Multi-Document Summarization Using Cluster-Based Link Analysis

Xiaojun Wan and Jianwu Yang
Institute of Computer Science and Technology, Peking University, Beijing 100871, China
{wanxiaojun, yangjianwu}@icst.pku.edu.cn

ABSTRACT

The Markov Random Walk model has been recently exploited for multi-document summarization by making use of the link relationships between sentences in the document set, under the assumption that all the sentences are indistinguishable from each other. However, a given document set usually covers a few topic themes with each theme represented by a cluster of sentences. The topic themes are usually not equally important and the sentences in an important theme cluster are deemed more salient than the sentences in a trivial theme cluster. This paper proposes the Cluster-based Conditional Markov Random Walk Model (ClusterCMRW) and the Cluster-based HITS Model (ClusterHITS) to fully leverage the cluster-level information. Experimental results on the DUC2001 and DUC2002 datasets demonstrate the good effectiveness of our proposed summarization models. The results also demonstrate that the ClusterCMRW model is more robust than the ClusterHITS model, with respect to different cluster numbers.

Categories and Subject Descriptors: H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing – *abstracting methods*; I.2.7 [**Artificial Intelligence**]: Natural Language Processing – *text analysis*

General Terms: Algorithms, Experimentation, Performance

Keywords: Multi-Document Summarization, Cluster-based Link Analysis, Conditional Markov Random Walk Model, HITS

1. INTRODUCTION

Multi-document summarization aims to produce a summary delivering the majority of information content from a set of documents about an explicit or implicit main topic. Multi-document summary can be used to concisely describe the information contained in a cluster of documents and facilitate the users to understand the document cluster. For example, a number of news services (e.g. Google News) have been developed to group news articles into news topics, and then produce a short summary for each news topic. The users can easily understand the topic they have interest in by taking a look at the short summary.

Automated multi-document summarization has drawn much attention in recent years. In the communities of natural language processing and information retrieval, a series of workshops and conferences on automatic text summarization (e.g. NTCIR, DUC), special topic sessions in ACL, COLING, and SIGIR have advanced the summarization techniques and produced a couple of experimental online systems.

A particular challenge for multi-document summarization is that a document set might contain diverse information, which is either related or unrelated to the main topic, and hence we need effective summarization methods to analyze the information stored in different documents and extract the globally important information to reflect the main topic. Another challenge for multi-document summarization is that the information stored in different documents inevitably overlaps with each other, and hence we need effective summarization methods to merge information stored in different documents, and if possible, contrast their differences. In recent years, both unsupervised and supervised methods have been proposed to analyze the information contained in a document set and extract highly salient sentences into the summary, based on syntactic or statistical features.

Most recently, the Markov Random Walk Model (abbr. MRW) has been successfully used for multi-document summarization by making use of the "voting" or "recommendations" between sentences in the documents [4, 21, 25]. The model first constructs a directed or undirected graph to reflect the relationships between the sentences and then applies the graph-based ranking algorithm to compute the rank scores for the sentences. The sentences with large rank scores are chosen into the summary. However, the model makes uniform use of the sentences in the document set, i.e. all the sentences are ranked without considering the higher-level information beyond the sentence-level information. Actually, given a document set, there usually exist a number of themes or subtopics, and each theme or subtopic is represented by a cluster of highly related sentences [6, 7]. The theme clusters are usually of different size and have different importance for users to understand the document set. For example, the theme clusters close to the main topic of the document set are usually more important than the theme clusters far away from the main topic of the document set. The cluster-level information is deemed to have great influence on the sentence ranking process. Moreover, the sentences in the same theme cluster cannot be treated uniformly. Some sentences in the cluster are more important than other sentences because of their different distances to the cluster's centroid. In brief, neither the cluster-level information nor the sentence-to-cluster relationship can be taken into account in the Markov Random Walk Model.

In order to address the above limitations of the Markov Random Walk Model, we propose two models to incorporate the cluster-level information into the process of sentence ranking. The first model is the Cluster-based Conditional Markov Random Walk Model (abbr. ClusterCMRW), which incorporates the cluster-level information into the link graph. The second model is the Cluster-based HITS Model (abbr. ClusterHITS), which considers the clusters and sentences as hubs and authorities in the

SIGIR'08, July 20–24, 2008, Singapore.
Copyright 2008 ACM 978-1-60558-164-4/08/07...$5.00.

HITS algorithm. Experiments have been performed on the DUC2001 and DUC2002 datasets, and the results demonstrate the good effectiveness of the two models. The experimental results also demonstrate that the ClusterCMRW model is more robust than the ClusterHITS model, with respect to different cluster numbers.

The rest of this paper is organized as follows: Section 2 introduces the related work. The basic Markov Random Walk Model is introduced in Section 3. And the two proposed models are presented in Sections 4. In Section 5, we describe the experiments and results. Lastly we conclude this paper in Section 6.

2. RELATED WORK

2.1 Multi-Document Summarization

A variety of multi-document summarization methods have been developed recently. Generally speaking, those methods can be either extractive summarization or abstractive summarization. Extractive summarization involves assigning saliency scores to some units (e.g. sentences, paragraphs) of the documents and extracting those with highest scores, while abstractive summarization (e.g. NewsBlaster) usually needs information fusion [2], sentence compression [10] and reformulation [20]. In this study, we focus on extractive summarization.

The centroid-based method [23] is one of the most popular extractive summarization methods. MEAD[1] is an implementation of the centroid-based method that scores sentences based on sentence-level and inter-sentence features, including cluster centroids, position, TFIDF, etc. NeATS [15] uses sentence position, term frequency, topic signature and term clustering to select important content, and use MMR [5] to remove redundancy. To further explore user interface issues, iNeATS [14] is developed based on NeATS. XDoX [7] is a cross document summarizer designed specifically to summarize large document sets by identifying the most salient themes within the set by passage clustering and then composes an extraction summary, which reflects these main themes. The passages are clustered based on n-gram matching. Much other work also explores to find topic themes in the documents for summarization, e.g. Harabagiu and Lacatusu [6] investigate five different topic representations and introduce a novel representation of topics based on topic themes. In addition, Marcu [19] selects important sentences based on the discourse structure of the text. TNO's system [11] scores sentences by combining a unigram language model approach with a Bayesian classifier based on surface features.

Most recently, the graph-based ranking methods have been proposed to rank sentences or passages based on the "votes" or "recommendations" between each other. Websumm [18] uses a graph-connectivity model and operates under the assumption that nodes which are connected to many other nodes are likely to carry salient information. LexPageRank [4] is an approach for computing sentence importance based on the concept of eigenvector centrality. It constructs a sentence connectivity matrix and computes sentence importance based on an algorithm similar to PageRank. Mihalcea and Tarau [21] also propose a similar algorithm based on PageRank [22] to compute sentence importance for single document summarization, and for multi-document summarization, they use a meta-summarization process to summarize the meta-document produced by assembling all the single summary of each document. Wan and Yang [25] improve the graph-ranking algorithm by differentiating intra-document links and inter-document links between sentences. All these methods make use of the relationships between sentences and select sentences according to the "votes" or "recommendations" from their neighboring sentences.

Other related work includes topic-focused document summarization [3], which aims to produce summary biased to a given topic or query.

2.2 Link Analysis

PageRank [22] and HITS [9] are two popular algorithms for link analysis between web pages and they have been successfully used to improve web retrieval. More advanced web link analysis methods have been proposed to leverage the multi-layer relationships between web pages. The Conditional Markov Random Walk Model has been successfully applied in the tasks of web page retrieval based on two-layer web graph [17]. Hierarchical structure of the web graph is also exploited for link analysis in [26]. In recent years, a few researches have focused on using link analysis methods to re-rank search results in order to improve the retrieval performance [12, 13, 27]. The links between documents are induced by computing the similarity between documents using the Cosine measure or language model measure. In addition, link analysis methods have also been applied in social network analysis [28] and other tasks.

3. THE BASIC MODEL

The Markov Random Walk Model (MRW) is essentially a way of deciding the importance of a vertex within a graph based on global information recursively drawn from the entire graph. The basic idea is that of "voting" or "recommendation" between the vertices. A link between two vertices is considered as a vote cast from one vertex to the other vertex. The score associated with a vertex is determined by the votes that are cast for it, and the score of the vertices casting these votes.

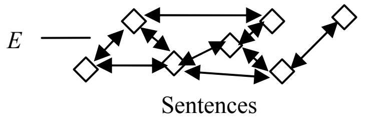

Figure 1: One-layer link graph

Formally, given a document set D, let G=(V, E) be a graph to reflect the relationships between sentences in the document set, as shown in Figure 1. V is the set of vertices and each vertex v_i in V is a sentence in the document set. E is the set of edges, which is a subset of $V\times V$. Each edge e_{ij} in E is associated with an affinity weight $f(i\rightarrow j)$ between sentences v_i and v_j ($i\neq j$). The weight is computed using the standard cosine measure [1] between the two sentences.

$$f(i \rightarrow j) = sim_{\cos ine}(v_i, v_j) = \frac{\vec{v}_i \cdot \vec{v}_j}{|\vec{v}_i| \times |\vec{v}_j|} \tag{1}$$

where $\vec{v}_i$ and $\vec{v}_j$ are the corresponding term vectors of v_i and v_j.

Two vertices are connected if their affinity weight is larger than 0 and we let $f(i\rightarrow i)$=0 to avoid self transition.

The transition probability from v_i to v_j is then defined by normalizing the corresponding affinity weight as follows.

[1] http://www.summarization.com/mead/

$$p(i \to j) = \begin{cases} \dfrac{f(i \to j)}{\sum_{k=1}^{|V|} f(i \to k)}, & \text{if } \sum f \neq 0 \\ 0 \ , & \text{otherwise} \end{cases} \qquad (2)$$

Note that $p(i \to j)$ is usually not equal to $p(j \to i)$. We use the row-normalized matrix $\widetilde{M} = (\widetilde{M}_{i,j})_{|V|\times|V|}$ to describe G with each entry corresponding to the transition probability.

$$\widetilde{M}_{i,j} = p(i \to j) \qquad (3)$$

In order to make $\widetilde{M}$ be a stochastic matrix, the rows with all zero elements are replaced by a smoothing vector with all elements set to $1/|V|$.

Based on the matrix $\widetilde{M}$, the saliency score $SenScore(v_i)$ for sentence v_i can be deduced from those of all other sentences linked with it and it can be formulated in a recursive form as in the PageRank algorithm.

$$SenScore(v_i) = \mu \cdot \sum_{all\, j \neq i} SenScore(v_j) \cdot \widetilde{M}_{j,i} + \frac{(1-\mu)}{|V|} \qquad (4)$$

And the matrix form is

$$\vec{\lambda} = \mu \widetilde{M}^T \vec{\lambda} + \frac{(1-\mu)}{|V|} \vec{e} \qquad (5)$$

where $\vec{\lambda} = [SenScore\ (v_i)]_{|V|\times 1}$ is the vector of saliency scores for the sentences. $\vec{e}$ is a column vector with all elements equaling to 1. μ is the damping factor usually set to 0.85, as in the PageRank algorithm.

The above process can be considered as a Markov chain by taking the sentences as the states and the final transition matrix is given by $A = \mu \widetilde{M}^T + \frac{(1-\mu)}{|V|} \vec{e}\vec{e}^T$, which is irreducible. The stationary probability distribution of each state is obtained by the principal eigenvector of the transition matrix.

For implementation, the initial scores of all sentences are set to 1 and the iteration algorithm in Equation (4) is adopted to compute the new scores of the sentences. Usually the convergence of the iteration algorithm is achieved when the difference between the scores computed at two successive iterations for any sentences falls below a given threshold (0.0001 in this study).

Note that after the saliency scores of sentences have been obtained, a variant of the MMR algorithm used in [27] is applied to penalize the sentences highly overlap with informative sentences and finally choose both informative and novel sentences into the summary.

4. THE PROPOSED MODELS

4.1 Overview

In the basic MRW model, all the sentences are indistinguishable from each other, i.e. the sentences are treated uniformly. However, as we mentioned in previous section, there may be many factors that can have influence on the importance analysis of the sentences. As shown in [6, 7], a document set usually contains a few topic themes and each theme can be represented by a cluster of topic-related sentences. The theme clusters are not equally important. Our assumption is that the sentences in an important theme cluster should be ranked higher than the sentences in other theme clusters, and an important sentence in a theme cluster should be ranked higher than other sentences in the cluster.

In order to leverage the cluster-level information, we propose two models to make use of the relationships between sentences and clusters. The first model is the Cluster-based Conditional Markov Random Walk Model (ClusterCMRW), which is an improvement of the MRW model or the PageRank algorithm [22] by incorporating the cluster-level information into the link graph. The second model is the Cluster-based HITS Model (ClusterHITS), which formalizes the sentence-cluster relationships as the authority-hub relationships in the HITS algorithm [9]. Both models are based on link analysis techniques.

Note that the above models are used to compute the saliency scores of the sentences in a document set, and other steps are needed to produce the final summary. The overall summarization framework consists of the following three steps:

1. **Theme cluster detection:** This step aims to detect theme clusters in the document set. In this study, we simply use the clustering algorithm to group sentences into a few theme clusters.
2. **Sentence score computation:** This step aims to compute the saliency scores of the sentences in the document set by using either the ClusterCMRW model or the ClusterHITS model to incorporate the cluster-level information.
3. **Summary extraction:** The same algorithm [27] as in the basic model is applied to remove redundancy and choose summary sentences.

The first two steps are key steps and the details will be described in next sections respectively. The last step is quite straightforward and we omit its details in this paper.

4.2 Theme Cluster Detection

In the experiments, three popular clustering algorithms are explored to produce theme clusters. In this study, given a document set, it is hard to predict the actual cluster number, and thus we typically set the number k of expected clusters as follows.

$$k = \sqrt{|V|} \qquad (6)$$

where $|V|$ is the number of all sentences in the document set.

The clustering algorithms are described as follows [8]:

Kmeans Clustering: It is a partition based clustering algorithm. The algorithm randomly selects k sentences as the initial centroids of the k clusters and then iteratively assigns all sentences to the closest cluster, and recomputes the centroid of each cluster, until the centroids do not change. The similarity between a sentence and a cluster centroid is computed using the standard cosine measure.

Agglomerative Clustering: It is a bottom-up hierarchical clustering algorithm and starts with the sentences as individual clusters and, at each step, merge the most similar or closest pair of clusters, until the number of the clusters reduces to the desired number k. The similarity between two clusters is computed using the AverageLink method, which computes the average of the cosine similarity values between any pair of sentences belonging to the two clusters respectively.

Divisive Clustering: It is a top-down hierarchical clustering algorithm and starts with one, all-inclusive cluster and, at each step, splits the largest cluster (i.e. the cluster with most sentences) into two small clusters using the Kmeans algorithm until the number of clusters increases to the desired number k.

4.3 Cluster-based Conditional Markov Random Walk Model

In order to incorporate the cluster-level information and the sentence-to-cluster relationship, the Conditional Markov Random Walk Model is based on the two-layer link graph including both the sentences and the clusters. The novel representation is shown in Figure 2. As can be seen, the lower layer is just the traditional link graph between sentences in the basic MRW model. And the upper layer represents the theme clusters. The dashed lines between these two layers indicate the conditional influence between the sentences and the clusters.

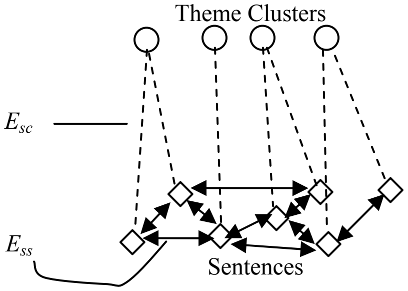

Figure 2: Two-layer link graph

Formally, the new representation for the two-layer graph is denoted as G^*=<V_s, V_c, E_{ss}, E_{sc}>, where V_s=V={v_i} is the set of sentences and V_c=C={c_j} is the set of hidden nodes representing the detected theme clusters; E_{ss}=E={$e_{ij}|v_i,v_j \in V_s$} corresponds to all links between sentences and E_{sc}={$e_{ij}|v_i \in V_s$, $c_j \in V_c$ and c_j=$clus(v_i)$} corresponds to the correlation between a sentence and its cluster. Here, $clus(v_i)$ denotes the theme cluster containing sentence v_i. For further discussions, we let $\pi(clus(v_i)) \in [0,1]$ denote the importance of cluster $clus(v_i)$ in the whole document set D, and let $\omega(v_i, clus(v_i)) \in [0,1]$ denote the strength of the correlation between sentence v_i and its cluster $clus(v_i)$.

We incorporate the two factors into the transition probability from v_i to v_j and the new transition probability is defined as follows:

$$p(i \to j \mid clus(v_i), clus(v_j)) = \begin{cases} \dfrac{f(i \to j \mid clus(v_i), clus(v_j))}{\sum_{k=1}^{|V|} f(i \to k \mid clus(v_i), clus(v_k))}, & \text{if } \sum f \neq 0 \\ 0 \ , & \text{otherwise} \end{cases} \quad (7)$$

$f(i \to j|clus(v_i), clus(v_j))$ is the new affinity weight between two sentences v_i and v_j, conditioned on the two clusters containing the two sentences. We propose to computes the conditional affinity weight by linearly combining the affinity weight conditioned on the source cluster (i.e. $f(i \to j|clus(v_i))$) and the affinity weight conditioned on the destination cluster (i.e. $f(i \to j|clus(v_j))$) as follows:

$$\begin{aligned} & f(i \to j \mid clus(v_i), clus(v_j)) \\ & \quad = \lambda \cdot f(i \to j \mid clus(v_i)) + (1-\lambda) \cdot f(i \to j \mid clus(v_j)) \\ & \quad = \lambda \cdot f(i \to j) \cdot \pi(clus(v_i)) \cdot \omega(v_i, clus(v_i)) \\ & \qquad + (1-\lambda) \cdot f(i \to j) \cdot \pi(clus(v_j)) \cdot \omega(v_j, clus(v_j)) \qquad (8) \\ & \quad = f(i \to j) \cdot (\lambda \cdot \pi(clus(v_i)) \cdot \omega(v_i, clus(v_i)) \\ & \qquad + (1-\lambda) \cdot \pi(clus(v_j)) \cdot \omega(v_j, clus(v_j))) \end{aligned}$$

where $\lambda \in [0,1]$ is the combination weight controlling the relative contributions from the source cluster and the destination cluster. Various methods can be used to compute the cluster importance and the sentence-to-cluster correlation strength, including the cosine measure, the language model measure, etc. In this study, we adopt the widely used cosine measure to measure the two factors.

$\pi(clus(v_i))$ aims to evaluate the importance of the cluster $clus(v_i)$ in the document set D, and it is set to the cosine similarity value between the cluster and the whole document set[2]:

$$\pi(clus(v_i)) = sim_{\cos ine}(clus(v_i), D) \qquad (9)$$

$\omega(v_i, clus(v_i))$ aims to evaluate the correlation between the sentence v_i and its cluster $clus(v_i)$, and it is set to the cosine similarity value between the sentence and the cluster:

$$\omega(v_i, clus(v_i)) = sim_{\cos ine}(v_i, clus(v_i)) \qquad (10)$$

Then the new row-normalized matrix $\tilde{M}^*$ is defined as follows:

$$\tilde{M}^*_{i,j} = p(i \to j \mid clus(v_i), clus(v_j)) \qquad (11)$$

The saliency scores for the sentences are then computed based on $\tilde{M}^*$ by using the iterative form in Equation (4). The final transition matrix in the Markov chain is then denoted by $A^* = \mu \tilde{M}^{*T} + \frac{(1-\mu)}{|V|} \vec{e}\vec{e}^{\,T}$ and the sentence scores is obtained by the principle eigenvector of the new transition matrix A^*.

4.4 Cluster-based HITS Model

Different from the MRW model and the ClusterCMRW model, the HITS model distinguishes the hubs and authorities in the objects. A hub object has links to many good 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 reinforcement way. In this study, we consider the theme clusters as hubs and the sentences as authorities. Figure 3 gives the bipartite graph representation, where the upper layer is the hubs and the lower layer is the authorities. The HITS model makes only use of the sentence-to-cluster relationships.

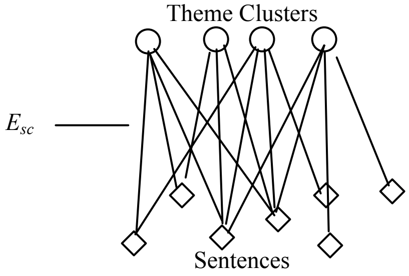

Figure 3: Bipartite link graph

Formally, the representation for the bipartite graph is denoted as $G^\#$=<V_s, V_c, E_{sc}>, where V_s=V={v_i} is the set of sentences (i.e. authorities) and V_c=C={c_j} is the set of theme clusters (i.e. hubs); E_{sc}={$e_{ij}|v_i \in V_s$, $c_j \in V_c$} corresponds to the correlations between any sentence and any cluster. Each edge e_{ij} is associated with a weight w_{ij} denoting the strength of the relationship between the sentence v_i and the cluster c_j. Similarly, the weight w_{ij} is computed

[2] A sentence cluster (or document set) is treated as a single text by concatenating all the sentence texts (or document texts).

by using the cosine measure. We let $L = (L_{i,j})_{|V_s|\times|V_c|}$ denote the adjacency matrix and L is defined as follows.

$$L_{i,j} = w_{ij} = sim_{\cos ine}(v_i, c_j) \tag{12}$$

Then the authority score $AuthScore^{(t+1)}(v_i)$ of sentence v_i and the hub score $HubScore^{(t+1)}(c_j)$ of cluster c_j at the $(t+1)^{\text{th}}$ iteration are computed based on the hub scores and authority scores at the t^{th} iteration as follows.

$$AuthScore^{(t+1)}(v_i) = \sum_{c_j \in V_c} w_{ij} \cdot HubScore^{(t)}(c_j) \tag{13}$$

$$HubScore^{(t+1)}(c_j) = \sum_{v_i \in V_s} w_{ij} \cdot AuthScore^{(t)}(v_i) \tag{14}$$

And the matrix form is

$$\vec{a}^{(t+1)} = L\vec{h}^{(t)} \tag{15}$$

$$\vec{h}^{(t+1)} = L^T \vec{a}^{(t)} \tag{16}$$

where $\vec{a}^{(t)} = [AuthScore^{(t)}(v_i)]_{|V_s|\times 1}$ is the vector of authority scores for the sentences at the t^{th} iteration and $\vec{h}^{(t)} = [HubScore(c_j)^{(t)}]_{|V_c|\times 1}$ is the vector of hub scores for the clusters at the t^{th} iteration. In order to guarantee the convergence of the iterative form, $\vec{a}$ and $\vec{h}$ are normalized after each iteration as follows.

$$\vec{a}^{(t+1)} = \vec{a}^{(t+1)} / \left|\vec{a}^{(t+1)}\right| \tag{17}$$

$$\vec{h}^{(t+1)} = \vec{h}^{(t+1)} / \left|\vec{h}^{(t+1)}\right| \tag{18}$$

It can be proved that authority vector $\vec{a}$ converges to the dominant eigenvector of the authority matrix LL^T, and hub vector $\vec{h}$ converges to the dominant eigenvector of the hub matrix L^TL. For numerical computation of the scores, the initial scores of all sentences and clusters 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 sentences and clusters falls below a given threshold (0.0001 in this study).

Finally, we use the authority scores as the saliency scores for the sentences. The sentences are then ranked and chosen into summary.

5. EXPERIMENTS

5.1 Data Set

Generic multi-document summarization has been one of the fundamental tasks in DUC 2001[3] and DUC 2002[4] (i.e. task 2 in DUC2001 and task 2 in DUC2002), and we used the two tasks for evaluation. DUC2001 provided 30 document sets and DUC2002 provided 59 document sets (D088 is excluded from the original 60 document sets by NIST) and generic abstracts of each document set with lengths of approximately 100 words or less were required to be created. The documents were news articles collected from TREC-9. The sentences in each article have been separated and the sentence information has been stored into files. The summary of the two datasets are shown in Table 1.

Table 1: Summary of data sets

	DUC 2001	DUC 2002
Task	Task 2	Task 2
Number of documents	309	567
Number of clusters	30	59
Data source	TREC-9	TREC-9
Summary length	100 words	100 words

5.2 Evaluation Metric

We used the ROUGE [16] toolkit[5] for evaluation, which has been widely adopted by DUC for automatic summarization evaluation. It measures summary quality by counting overlapping units such as the n-gram, word sequences and word pairs between the candidate summary and the reference summary. ROUGE-N is an n-gram recall measure computed as follows:

$$ROUGE\!-\!N = \frac{\sum_{S\in\{\operatorname{Re} f\,Sum\}} \sum_{n\text{-}gram\in S} Count_{match}(n-gram)}{\sum_{S\in\{\operatorname{Re} f\,Sum\}} \sum_{n\text{-}gram\in S} Count(n-gram)} \tag{19}$$

where n stands for the length of the n-gram, and *Count*$Count_{match}(n\text{-}gram)$ is the maximum number of n-grams co-occurring in a candidate summary and a set of reference summaries. *Count(n-gram)* is the number of n-grams in the reference summaries.

ROUGE toolkit reports separate scores for 1, 2, 3 and 4-gram, and also for longest common subsequence co-occurrences. Among these different scores, unigram-based ROUGE score (ROUGE-1) has been shown to agree with human judgment most [16]. We show three of the ROUGE metrics in the experimental results: ROUGE-1 (unigram-based), ROUGE-2 (bigram-based), and ROUGE-W (based on weighted longest common subsequence, weight=1.2).

In order to truncate summaries longer than the length limit, we use the "-l" option in ROUGE toolkit and we also use the "-m" option for word stemming.

5.3 Experimental Results

The proposed ClusterCMRW and ClusterHITS models with different clustering algorithms are compared with the baseline MRW model, the top three performing systems and two baseline systems on DUC2001 and DUC2002 respectively. The top three systems are the systems with highest ROUGE scores, chosen from the performing systems on each task respectively. The lead baseline and coverage baseline are two baselines employed in the generic multi-document summarization tasks of DUC2001 and DUC2002. The lead baseline takes the first sentences one by one in the last document in the collection, where documents are assumed to be ordered chronologically. And the coverage baseline takes the first sentence one by one from the first document to the last document. Tables 2 and 3 show the comparison results on DUC2001 and DUC2002 respectively. In Table 2, SystemN, SystemP and SystemT are the top three performing systems for DUC2001. In Table 3, System19, System26, System28 are the top

[3] http://www-nlpir.nist.gov/projects/duc/guidelines/2001.html

[4] http://www-nlpir.nist.gov/projects/duc/guidelines/2002.html

[5] We use ROUGEeval-1.4.2 downloaded from http://haydn. isi.edu/ROUGE/

three performing systems for DUC2002. ClusterCMRW and ClusterHITS rely on the underlying clustering algorithm. For example, ClusterCMRW(Kmeans) refers to the ClusterCMRW model using the Kmeans algorithm to detect theme clusters. For the ClusterCMRW models, the combination weight λ is typically set to 0.5 without tuning, i.e. the two clusters for two sentences contribute equally to the conditional transition probability.

Table 2: Comparison results on DUC2001

System	ROUGE-1	ROUGE-2	ROUGE-W
ClusterCMRW (Kmeans)	0.35824	0.06458*	0.10770
ClusterCMRW (Agglomerative)	0.35707	0.06548*	0.10841
ClusterCMRW (Divisive)	0.35549	0.06073	0.10722
ClusterHITS (Kmeans)	0.35756	0.05944	0.10771
ClusterHITS (Agglomerative)	0.36897*	0.06392*	0.11139*
ClusterHITS (Divisive)	0.37419*	0.06881*	0.11245*
MRW	0.35527	0.05608	0.10641
SystemN	0.33910	0.06853	0.10240
SystemP	0.33332	0.06651	0.10068
SystemT	0.33029	0.07862	0.10215
Coverage	0.33130	0.06898	0.10182
Lead	0.29419	0.04033	0.08880

Table 3: Comparison results on DUC2002

System	ROUGE-1	ROUGE-2	ROUGE-W
ClusterCMRW (Kmeans)	0.38221*	0.08321	0.12362
ClusterCMRW (Agglomerative)	0.38546*	0.08652*	0.12490*
ClusterCMRW (Divisive)	0.37999	0.08389	0.12384*
ClusterHITS (Kmeans)	0.37643	0.08135	0.12141
ClusterHITS (Agglomerative)	0.37768	0.07791	0.12271
ClusterHITS (Divisive)	0.37872	0.08133	0.12282
MRW	0.37595	0.08304	0.12173
System26	0.35151	0.07642	0.11448
System19	0.34504	0.07936	0.11332
System28	0.34355	0.07521	0.10956
Coverage	0.32894	0.07148	0.10847
Lead	0.28684	0.05283	0.09525

(* indicates that the improvement over the baseline MRW model is statistically significant.)

Seen from the tables, both the ClusterCMRW model and the ClusterHITS model with different clustering algorithms can outperform the basic MRW model and other baselines over almost all three metrics on both DUC2001 and DUC2002 datasets. The results demonstrate the good effectiveness of the proposed models. Moreover, the three clustering algorithms are validated to be as effective as each other. It is a little disappointing that one proposed model cannot always outperform the other proposed model on both datasets.

In order to investigate how the combination weight influences the summarization performance of the ClusterCMRW model, we vary the combination weight λ from 0 to 1 and Figures 4-7 show the ROUGE-1 and ROUGE-2 curves on the DUC2001 and DUC2002 datasets respectively. The similar ROUGE-W curves are omitted due to the page limit. We can see from the figures that the proposed ClusterCMRW model with different clustering algorithms can almost always outperform the baseline MRW model, under different values of λ. The results show the robustness of the proposed ClusterCMRW model, with respect to different combination weights.

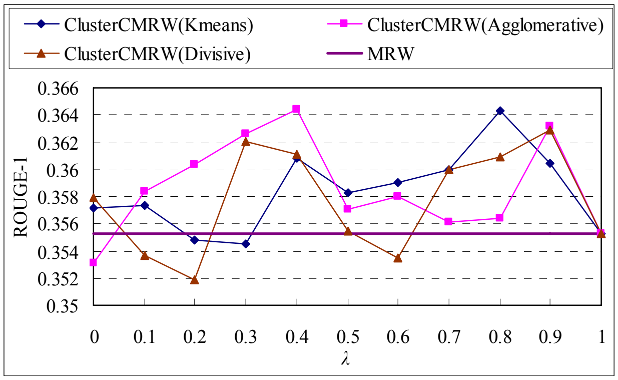

Figure 4: ROUGE-1 vs. λ for ClusterCMRW on DUC2001

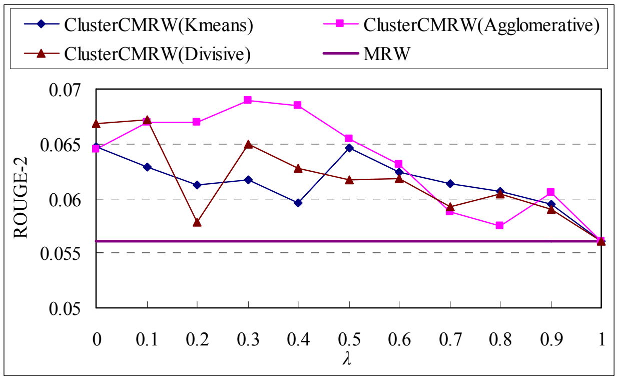

Figure 5: ROUGE-2 vs. λ for ClusterCMRW on DUC2001

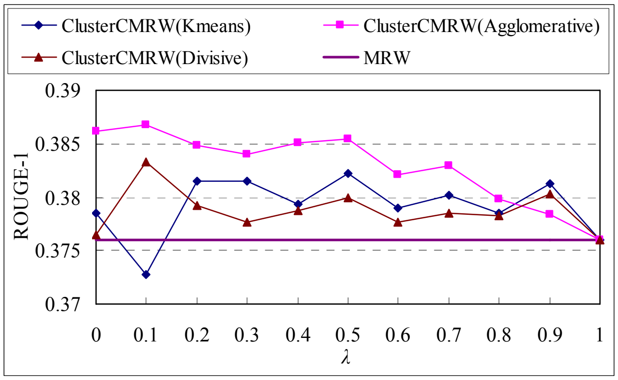

Figure 6: ROUGE-1 vs. λ for ClusterCMRW on DUC2002

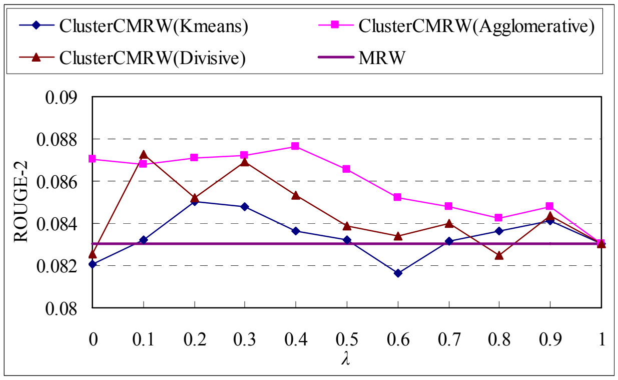

Figure 7: ROUGE-2 vs. *λ* for ClusterCMRW on DUC2002

Note that in the above experiments, the cluster number *k* is typically set to the square root of the sentence number. We further vary *k* to investigate how the cluster number influences the summarization performance. Given a document set, we let *V* denote the sentence collection for the document set, and *k* is set in the following way:

$$k = r \times | V | \qquad (20)$$

where *r*□(0,1) is a ratio controlling the expected cluster number for the document set. The larger *r* is, the more clusters will be produced and used in the algorithm. *r* ranges from 0.1 to 0.9 in the experiments and Figures 8-11 show the ROUGE-1 and ROUGE-2 results of ClusterCMRW and ClusterHITS on the DUC2001 and DUC2002 datasets, respectively.

Seen from the figures, the ClusterCMRW models can almost always outperform the baseline MRW model, no matter how many clusters are used. However, the ClusterHITS models are much influenced by the cluster number and very many clusters will deteriorate the performances of the ClusterHITS models. We can see from Figures 8, 10 and 11 that the performances of the ClusterHITS models are even worse than the baseline MRW model, when *r* is set to a large value. The results demonstrate that the ClusterCMRW model is more robust than the ClusterHITS model, with respect to different cluster numbers. The results can be explained that the ClusterCMRW model involves both the sentence-to-sentence relationships and the sentence-to-cluster relationships, while the ClusterHITS model makes only use of the sentence-to-cluster relationships, so the performance of the ClusterHITS model will be highly affected by the detected theme clusters.

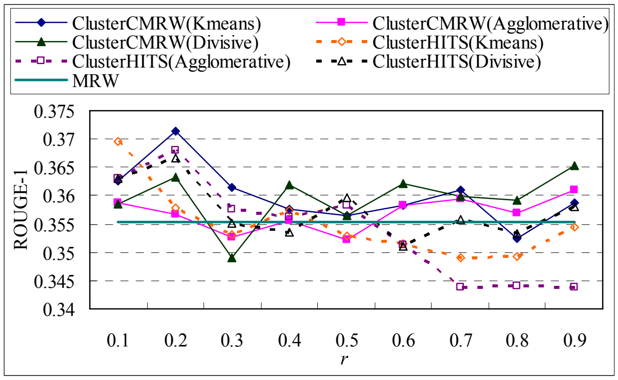

Figure 8: ROUGE-1 vs. *r* on DUC2001

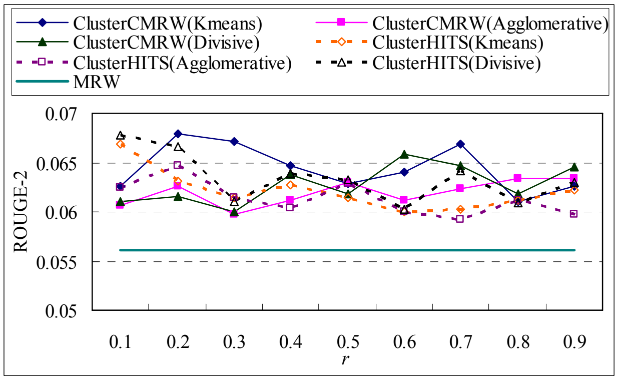

Figure 9: ROUGE-2 vs. *r* on DUC2001

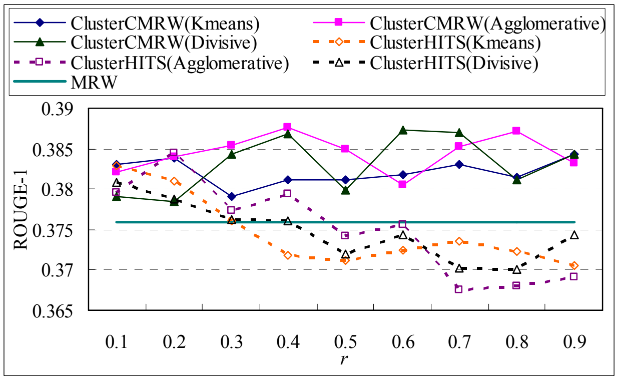

Figure 10: ROUGE-1 vs. *r* on DUC2002

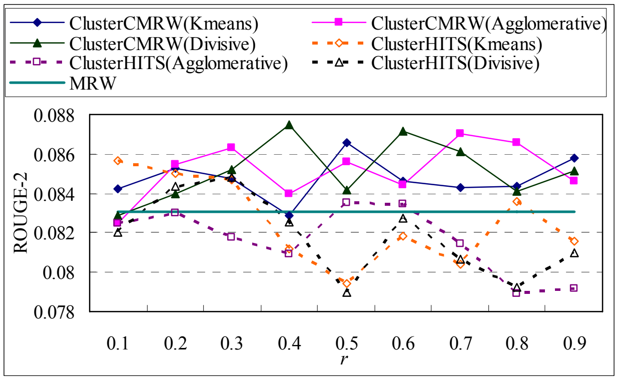

Figure 11: ROUGE-2 vs. *r* on DUC2002

6. CONCLUSION AND FUTURE WORK

In this paper we propose two novel summarization models to make use of the theme clusters in the document set. The first model incorporates the cluster information in the Conditional Markov Random Walk Model and the second model uses the HITS algorithm by considering the cluster as hubs and the sentences as authorities. Experimental results on the DUC2001 and DUC2002 datasets demonstrate the good effectiveness of the models, and the cluster-based Conditional Markov Random Walk Model is validated to be more robust than the Cluster-based HITS Model.

In this study, the themes in the document set are discovered by simply clustering the sentences, and the quality of the clusters might not be guaranteed. In future work we will use other theme detection methods to find meaningful theme clusters. Moreover, we will exploit other link analysis methods to incorporating the cluster-level information.

7. ACKNOWLEDGMENTS

This work was supported by the National Science Foundation of China (No.60703064) and the Research Fund for the Doctoral Program of Higher Education of China (No.20070001059). We thank the anonymous reviewers for their useful comments.

8. REFERENCES

[1] R. Baeza-Yates and B. Ribeiro-Neto. Modern Information Retrival. ACM Press and Addison Wesley, 1999.

[2] R. Barzilay, K. R. McKeown, and M. Elhadad. Information fusion in the context of multi-document summarization. In Proceedings of ACL1999.

[3] H. Daumé and D. Marcu. Bayesian query-focused summarization. In Proceedings of COLING-ACL2006.

[4] G. Erkan and D. Radev. LexPageRank: prestige in multi-document text summarization. In Proceedings of EMNLP2004.

[5] J. Goldstein, M. Kantrowitz, V. Mittal, and J. Carbonell. Summarizing Text Documents: Sentence Selection and Evaluation Metrics. In Proceedings of ACM SIGIR1999.

[6] S. Harabagiu and F. Lacatusu. Topic themes for multi-document summarization. In Proceedings of SIGIR2005.

[7] H. Hardy, N. Shimizu, T. Strzalkowski, L. Ting, G. B. Wise, and X. Zhang. Cross-document summarization by concept classification. In Proceedings of SIGIR2002.

[8] K. Jain, M. N. Murty and P. J. Flynn. Data clustering: a review. ACM Computing Surveys, 31(3):264-323, 1999.

[9] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. Journal of the ACM, 46(5):604-632, 1999.

[10] K. Knight and D. Marcu. Summarization beyond sentence extraction: a probabilistic approach to sentence compression, Artificial Intelligence, 139(1), 2002.

[11] W. Kraaij, M. Spitters and M. van der Heijden. Combining a mixture language model and Naïve Bayes for multi-document summarization. In SIGIR2001 Workshop on Text Summarization.

[12] O. Kurland and L. Lee. PageRank without hyperlinks: structural re-ranking using links induced by language models. In Proceedings of SIGIR2005.

[13] O. Kurland and L. Lee. Respect my authority! HITS without hyperlinks, utilizing cluster-based language models. In Proceedings of SIGIR2006.

[14] A. Leuski, C.-Y. Lin and E. Hovy. iNeATS: interactive multi-document summarization. In Proceedings of ACL2003.

[15] C.-Y. Lin and E.H. Hovy. From Single to Multi-document Summarization: A Prototype System and its Evaluation. In Proceedings of ACL2002.

[16] C.-Y. Lin and E.H. Hovy. Automatic Evaluation of Summaries Using N-gram Co-occurrence Statistics. In Proceedings of HLT-NAACL 2003.

[17] T.-Y. Liu and W.-Y. Ma. Webpage importance analysis using Conditional Markov Random Walk. In Proceedings of IEEE WI2005.

[18] I. Mani and E. Bloedorn. Summarizing Similarities and Differences Among Related Documents. Information Retrieval, 1(1), 2000.

[19] D. Marcu. Discourse-based summarization in DUC–2001. 2001. In SIGIR 2001 Workshop on Text Summarization.

[20] K. McKeown, J. Klavans, V. Hatzivassiloglou, R. Barzilay, and E. Eskin. Towards multidocument summarization by reformulation: progress and prospects, in Proceedings of AAAI1999.

[21] R. Mihalcea and P. Tarau. A language independent algorithm for single and multiple document summarization. In Proceedings of IJCNLP2005.

[22] L. Page, S. Brin, R. Motwani and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Libraries, 1998.

[23] D. R. Radev, H. Y. Jing, M. Stys and D. Tam. Centroid-based summarization of multiple documents. Information Processing and Management, 40: 919-938, 2004.

[24] H. Saggion, K. Bontcheva, and H. Cunningham. Robust generic and query-based summarization. In Proceedings of EACL2003.

[25] X. Wan and J. Yang. 2006. Improved affinity graph based multi-document summarization. In Proceedings of HLT-NAACL2006.

[26] G.-R. Xue, Q. Yang, H.-J. Zeng, Y. Yu and Z. Chen. Exploiting the hierarchical structure for link analysis. In Proceedings of SIGIR2005.

[27] B. Zhang, H. Li, Y. Liu, L. Ji, W. Xi, W. Fan, Z. Chen, and W.-Y. Ma. Improving web search results using affinity graph. In Proceedings of SIGIR2005.

[28] D. Zhou, S. A. Orshanskiy, H. Zha and C. L. Giles. Co-ranking authors and documents in a heterogeneous network. In Proceedings of IEEE ICDM2007.