforcement Principle and Sentence Clustering*. In Proceedings of SIGIR2002.