HITIR's update summary at TAC2008: Extractive content selection for language independence

Ruifang He, Yang Liu, Bing Qin, Ting Liu, Sheng Li
Information Retrieval Lab,School of Computer Science and Technology
Harbin Institute of Technology, Harbin 15001, China
rfhe@ir.hit.edu.cn

Abstract

The update summary aims to capture evolving information of a single topic changing over time. It delivers salient and novel information to a user who has already read a set of older documents covering the same topic. According to the new challenges brought by update summary, we propose the evolutionary manifold-ranking algorithm, and further integrate the sub-topics partition with spectral clustering to have a content selection, which is completely language independence. Three systems: 11, 41 and 62 are submitted. Our best system ranks three top 1 under average modified (pyramid) score, average numSCUs and macro-average modified score with 3 models of PYRAMID, ranks 13^{th} *in ROUGE-2, ranks* 15^{th} *in ROUGE-SU4 and ranks* 17^{th} *in BE. Though the evaluation results show the interesting performance of the proposed method, yet the problem is far from solved.*

1. Introduction

Update summary task is proposed by Document Understanding Conference(DUC)2007, and becomes a sub-task of Text Analysis Conference(TAC)2008. It was conducted by National Institute of Standards and Technology(NIST) to have a first evaluation conference on January, 2007 under the support from Intelligence Advanced Research Projects Activity(IARPA). Update summary is the natural extension of traditional multi-document summarization, which captures evolving information of a single topic changing over time under the assumption that people has already read a set of older documents covering the same topic. Their difference lies that update summary deals with the dynamic document collection, yet not static document collection in the same period of time. A news topic usually has its life cycle, including birth, growth, decay, and death, reflecting its popularity over time. Therefore, people hope to incrementally care the important and novel information so as to reduce the burden of acquiring information.

TAC is organized by the Retrieval Group of the Information Access Division (IAD) in the Information Technology Laboratory at the NIST. Initiated in 2008, TAC grew out of NIST's DUC for text summarization, and the Question Answering Track of the Text Retrieval Conference (TREC). Just like the expectation of the evaluation road map for summarization research[1], which encourages some integration of the summarization and question answering tasks, the tracks in TAC also demonstrate the robustness of core NLP technology in that the same techniques are frequently appropriate for a variety of tasks. Much of the past data from TREC and DUC are available from NIST and may be useful for system development for some of the TAC tracks. The data include test topics/judgments (on the TREC and DUC web sites) and document disks. The document disks that are available from NIST include the TIPSTER set (disks 1-3), the TREC set (disks 4-5), the AQUAINT set, and the AQUAINT-2 set. For update summary, test documents comes from the AQUAINT-2 collection. The test dataset comprises approximately 48 topics. Each topic has a topic statement (title and narrative) and 20 relevant documents which have been divided into 2 sets: document set A and document set B. Each document set has 10 documents, where all the documents in set A chronologically precede the documents in set B. The TAC 2008 Update Summary task is to generate short (100 words) fluent multi-document summaries of news articles under the assumption that the user has already read a set of earlier articles. The purpose of each update summary will be to inform the reader of new information about a particular topic.

The key problem of summarization is how to identify important content and remove redundant content. Traditional multi-document summarization has drawn much attention for recent years, while the research about the next generation summarization, update summary just begins. Their common problem is that the information in different documents inevitably overlaps with each other, and therefore effective summarization methods are needed to contrast their similarities and differences. The difference between the two ones is that the former deals with static document set and the latter handles the dynamic document set where the objects to be summarized face to some special topics and evolve with time. Therefore, update summary raises new challenges to traditional summarization algorithms. The first challenge is that the information in the summary must be biased to the given topic, and the second is that the information in summary must contain the evolving content. So we need to effectively take into account this topic-biased and temporally evolving characteristics during the summarization process. Thus a good update summary must include information as much as possible, keeping information as novel as possible, and moreover, the information must be biased to the given topic.

In [7], an extractive approach based on manifold-ranking of sentences to topic-focused multi-document summarization by using the underlying manifold structure in data points is proposed, yet which could not model the temporally evolving characteristic. Inspired by this, for the TAC2008 update summary task, we propose a new manifold-ranking frame based on iterative feedback mechanism, which has the temporally adaptive characteristic. We assume that the data points evolving over time have the long and narrow manifold structure. However, the common topic for consecutive document subsets is a static query, which cannot represent the dynamically evolving information. Therefore, we use the iterative feedback mechanism to extend the topic by using the summarization sentences of previous timeslices and the first sentences of documents in current timeslice. We believe this topic extension can represent the relay propagation of information in temporally evolving data and improve the ranking score. The proposed approach employs iterative feedback based evolutionary manifold-ranking process to compute the evolutionary ranking score for each sentence that denotes the importance and the topic-relevance of sentence. Then the sentences highly overlapping with other informative ones are penalized by the greedy algorithm. The summary is produced by choosing the sentences with highest overall scores, which are considered informative, novel and evolving. In this improved manifold-ranking algorithm, the intra-document and inter-document relationships between sentences are also differentiated with different weights.

Though we can choose the summary sentences only by evolutionary manifold-ranking and removal algorithms, yet the problem of the coverage about summary could not be better resolved judged by the experiments on DUC 2007 update summary task. Therefore, we further propose a new extractive approach based on sub-topic partition with spectral clustering and evolutionary manifold-ranking to update summary. Experiments results on datasets of TAC 2008 update task demonstrate the competitive performance of the proposed approach.

The rest of this paper is organized as follows: Section 2 provides an overview of our update summary system. Section 3 discusses the results from the official TAC 2008 evaluation, and we conclude this paper in Section 4.

2. System description

In order to model the new characteristics of update summary and resolve the problem about full coverage of summary content, we design three groups of experiments:

1) Spectral clustering (post-processing: properties of eigenvector) + evolutionary manifold-ranking;

2) Spectral clustering (post-processing:k-means) + evolutionary manifold-ranking;

3) Evolutionary manifold-ranking;

Temporally evolving document collection can be considered as several relevantly continuous timeslices. Every timesslice are composed of dynamical documents, which consists of scattered sentences. For each timeslice, say document collection A or B in the corpus of TAC 2008 update summary task, through spectral clustering, the number of sub-topics are automatically determined by the multiplicity of eigenvalue 0 of similarity matrix about sentences. Thus partition of sub-topics helps to select summary sentences from different aspects and improve the full coverage of summary content. Ordering sub-topics and selecting sentences are dependent on the rank score from evolutionary manifold-ranking, where iterative feedback mechanism is applied to model the dynamically evolving characteristics and represent the relay propagation of information in temporally evolving data. The rank score of a sentence also represents its topic relevance and the importance among document collection in current timeslice and topic. The summary is iteratively produced by choosing the sentences in sub-topic, which are considered informative, novel and evolving. Then the sentences highly overlapping with other informative ones in the sub-topic are penalized by the greedy algorithm. This method not only improves the coverage of summary content through spectral clustering, but also integrates the temporally evolving characteristic and ranks data points along their underlying manifold structure.

The proposed approach has two-fold: one is to improve the full coverage of summary content by spectral clustering; the other is to model the relay propagation of information in temporally evolving dataset of a single topic. For every timeslice, it consists of the following steps:

1) Partition sentences in current timeslice into sub-topics by spectral clustering;

2) Rank sentences through evolutionary manifold-ranking algorithm;

3) Order the sub-topics and select sentences according to their rank scores;

In fact, step 1) and 2) are independent and there are no order. We can only depend on the step 2) and 3) to produce the summary. These key steps will be illustrated in detail in next sections, respectively.

2.1 The Spectral Clustering Algorithm for Sub-topics Partition

Clustering similar sentences in semantic, the logic sub-topics in document collection are formed, then summary is extracted from different logic sub-topics. This method can reduce the redundancy and improve the full coverage of summary content. Furthermore, it is not limited to the shallow understanding for text units in document collection, but analyzes their logic structure. However, it needs to decide the number of logic sub-topic in advance, which is judged by the compactness of the content. Since Endre Boros[2] applied the clustering to summary system, people began to consider using it to resolve the coverage problem of summary content. Nevertheless, traditional clustering methods cannot automatically get the number of clusters, and can only deal with the data with convex shape. Spectral clustering not only can automatically determine the number of logical sub-topics, but also can cluster the data points with arbitrary shape and converge to the globally optimal solution[8]. It is a kind of technique which depends on the eigenvalue and eigenvector structure of a similarity matrix to partition data points into disjoint clusters. It makes sure that points in the same cluster have high similarity and points in different clusters have low similarity. Spectral clustering builds on the spectral graph theory and the analysis of spectral graph theory are based on the graph Laplacians matrices.

2.1.1 Graph Laplacians and Number of clusters

Given a set of data points $x_1, ...x_n$, let an undirected and weighted similarity graph $G = (V, E)$ represent data points. Vertex set $V = x_1, ...x_n$and each edge between two vertices carries a non-negative weight $w_{ij \geq 0}$. The weighted adjacent matrix $W = (w_{ij})_{i,j=1,...,n}$. If $w_{ij} = 0$ this means that the vertices x_i and x_j are not connected. Since G is undirected, W is a symmetric matrix. D is the diagonal matrix with (i, i)-element equal to the sum of the i-th row of W.

The unnormalized graph Laplacian matrix is defined as $L = D - W$, and the normalized graph Laplacians are defined as

$L_{sym} := D^{-1/2}LD^{-1/2} = I - D^{-1/2}WD^{-1/2}$

$L_{rw} := D^{-1}L = I - D^{-1}W$

Here, L_{sym} is a symmetric matrix, and the L_{rw} is closely connected to a random walk. Laplacian has many good properties. The most significant property for our problem is that the multiplicity k of the eigenvalue 0 of L,L_{rw} and L_{sym} equals the number of connected components in the graph, say, the number of clusters formed by data points. The specific proof can be referenced in Ulrike von Luxburg[6].

2.1.2 Normalized Spectral Clustering

From graph cut point of view, spectral clustering can be derived as an approximation to graph partitioning problems. Because mincut graph problem is a NP hard problem, spectral clustering is a way to solve relaxed versions of this problem[6]. Relaxing Ncut leads to normalized spectral clustering, while relaxing RatioCut leads to unnorlized spectral clustering. Normalized spectral clustering implements both minimization of between-cluster similarity and maximization of within-cluster similarity, while unnormalized spectral clustering only implements the first objective. Therefore, we select the normalized spectral clustering working with eigenvectors of the normalized Laplacian matrix L_{rw}. For choosement of the normalized Laplacian matrix, we are in favor of L_{rw}. The reason is that L_{sym} might lead to undesired artifacts and fail to converge. At the same time, as using L_{sym} also does not have any computational advantages, we thus advocate for using L_{rw}.

For each timeslice, the following algorithm is used to partition the sentences in current timeslice into sub-topics.

Algorithm 1 Normalized spectral clustering

Input: Sentences set $X = \{x_1, ..., x_n\}$;
Output: Sub-topics $\{C_1, ..., C_k\}$;

1: Construct the similarity matrix $W \in R^{n\times n}$ according to the $\varepsilon-$neighbour graph;
2: Compute the normalized Laplacian L_{rw} and its eigenvalue;
3: Rank eigenvalues $0 \leq \lambda_1 \leq \lambda_2 \leq ... \leq \lambda_n$ and compute the first k eigenvector $v_1, ..., v_k$ of the generalized eigenproblem $L_{rw}v = \lambda v$;
4: Let $V \in R^{n\times n}$ be the matrix containing the vectors $v_1, ..., v_k$ as columns;
5: For $i = 1, ..., n$, let $y_i \in R^k$ be the vector corresponding to the $i - th$ row of V;
6: Cluster the points $(y_i)_{i=1,...,n} \in R^k$ with post-processing into clusters $C_1, ..., C_k$;

The algorithm works with eigenvectors of the normalized Laplacian L_{rw}, and hence is called normalized spectral clustering. In the first step, we compute the pair-wise similarity values between sentences (data points) using the standard Cosine measure. Given two sentences x_i and x_j, the Cosine similarity is denoted as $sim(x_i, x_j)$. We define the affinity matrix W by $W_{ij} = sim(x_i, x_j)$ if there is an edge linking x_i and x_j and $sim(x_i, x_j) \geq \varepsilon$. Note that we let $W_{ii} = 0$ avoid loops in the similarity graph. We remove the stop words in each sentence, and stem the remaining words. The weight associated with term t is calculated with the $tf_t * isf_t$ formula, where tf_t is the frequency of term t in the sentence and isf_t is the inverse sentence frequency of term t, i.e. $1 + log(N/n_t)$, where N is the total number of sentences and n_t is the number of the sentences containing term t. Then $sim(x_i, x_j)$ is computed according to the normalized inner product of the corresponding term vectors.

During the phase of post-processing, we can use the properties of eigenvector to cluster data points, and also can use the k-means to do this thing.

When sub-topics are partitioned through spectral clustering, we need to compute the evolutionary rank score of sentences in current timeslice. In fact, we also can compute it firstly.

2.2 Iterative Feedback Mechanism based Evolutionary Manifold-ranking

The manifold-ranking method[10, 9] is a universal ranking algorithm. Given a query and a set of data points, the task of manifold-ranking[9] is to rank the data points along their underlying manifold structure according to their relevance to the query. However, this method cannot model the temporally evolving characteristic, say, which is not temporally adaptively. For the TAC 2008 update summary, it can be considered as a topic-relevant ranking question. We assume that the data points evolving over time have the long and narrow manifold-structure. Nevertheless, the common topic for consecutive timeslices is a static query, which cannot represent the dynamically evolving information. Therefore, we improve the manifold-ranking by adding evolutionary adaptiveness and apply the iterative feedback mechanism to extend the topic by using the summarization of previous timeslices and the first sentences of documents in the current timeslice.

The iterative feedback mechanism based evolutionary manifold-ranking approach consists of two steps: (1) iterative feedback mechanism is used to extend the topic; (2) ranking score is computed for each sentence in the evolutionary manifold-ranking process where score denotes the importance of a sentence relevant to the sentence collection and topic;

2.2.1 Iterative Feedback Mechanism

Given a set of timeslices $TS = \{timeslice_i | 1 \leq i \leq m\}$ and a topic $T = \{topic_i | 1 \leq i \leq m\}$, every $timeslice_i$ consists of documents, $timeslice_i = \{d_j | 1 \leq j \leq n\}$, every document consists of sentences. Let s_{ij} denote the first sentence of document d_j in $timeslice_i$, then first sentences of all documents in $timeslice_i$ $s_{first}(i) = \{s_{ij} | 1 \leq i \leq m, 1 \leq j \leq n\}$. The timeslices are ordered chronologically. Every timeslice corresponds to an update summary. When summarizing, the current $timeslice_i$ just can refer to the previous timeslices from 1 to i-1, but cannot refer to the ones from i+1 to m. Let $updateSum_i$ denote the update summary of the current $timeslice_i$, and then $topic_i$ is extended as follows:
$topic_i = \{PubTopic \cup \overset{i-1}{\underset{k=1}{\cup}} updateSum_k \cup s_{first}(i) | 1 \leq i \leq m\}$, $PubTopic$ denotes the public topic description of all timeslices.

We believe this topic extension can represent the relay propagation of information in temporally evolving data points and help to capture the changes of a single topic over time.

2.2.2 Evolutionary Manifold-Ranking Process

In our context, the data points are denoted by the topic description and all the sentences in the documents, where topic description dynamically evolves over time. The iterative feedback mechanism based evolutionary manifold-ranking process in our context can be formalized as following Algorithm 2:

For $timeslice_i$, given a set of data points $X = \{x_1, ..., x_t, x_{t+1}, ..., x_n\} \subset R^m$, the first t data points are the topic description and the rest data points are the sentences in the documents. According to the iterative feedback mechanism, x_1 denotes the $PubTopic$, $x_2 ... x_p$ denotes the $\overset{i-1}{\underset{k=1}{\cup}} updateSum_k$ and $x_{p+1} ... x_t$ denotes the $s_{first}(i)$. Note that because the $PubTopic$ is usually short in our experiments, we treat it as a pseudo-sentence. Then it can be processed in the same way as other sentences. Let $f : X \rightarrow R$ denote a ranking function which assigns to each point $x_q (1 \leq q \leq n)$ a ranking value f_q. We can view f as a vector $f = [f_1, ..., f_n]^T$. We also define three vectors, $Y_1 = [y_1, ..., y_n]^T$, in which $y_1 = 1$ because x_1 is the $PubTopic$ and $y_q = 0 (2 \leq q \leq n)$ for all the sentences in the documents; similarly, $Y_2 = [y_1, ..., y_n]^T$, in which $y_2 ... y_p = 1$ because $x_2 ... x_p$ denotes the $\overset{i-1}{\underset{k=1}{\cup}} updateSum_k$ and $y_q = 0 (q = 1, p + 1 \leq q \leq n)$; $Y_3 = [y_1, ..., y_n]^T$, in which $y_{p+1} ... y_t = 1$ because $x_{p+1} ... x_t$ denotes the $s_{first}(i)$ and $y_q = 0 (1 \leq q \leq p, t + 1 \leq q \leq n)$. The iterative feedback based manifold-ranking algorithm goes as follows:

In the first step of the algorithm, a connected network is formed and weighted. The weight is symmetrically normalized in the second step. The normalization is necessary to prove the algorithm's convergence. The third step is the key step of the algorithm, where all points spread their ranking score to their neighbors via the weighted network.

Algorithm 2 Iterative feedback based evolutionary manifold-ranking

Input: $X = \{x_1, ..., x_t, x_{t+1}, ..., x_n\}$
Output: $f = \{f_i^* | i = 1...n\}$

1: Construct the similarity graph matrix W similar to the one in the Algorithm 1;
2: Connect any two points with an edge if their similarity value exceeds 0.
3: Normalize W by $S = D^{-1}W$ in which D is the diagonal matrix with (i, i)-element equal to the sum of the i-th row of W;
4: Iterate $f(t+1) = \alpha S f(t) + (\beta Y_1 + \gamma Y_2 + \eta Y_3)$ until convergence, where $\alpha, \beta, \gamma, \eta$ are parameters in (0,1);
5: Let f_i^* denote the limit of the sequence $\{f_i(t)\}$. Each sentence x_i gets its ranking score f_i^*;

The spread process is repeated until a global stable state is achieved, and we get the rank score in the fifth step. The parameter α specifies the relative contributions to the rank scores from neighbors and the initial rank scores, and the parameter β, γ, η denotes the relative contribution to rank scores from the $PubTopic$, the update summary sentences in the previous timeslices and the first sentences of all documents in the current timeslice, respectively. Note that self-reinforcement is avoided since the diagonal elements of the affinity matrix are set to zero.

For the original manifold-ranking, the iterative formula of the fourth step is $f(t+1) = \alpha S f(t) + (1 - \alpha)Y$. The theorem in [9] guarantees that the sequence $f(t)$ converges to

$$f^* = (I - \alpha S)^{-1} Y \tag{1}$$

Without loss of the generality, we can extend the vector Y. Since $(I - \alpha S)$ is invertible, we have

$$f^* = (I - \alpha S)^{-1} (\beta Y_1 + \gamma Y_2 + \eta Y_3) \tag{2}$$

For real-world problems, the iteration algorithm is preferable due to high computational efficiency. Usually when the difference between the scores computed at two successive iterations for any point falls below a given threshold(0.0001 in this paper), the iteration algorithm will converge.

Evolutionary manifold-ranking process also naturally make use of both the relationships among all the sentences in the documents and relationships between the topic and the sentences. After considering this, the original normalized similarity matrix S is over again normalized into $S^{'}$ in the third step and the fourth step uses the following iteration form: $f(t+1) = \alpha S^{'} f(t) + (\beta Y_1 + \gamma Y_2 + \eta Y_3)$. The iteration process is shown in Algorithm 3:

Algorithm 3 Power method for computing the stable state of iterative feedback based evolutionary manifold-ranking

Input: normalized similarity matrix $S^{'}$
Input: matrix size N, error tolerance ε
Output: eigenvector f

1: $f(0) = \frac{1}{N}$;
2: $t = 0$;
3: **repeat**
4: $f(t+1) = \alpha S^{'T} f(t) + (\beta Y_1 + \gamma Y_2 + \eta Y_3)$;
5: $t = t + 1$;
6: $\delta = ||f(t+1) - f(t)||$;
7: **until** $\delta < \varepsilon$;
8: return $f(t+1)$;

2.3 Ordering Sub-topics and Selecting Sentences

Based on the sub-topics partition using spectral clustering and evolutionary manifold-ranking, we designed a new algorithm of extracting sentences. The final overall ranking scores represent both the importance and the novelty of the sentences.

The algorithm is based on the idea that documents in each timeslice can be represented as the structure of logical sub-topics, which helps to understand the topic from different aspects; the overall rank score of less informative sentences overlapping with the sentences in update summary is decreased. The sentence with highest rank score in the most important subtopic is chosen to produce the summary until satisfying the summary length limit, which are considered informative, novel and evolving.

3. Evaluation results and discussion

TAC 2008 includes three open evaluation tracks, question answering, recognizing textual entailment and summarization. Summarization track consists of update summarization task and opinion summarization task. We just participated the update summarization task. In this task, there are 33 participants and 71 systems, and every participant can submit three systems at best. The official evaluation comprises three methods under different assumption: ROUGE[4], PYRAMID[5], and BE[3]. NIST evaluated all systems automatically using ROUGE/BE. NIST provided manual evaluations only for systems with priority 1 and 2. The participants' summarizer IDs are 0-71 in the ROUGE/BE evaluations, and 0-57 in the manual evaluations.

We submit three systems based on the schemes mentioned in Section 2 and system No. is 11, 41 and 62, respectively. The performance of our best system has three top 1 in

Table 1. The Update Summary Evaluation Results($Rank|Score$)

Evaluation Principle	11	41	62
average modified (pyramid) score	$1|0.336$	$4|0.318$	
average numSCUs	$1|4.781$	$2|4.469$	
average numrepetitions	$9|1.042$	$9|1.042$	
macroaverage modified score with 3 models	$1|0.331$	$3|0.313$	
average linguistic quality	$27|2.406$	$33|2.323$	
average overall responsiveness	$5|2.542$	$8|2.479$	
ROUGE-2	$13|0.08854$	$19|0.08353$	$15|0.08729$
ROUGE-SU4	$15|0.12477$	$19|0.12073$	$16|0.12250$
BE	$21|0.05134$	$31|0.04813$	$17|0.05228$

the evaluation system of PYRAMID: 1) Average modified (pyramid) score; 2) Average numSCUs; 3) Macro-average modified score with 3 models; ranks 13^{th} in ROUGE-2, 15^{th} in ROUGE-SU4 and 17^{th} in BE.

In our systems, N0. 11, 41 and 62 corresponds to the experiment scheme of the following 1), 2) and 3), respectively.

1) Spectral clustering (post-processing: properties of eigenvector) + evolutionary manifold-ranking;

2) Spectral clustering (post-processing:k-means) + evolutionary manifold-ranking;

3) Evolutionary manifold-ranking;

Through spectral clustering, the number of sub-topics are automatically determined by the multiplicity of eigenvalue 0 of similarity matrix about sentences. Thus partition of sub-topics helps to select summary sentences from different aspects and improve the full coverage of summary content. Ordering sub-topics and selecting sentences are dependent on the rank score from evolutionary manifold-ranking, where iterative feedback mechanism is applied to model the dynamically evolving characteristics and represent the relay propagation of information in temporally evolving data. The rank score of a sentence also represents its topic relevance. The summary is iteratively produced by choosing the sentences in sub-topic with highest rank score, which are considered informative, novel and evolving. Then the sentences highly overlapping with other informative ones in the sub-topic are penalized by the greedy algorithm.

In the experiments, we adopt off the shelf sentence segmentation and stemming modules. The stop words in each sentence were removed and the remaining words were stemmed. We also don't consider the short sentence limited to a certain threshold, which cannot carry enough information. About topic description in update summary corpus in TAC 2008, we just use title to be topic in the algorithm, which is treated as a pseudo-sentence, though it may be have more than one sentence. Table 1 is the detailed scores and rank of system 11, 41,62.

Seen from the table 1, our evaluation rank is not balanced under three different evaluation methods. We get three top 1 in PYRAMID method, nevertheless, the rank in ROUGE/BE method is just higher than the middle level. It is surprised that there is so great difference. It is probably because that our content selection method suits the idea of Pyramid evaluation method. In essence, Pyramid evaluation method adopts the voting idea to give the different weight for different importance Summary Content Unit(SUC). For our approach, we essentially find the stationary distribution of random walk in evolutionary manifold-ranking, and similarly give the higher weight for sentence with many votes through a similarity spread process. This idea is similar to the evaluation idea of Pyramid method and more importance is that we catched the evolutionary characteristic of dynamically evolving document collection. Whereas we don't do any processing of coherence and got the less linguistic quality. For the principle of average numrepetitions, we think that the more this value nears 1, the better, which means that there is little redundancy. Measuring both the content selection and linguistic quality, our overall responsiveness is not bad, ranking 5^{th}. Hovy[3]analyzed the correlation among three evaluation methods, and it showed that good relevance. However, at leat in our update summary evaluation results, I don't think so, because there is so much difference.

Three evaluation methods have three level evaluation content unit. We think that ROUGE and BE are suitable to evaluate the content selection of generative summary, which usually have the relatively short SUC and PYRAMID is suitable to evaluate the content selection of extractive summary, which has the relatively long SUC.

4. Conclusion and future work

This paper demonstrated how to use spectral clustering and evolutionary manifold-ranking to model the new characteristics of update summary and develop the extractive content selection method for language independence. It is the first time that we participated the summarization evaluation. Our TAC 2008 results is encouraging and the proposed approach not only improves the coverage of summary content through spectral clustering, but also integrates the temporally evolving characteristic. We also found some paradox need to be further investigated and there is still space to improve the state of art method.

However, it is valuable to note that the performance difference is great due to different evaluation methods. Which on earth is the best evaluation method reflecting the summary quality, we need to explore further.

5. Acknowledgments

This work was also supported by the National Natural Science Foundation of China under Grant No. 60435020, 60675034, and the National 863 High-tech Project No.2008AA01Z144. We also would like to thank the TAC 2008 organizers.

References

[1] B. Baldwin, R. Donaway, E. Hovy, E. Liddy, I. Mani, D. Marcu, K. McKeown, V. Mittal, M. Moens, D. Radev, et al. An evaluation road map for summarization research. In *TIDES*, 2000.

[2] E. Boros, P. Kantor, and D. Neu. A Clustering Based Approach to Creating Multi-Document Summaries. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2001.

[3] E. Hovy, C. Lin, L. Zhou, and J. Fukumoto. Automated Summarization Evaluation with Basic Elements. In *Proceedings of the Fifth Conference on Language Resources and Evaluation (LREC)*, 2006.

[4] C. Lin. ROUGE: A Package for Automatic Evaluation of Summaries. *Proceedings of the Workshop on Text Summarization Branches Out*, pages 25–26, 2004.

[5] A. Nenkova, R. Passonneau, and K. McKeown. The pyramid method: Incorporating human content selection variation in summarization evaluation. *ACM Trans. Speech Lang. Process.*, 4(2):4, 2007.

[6] U. von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

[7] X. Wan, J. Yang, and J. Xiao. Manifold-ranking based topic-focused multi-document summarization. In *IJCAI*, pages 2903–2908, 2007.

[8] L. Zelnik-Manor and P. Perona. Self-tuning spectral clustering. *Advances in Neural Information Processing Systems*, 17(1601-1608):16, 2004.

[9] D. Zhou, O. Bousquet, T. Lai, J. Weston, and B. Scholkopf. Learning with Local and Global Consistency. *In Proceedings of NIPS2003*, 2003.

[10] D. Zhou, J. Weston, A. Gretton, O. Bousquet, and B. Scholkopf. Ranking on Data Manifolds. *In Proceedings of NIPS2003*, 2003.