A supervised aggregation framework for multi-document summarization

Yulong Pei Wenpeng Yin Qifeng Fan Lian'en Huang
The Shenzhen Key Lab for Cloud Computing Technology & Applications (SPCCTA)
Shenzhen Graduate School
Peking University, Shenzhen 518055, P.R. China
{paul.yulong.pei,mr.yinwenpeng,fanqf1026}@gmail.com, hle@net.pku.edu.cn

ABSTRACT
In most summarization approaches, sentence ranking plays a vital role. Most previous work explored different features and combined them into unified ranking methods. However, it would be imprecise to rank sentences from a single point of view because contributions from the features are onefold in these methods. In this paper, a novel supervised aggregation approach for summarization is proposed which combines different summarization methods including LexPageRank, LexHITS, manifold-ranking method and DivRank. Human labeled data are used to train an optimization model which combines these multiple summarizers and then the weights assigned to each individual summarizer are learned. Experiments are conducted on DUC2004 data set and the results demonstrate the effectiveness of the supervised aggregation method compared with typical ensemble approaches. In addition, we also investigate the influence of training data construction and component diversity on the summarization results.

KEYWORDS: Multi-document summarization, supervised aggregation framework.

Proceedings of COLING 2012: Technical Papers, pages 2225–2242,
COLING 2012, Mumbai, December 2012.

1 Introduction

Multi-document summarization aims to generate a compressed summary by extracting the major information from a collection of documents sharing the same or similar topics. With the massive explosion of information on the web, e.g., news, blogs and microblogs, multi-document summarization, as an effective solution for information explosion, provides improved mechanisms for understanding documents and reducing information overload. Therefore it has attracted considerable attention recently.

Generally speaking, summarization can be categorized into two types: extractive summarization and abstractive summarization. Extractive summarization generates summary directly by choosing sentences from original documents while abstractive summarization requires formulating new sentences according to the text content. Although abstractive summarization could be more concise and understandable, it usually involves heavy machinery from natural language processing (Hahn and Mani, 2000). In this paper, we mainly focus on extractive multi-document summarization.

In extractive summarization tasks, sentence ranking is the issue of most concern (Wei et al., 2009). A number of methods have been proposed in the literature from different aspects to rank sentences. Feature-based approaches rank sentences by exploring combination of different features of sentences such as term frequency, sentence position, length and etc. Graph-based ranking approaches aim to design different strategies to rank sentences using random walk model to capture relations between sentences, i.e., LexPageRank (Erkan and Radev, 2004), TextRank (Mihalcea and Tarau, 2004) and DivRank (Mei et al., 2010). Nowadays, some machine learning algorithms have also been applied in summarization for learning optimal feature weights automatically, for instance, some leaning to rank models (Svore et al., 2007; Jin et al., 2010; Shen and Li, 2011) have been introduced in summarization tasks.

Most previous work concentrated on exploring different features, and the features were combined in unified ranking strategies for summarization. However, to identify the importance of sentences from a single point of view would be difficult (Wong et al., 2008) because contributions from the features are onefold in these methods. To address this problem, it is natural to propose an ensemble approach which combines different summarization methods to rank the sentences. Ensemble methods have been used in a variety of applications including web search, spam detection and collaborative filtering. Specific to ranking problems, ensemble ranking or ranking aggregation methods have been widely studied in many different tasks especially in information retrieval. However, there are limited work attempts on applying ensemble methods to summarization. In (Wang and Li, 2010) and (Wang and Li, 2011), a weighted consensus method was proposed to aggregate multiple summarization methods. Although the ensemble method used in these work outperforms individual summarizers and some other combination methods, there exists a serious drawback in unsupervised methods: because the assignment of weights to different summarizers is based on the consensus, contribution from some summarizer containing inferior ranking results may lead to an inaccurate final result.

In order to deal with the drawbacks in unified ranking strategies and unsupervised aggregation methods, we propose a supervised aggregation framework for summarization in this study. Taking a summarization task as a ranking problem, we combine several different summarizers, learn the weights assigned to each summarizer with human labeled data and then rank sentences according to their combined scores. This aggregation approach generates promising results by aggregating several different summarization methods. Experiments on DUC2004

data set have been conducted and the results demonstrate the effectiveness of the proposed supervised summarization aggregation method which outperforms typical ensemble schemes under various evaluation metrics. In addition, the influence of schemes to construct training data and component diversity on the summarization results has also been investigated.

The rest of this article is organized as follows. We briefly review the related work in Section 2 and the supervised aggregation framework for summarization is introduced in Section 3. We present the supervised aggregation summarization method implementation in Section 4 and experiments are discussed in Section 5. Finally, we draw a conclusion of this study.

2 Related work

The related work will be introduced from two aspects, first we describe some representative summarization methods and then the typical work about rank aggregation are presented briefly.

2.1 Multi-document summarization

Multi-document summarization is a process to generate a summary by reducing documents in size while retaining the main characteristics of the original documents. In order to archive this goal, different features and ranking strategies have been studied.

Traditional feature-based ranking methods explored different features of sentences to score and rank the sentences. One of the most popular feature-based methods is centroid-based method (Radev et al., 2004). Radev et al. implemented MEAD as a cetroid-based summarizer by combining several predefined features including TF*IDF, cluster centroid and position to score the sentences. Lin and Hovy (Lin and Hovy, 2002) used term frequency, sentence position, stigma words and simplified Maximal Marginal Relevance (MMR) to build the NeATS multi-document summarization system. High frequent words were proved crucial in reflecting the focus of documents (Nenkova et al., 2006) and You Ouyang et al. studied the influence of different word positions in summarization (Ouyang et al., 2010).

Graph-based ranking algorithms nowadays are successfully applied in summarization and LexPageRank (Erkan and Radev, 2004) is the representative work which is based on the PageRank algorithm (Page et al., 1999). Graph-based ranking algorithms take global information into consideration rather than rely only on vertex-specific information, therefore have been proved successful in multi-document summarization. Some methods have extended the traditional graph-based models recently including multi-layer graph incorporated with different relationship (Wan and Yang, 2008), multi-modality graph based on the manifold-ranking method (Wan and Xiao, 2009) and DivRank (Mei et al., 2010) introducing the time-variant matrix into a reinforced random walk to balance prestige and diversity.

Topic model has also been exploited in summarization recently. The query Latent Dirichlet Allocation (qLDA) model was proposed in (Tang et al., 2009), and this model takes into account the query information to extract query-oriented summaries. HIRESUM model (Haghighi and Vanderwende, 2009) was presented based on hierarchical Latent Dirichlet Allocation (hLDA) to represent content specificity as a hierarchy of topic vocabulary distributions. Celikyilmaz and Hakkani-Tur also utilized a hLDA-style model to devise a sentence-level probabilistic topic model and a hybrid learning algorithm for extracting salient features of sentences to generate summaries (Celikyilmaz and Hakkani-Tur, 2010).

To date, various machine learning methods, including unsupervised and supervised methods, have been developed for extractive summarization by learning to summarize documents automatically. For instance, Shen et al. proposed a conditional random field (CRF) based method which treats summarization task as a sequence labeling problem (Shen et al., 2007). The structural SVM approach was explored in (Li et al., 2009) to enhance diversity, coverage and balance of summary simultaneously. Learning to rank (Li, 2011) methods that are widely studied in information retrieval community have been applied in summarization (Svore et al., 2007; Jin et al., 2010; Shen and Li, 2011). These studies have used different learning strategies to rank sentences for summarizing documents.

In order to identify the importance of sentences from multiple aspects, aggregation methods can be used in summarization to combine results from different summarizers. However, aggregation methods for summarization are seldom been discussed in previous work. An exception is proposed in (Wang and Li, 2011), in this study a weighted consensus summarization based on optimization was applied in summarization by aggregating four different summarization methods.

2.2 Rank aggregation

Rank aggregation is aimed at combining results of objects from multiple ranking functions to generate a better one and it has been applied into a variety of applications including information retrieval and collaborative filtering. In general, rank aggregation can be categorized into two types: order-based and score-based (Liu et al., 2007). Order-based aggregation method takes order information as input from individual rankers and score-based method utilizes ranking scores from component rankers.

In most existing unsupervised rank aggregation methods, the final ranking decisions depend on majority voting. Median rank aggregation (Van and Erp, 2000) sorts entities based on the medians of their ranks in all the ranking lists. To treat different ranking lists with different weights, Klementiev, Roth and Small proposed an unsupervised aggregation algorithm named ULARA (Klementiev et al., 2007) to learn the weights of ranking lists online by optimizing the weighted Borda count. However, as mentioned in the Introduction, a serious drawback exists in unsupervised aggregation methods, that some inferior rankers may influence the overall performance.

In order to improve the quality of ranking aggregation, some supervised learning methods have also been proposed. The work in (Liu et al., 2007) incorporated labeled data into a supervised rank aggregation method to minimize disagreements between ranking results and labeled data. (Chen et al., 2011) proposed a semi-supervised rank aggregation approach and the work minimizes the weight disagreements of different rankers to learn the aggregation function. In the semi-supervised case, the preference constraints on several item pairs were incorporated and the intrinsic manifold structures of items are also taken into account. In (Hoi and Jin, 2008), a different semi-supervised method was proposed, which learns query-dependent weights by exploring the underlying distribution of items to be ranked and assigns two similar retrieved items with similar ranking scores.

Since ranking sentences plays an important role in summarization tasks, we can regard each result from an individual system as a ranking of sentences (Wang and Li, 2011). Then, aggregation methods can also be applied to summarization to combine multiple ranking results into an aggregation ranking to generate the combined results. However, aggregation meth-

ods for summarization are seldom been discussed in the literature. (Wang and Li, 2010) and (Wang and Li, 2011) are attempts and, in their work, a weighted consensus method was proposed for summarization and an unsupervised iteration method was applied to solve the optimization problem. Different from the unsupervised aggregation method used in these work, this paper proposes a supervised aggregation framework for summarization which combines different summarization methods and learns weights automatically with human labeled data.

3 Supervised summarization aggregation framework

In this section, we first state the problem by describing the general framework of aggregation method for summarization and then the proposed supervised summarization method is introduced in details.

3.1 Problem statement

First we consider the general framework of summarization aggregation that combines results from multiple summarizers, and each summarizer can produce a score list for sentences of a document cluster. An illustration of the framework is given by Figure 1. Suppose we have M document clusters $\{c_1, c_2, ..., c_M\}$ in the training data and the ith document cluster contains N_i sentences $\{s_1^i, s_2^i, ..., s_{N_i}^i\}$. Each sentence s_j^i is associated with a label l_j^i to denote whether it will be chosen as a summary sentence and the labels are categorized into three types as follows.

$$l_j^i = \begin{cases} +1 & \text{summary;} \\ 0 & \text{possible summary;} \\ -1 & \text{non-summary.} \end{cases} \quad (1)$$

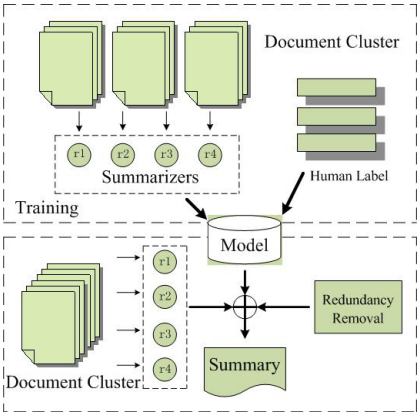

Figure 1: The framework of supervised aggregation for summarization.

Let $\mathscr{R} = \{r_1(\cdot), r_2(\cdot), ..., r_K(\cdot)\}$ denotes the set of K summarization methods and each method can produce a score list. The task of summarization aggregation is to combine the score lists

given by methods in $\mathscr{R}$ to produce better ranking results than any individual summarizer. In this study, we express the aggregation in a linear combining method and the combined ranking function can be denoted as

$$f(s) = \sum_{k=1}^{K} w_k r_k(s), \tag{2}$$

where w_k is the weight assigned to the kth individual summarization method. Thus, to learn the combination weights w_k $(k = 1, 2, ..., K)$ is pivotal in the aggregation method.

For simplicity, the aggregation score of sentence s_j^i can be rewritten in a matrix form:

$$f(s_j^i) = \mathbf{x}_j^i \mathbf{w}, \tag{3}$$

where $\mathbf{w} = [w_1, w_2, ...w_K]^T$ is the combination weights vector. $\mathbf{x}_j^i$ is a K-dimensional vector for representing the ranking scores computed by K different summarizers and denoted as

$$\mathbf{x}_j^i = [x_{j,1}^i, x_{j,2}^i, ..., x_{j,K}^i] = [r_1(s_j^i), r_2(s_j^i), ..., r_K(s_j^i)]. \tag{4}$$

3.2 Method description

In the supervised ranking aggregation method, we apply Ranking SVM (Joachims, 2002) directly. Ranking SVM trains the ranking model by decomposing a ranking list into the ordered pairs of items. In the document cluster c_i, given two sentences s_j^i and s_k^i with their score vectors $\mathbf{x}_j^i$ and $\mathbf{x}_k^i$, the training example can be built as the form $(\mathbf{x}_j^i - \mathbf{x}_k^i, z_{jk}^i)$ and the training label z_{jk}^i is defined as:

$$z_{jk}^i = \begin{cases} +1 & \text{if } s_j^i \succ s_k^i; \\ -1 & \text{if } s_k^i \succ s_j^i. \end{cases} \tag{5}$$

where $s_j^i \succ s_k^i$ denotes that sentence s_j^i is ranked higher than s_k^i. By defining the new training examples, the mathematical formulation of Ranking SVM is shown below, where the linear scoring function introduced in Formula (2) is used:

$$\begin{aligned} \min_{w} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C \sum_{i=1}^{M} \sum_{j=1}^{N_i} \sum_{k=j+1}^{N_i} \xi_{jk}^i \\ s.t. \quad & z_{jk}^i (\mathbf{w} \cdot (\mathbf{x}_j^i - \mathbf{x}_k^i)) \geq 1 - \xi_{jk}^i, i = 1, 2, ..., M, \\ & \xi_{jk}^i \geq 0, j < k \ \ and \ \ j, k \in [1, 2, ..., N_i]. \end{aligned} \tag{6}$$

where function complexity regularizer $\|\mathbf{w}\|^2$ is introduced to guarantee the generalization capacity and ξ_{jk}^i is the slack variables. C is a parameter that allows trading-off margin size against training error. Authors in (Hoi and Jin, 2008) pointed out that the above optimization problem has a drawback in training efficiency because the number of training pairs is quadratic of the number of items. Following their improved approach which can decrease the number of constraints significantly, we first build a relevance matrix A^i for ith document cluster to replace

above training labels, and the elements in the matrix are defined as follows.

$$A^i_{j,k} = \begin{cases} +1 & \text{if } l^i_j = 1 \text{ and } l^i_k = -1; \\ -1 & \text{if } l^i_j = -1 \text{ and } l^i_k = 1; \\ 1/2 & \text{if } l^i_j = 1 \text{ and } l^i_k = 0; \\ -1/2 & \text{if } l^i_j = 0 \text{ and } l^i_k = 1; \\ 0 & \text{otherwise.} \end{cases} \tag{7}$$

where l^i_j represents the label assigned to sentence s_j in the document cluster c_i and is defined in Formula (1).

Furthermore, a ranking matrix R is defined to denote the ranking results generated by the set of summarizers. For document cluster c_i, matrix $R^{i,j}$ stands for the ranking results output by summarizer $r_j(\cdot), j = 1, 2, ..., K$. Specifically, $R^{i,j}_{k,l} = 1$ if sentence s^i_k obtains higher score than s^i_l by $r_j(\cdot)$, and 0 otherwise. Next the matrix is normalized by a column-based normalization to be a transition matrix $\tilde{R}^{i,j}$, i.e. $\sum_{k=1}^{N_i} \tilde{R}^{i,j}_{k,l} = 1$.

After introducing the relevance matrix A and ranking matrix R, the original Ranking SVM shown in Formula (6) can be reformulated as:

$$\begin{aligned} \min_{w} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{M}\xi_i \\ s.t. \quad & sim([A^i]^T, [\sum_{j=1}^{K} w_j \tilde{R}^{i,j}]) \geq 1 - \xi_i, , i = 1, 2, ..., M, \\ & \xi^i_{jk} \geq 0, j < k \ \ and \ \ j, k \in [1, 2, ..., N_i]. \end{aligned} \tag{8}$$

where $sim(\cdot,\cdot)$ is a function that measures the similarity between two matrices: the transposition of relevance matrix A^i and the combined ranking matrix $\sum_{j=1}^{K} w_j \tilde{R}^{i,j}$. In the experiments, we follow (Hoi and Jin, 2008) by using trace function to measure the similarity of the two matrices. Compared with Formula (6), the number of constraints is significantly decreased from $\mathcal{O}(MN_i^2)$ to $\mathcal{O}(M)$.

4 Supervised summarization aggregation method implementation

4.1 Construction of training data

In order to apply supervised aggregation approach in summarization, we need to construct the training set in the form $\{(s^i_1, l^i_1), (s^i_2, l^i_2), ..., (s^i_{N_i}, l^i_{N_i})\}$ where s^i_j is the jth sentence in the document cluster c_i and l^i_j is the label assigned to the sentence. To capture the features contained in suboptimal sentences, we label sentences using three categories mentioned in Section 3.1: summary (+1), possible summary (0) and non-summary (-1).

Given a document cluster c which includes N sentences $\{s_1, s_2, ..., s_N\}$ and the corresponding human generated summary set $H = \{H_1, H_2, ..., H_m\}$ (H_i is the human summary generated by the ith linguist), we compute the score $score(s|H)$ for each sentence s in document cluster c to measure whether it can be chosen as the summary sentence. Motivated by ROUGE evaluation methods (Lin and Hovy, 2003), our scoring methods compute the combination of

multiple n-grams (1-gram and 2-gram are used in this study) probabilities of each sentence to be recognized as a summary sentence based on the human summary set.

First we compute the probability of an n-gram t under a human summary H_i as:

$$p(t|H_i) = tf(t)/|H_i|, \tag{9}$$

where $tf(t)$ is the frequency of t in H_i and $|H_i|$ is the number of n-grams (n is corresponding to the length of t) in H_i. In the data set, several human summaries are provided for each document cluster, and both average and maximum schemes introduced in (Ouyang et al., 2007) can be utilized[1]. The average scheme to obtain the probability of t under all human summaries is described as:

$$p_{avg}(t|H) = \sum_{H_i \in H} p(t|H_i)/|H|, \tag{10}$$

where $|H|$ is the number of summaries in the human summary set. And for maximum scheme the computation method is defined as:

$$p_{max}(t|H) = \max_{H_i \in H} p(t|H_i) \tag{11}$$

Motivated by the ROUGE evaluation metrics, we take into account both 1-gram and 2-gram to calculate the final score for each sentence in the following formula.

$$Score(s|H) = \alpha \sum_{t_{1-gram} \in s} p(t_{1-gram}|H) + (1-\alpha) \sum_{t_{2-gram} \in s} p(t_{2-gram}|H), \tag{12}$$

where α is used to control the ratios of these two types of n-grams in computing scores. Since 1-gram based ROUGE score has been shown to agree with human judgment most (Lin and Hovy, 2003), in our experiments α is set empirically to be 0.7.

4.2 Individual summarization methods

To evaluate the proposed supervised aggregation method for summarization, we introduce four typical summarization approaches (i.e., LexPageRank, LexHITS. Manifold-ranking, and DivRank) in the system implementation. In this section, we briefly describe these approaches which all have been proved effective in summarization task.

1. LexPageRank
   LexPageRank (Erkan and Radev, 2004) is a graph-based summarization method by introducing PageRank into summarization, which computes sentence scores by making use of the voting or recommendations between sentences. Sentences are used as nodes in the graph and the computational process can be described as:

$$PR(s_i) = \lambda \cdot \sum_{j:j \neq i} PR(s_j) \cdot w_{ji} + \frac{(1-\lambda)}{|S|}, \tag{13}$$

   where $PR(s_i)$ is the score of sentence s_i and $|S|$ denotes the number of sentences in a document cluster. w_{ji} represents the weight (e.g., cosine similarity) between sentence s_j and s_i and λ is the damping factor to control the probability to walk to a random sentence.

[1]The experimental comparison of two schemes will be described in Section 5.3.2

2. LexHITS
   Similar to LexPageRank, HITS (Kleinberg, 1999) algorithm can be applied in summarization task as well and correspondingly is named LexHITS. In LexHITS, sentences denote both authority nodes and hub nodes and the iteration process is written as follows.

$$\begin{aligned} Auth^{(T+1)}(s_i) &= \sum_{j:j\neq i} w_{ij} \cdot Hub^{(T)}(s_j) \\ Hub^{(T+1)}(s_j) &= \sum_{i:i\neq j} w_{ji} \cdot Auth^{(T)}(s_i) \end{aligned}, \tag{14}$$

   where $Auth^{(T)}(s_i)$ and $Hub^{(T)}(s_i)$ represent the authority score and hub score of sentence s_i at the Tth iteration, respectively. w_{ij} denotes the weight between sentence s_i and s_j same as Formula (13). Then sentences are ranked according to their authority scores.

3. Manifold
   Manifold-ranking based method (Zhou et al., 2004) is a universal ranking algorithm and can capture the underlying manifold structure of data. Authors in (Wan et al., 2007) used manifold-ranking approach for summarization. First the similarity matrix W is built and the element w_{ij} denotes the weight between sentence s_i and s_j like the settings in LexPageRank and LexHITS. Then normalize W by $\tilde{W} = D^{-1/2}WD^{-1/2}$ in which D is the diagonal matrix with (i,i)-element equal to the sum of the ith row of W. By incorporating a n-dimensional vector y, the score of each sentence can be iterated as:

$$MF^{(T+1)}(s_i) = \alpha \sum_{j:j\neq i} \tilde{w}_{ij} MF^{(T)}(s_j) + (1-\alpha)y_i, \tag{15}$$

   where $MF^{(T)}(s_i)$ denotes the manifold-ranking score of sentence s_i at the Tth iteration and $\tilde{w}_{ij}$ is the element in the normalized matrix $\tilde{W}$. α is a parameter between 0 and 1.

4. DivRank
   DivRank (Mei et al., 2010) belongs to the time-variant random walk process family which incorporates a variable to record the number of times of nodes having been visited. DivRank can balance prestige and diversity simultaneously by decreasing the visiting times of certain nodes. The iteration process of DivRank can be described as follows:

$$DR^{T+1}(s_i) = (1-\lambda)p^*(s_i) + \lambda \sum_{j:j\neq i} \frac{w_{ji} \cdot N^T(s_i)}{D^T(s_j)} DR^T(s_j), \tag{16}$$

   where $DR^T(s_i)$ denotes the DivRank score of sentence s_i at the Tth iteration and $p^*(s_i)$ represents the prior value of sentence s_i. $N^T(s_i)$ is the number of times the walk has visited s_i up to time T and $D^T(s_j) = \sum_{k:k\neq j} w_{kj} N^T(s_k)$. Similarly, w_{ji} is the weight between sentence s_j and s_i and denotes the transition probability from s_j to s_i.

4.3 Aggregation methods

Aiming to compare the proposed supervised aggregation method with other aggregation methods, we implement several aggregation methods using different ensemble strategies in the experimental studies including average scores, Round Robin (RR), unsupervised learning algorithm for rank aggregation (ULARA), and Weighted consensus summarization (WCS). A brief description of these aggregation methods is presented in this section.

1. Average Score (Avg_Score)
   This method normalizes the raw scores generated by different summarization systems between 0 and 1, and then uses the average score $Avg\_Score(s) = \frac{\sum_{k=1}^{K} score_k(s)}{K}$ to rank the sentences. In the formula, K is the number of summarization systems and $score_k(s)$ is the score of sentence s from kth summarization system.

2. Round Robin (RR)
   Refer to (Wang and Li, 2011), RR chooses the first sentence produced by the first summarizer and then the first sentence by the second summarizer. After all the first sentence are selected in the first round, the second round chooses the second sentences in the same way until reaching the summary length limit.

3. ULARA
   Unsupervised learning algorithm for rank aggregation (ULARA) is proposed in (Klementiev et al., 2007). ULARA applied a linear combination of the individual ranking approaches to form the aggregation result by rewarding ordering agreement between different rankers. By minimizing the weighted variance-like measures, the optimal weights assigned to component rankers are obtained.

4. Weighted Consensus Summarization (WCS)
   WCS algorithm (Wang and Li, 2011) utilizes a weighted consensus scheme to combine the results from individual summarizers. In this algorithm, the contribution from each summarization system is determined by its agreement with other systems. By minimizing the weighted distance between the consensus ranking and the individual ranking lists generated by different summarization systems, the weights that will be assigned to individual summarizers are obtained.

5. Supervised Summarization Aggregation Method (SSA)
   SSA is the supervised aggregation method for summarization described in Section 3.

In the experiments, we study the summarization performance of the implemented individual and aggregation systems, and compare the proposed supervised summarization aggregation method with other combination methods.

4.4 Redundancy removal

In order to choose more informative but less redundant sentences as the final summary, a redundancy removal step is conducted to impose the diversity penalty. In the experiments, we use the diversity penalty algorithm proposed in (Wan et al., 2007) to remove redundant sentences by introducing a penalty degree factor ω. The algorithm is described briefly in Algorithm 1.

5 Experiments

5.1 Data set

To evaluate the summarization results empirically, we use DUC2004[2] data set since generic multi-document summarization is one of the fundamental tasks in DUC2004. The data set

[2]http://duc.nist.gov/

Algorithm 1 Redundancy Removal Algorithm

1: Initialize set $S_A = \emptyset$, $S_B = \{s_1, s_2, ..., s_n\}$, and every sentence s_i in set S_B has a score $score(s_i)$ which initially is computed by the score function $r(\cdot)$;
2: Sort sentences in set S_B according to their scores in descending order;
3: Choose sentence s^* with the highest score in set S_B and move it from S_B to S_A;
4: **for** $s_j \in S_B$ **do**
5: $score(s_j) = score(s_j) - \omega \cdot w_{ji} \cdot r(s_i)$
6: **end for**
7: Go to step 2 and iterate until S_A reaches the length limit of a summary or $S_B = \emptyset$.

provides 50 document clusters and every generated summary is limited to 665 bytes. In the experiments, we randomly choose 40 document clusters as training data and the remaining 10 clusters are used as the testing data. For consistency, all the evaluation results are based on the testing set.

5.2 Evaluation methods

ROUGE (Lin and Hovy, 2003) (Recall Oriented Understudy for Gisting Evaluation) is widely applied for summarization evaluation by DUC. Therefore, we use the ROUGE toolkit[3] to evaluate the summarization results. It evaluates the quality of a summary by counting the overlapping units between the candidate summary and model summaries. ROUGE implements multiple evaluation metrics to measure the system-generated summarization such as ROUGE-N, ROUGE-L, ROUGE-W and ROUGE-SU. ROUGE-N is an n-gram recall measure computed as follows:

$$ROUGE-N = \frac{\sum_{S \in ref} \sum_{n-gram \in S} Count_{match}(n-gram)}{\sum_{S \in ref} \sum_{n-gram \in S} Count(n-gram)} \quad (17)$$

where n represents the length of n-gram, and $Count_{match}(n-gram)$ is the maximum number of n-grams co-occurring in the candidate summary and reference summaries. $Count(n-gram)$ is the number of n-grams in the reference summaries.

The ROUGE toolkit can report separate scores for 1, 2, 3 and 4-gram and among these different metrics, unigram-based ROUGE score (ROUGE-1) has been shown to correlate well with human judgments. Besides, longest common subsequence (LCS), weighted LCS and skip-bigram co-occurrences statistics are also used in ROUGE. ROUGE can generate three scores, i.e. recall, precision and F-measure, for each of the methods. In the experimental results we show three of the ROUGE metrics: ROUGE-1 (unigram-based), ROUGE-2 (bigram-based), and ROUGESU4 (extension of ROUGE-S, which is the skip-bigram co-occurrences statistics) metrics.

5.3 Evaluation results

5.3.1 Performance comparison

The proposed supervised summarization aggregation method is compared with different aggregation schemes including average score, round robin, ULARA, and WCS which are introduced in Section 4.3. In order to analyze the improvement of the aggregation method, we also list the

[3]ROUGE version 1.5.5 is used in this study, and it can be found on the website http://www.isi.edu/licensed-sw/see/rouge/

performance of all the individual methods. Besides, we also use Lead method as the baseline. The lead baseline takes the first sentences one by one in the last document in a document set, where documents are assumed to be ordered chronologically. Table 1 shows the comparison results (F-measure) on DUC2004 data set in ROUGE-1, ROUGE-2 and ROUGE-SU4 along with corresponding 95% confidence intervals and Figure 2 gives an illustration of the comparison on ROUGE-1 metric (LexPR is short for LexPageRank shown in the figure).

Systems	ROUGE-1	ROUGE-2	ROUGE-SU4
Lead	0.31861 (0.30886 - 0.32820)	0.06814 (0.06102 - 0.07631)	0.10554 (0.09953 - 0.11208)
LexPageRank	0.36211 (0.35081 - 0.37384)	0.07808 (0.07027 – 0.08675)	0.11982 (0.11311 - 0.12717)
LexHITS	0.35285 (0.33981 - 0.36551)	0.06911 (0.06137 – 0.07675)	0.11485 (0.10771 - 0.12233)
Manifold	0.37809 (0.36785 - 0.38866)	0.08046 (0.07220 – 0.08890)	0.12577 (0.11952 - 0.13224)
DivRank	0.37442 (0.36076 - 0.38693)	0.08255 (0.07400 – 0.09058)	0.12503 (0.11721 - 0.13248)
Avg_Score	0.37814 (0.36716 - 0.38914)	0.08690 (0.07900 – 0.09520)	0.12823 (0.12155 - 0.13490)
RR	0.36809 (0.35489 - 0.38028)	0.08095 (0.07255 – 0.08984)	0.12412 (0.11672 - 0.13141)
ULARA	0.37971 (0.36720 - 0.39183)	0.09010 (0.08186 - 0.09837)	0.13163 (0.12399 - 0.13880)
WCS	0.38227 (0.37019 - 0.39334)	0.09133 (0.06669 - 0.11785)	0.13285 (0.11178 - 0.15798)
SSA	**0.39766** (0.36761 - 0.42835)	**0.09528** (0.07186 - 0.12093)	**0.13939** (0.11906 - 0.16217)

Table 1: Overall performance comparison on DUC2004

From the comparison results, it can be seen that the proposed supervised summarization aggregation (SSA) method can outperform all the other ensemble methods and individual summarization approaches on all the three metrices. This comparison indicates that by incorporating human labeled data, supervised aggregation method has an advantage over unsupervised ensemble methods, i.e., when identifying the reliability of score list from a single summarizer, human labeled data could serve as an precise guidance. Moreover, almost all the aggregation methods perform better than individual summarization systems except the round robin method and this result is consistent with the comparison in (Wang and Li, 2011). For one thing, the results demonstrate that generally ensemble methods can effectively enhance the performance since these methods make the best of individual summarization methods which rank sentences from different aspects. For another, the poor performance of round robin method may result from that simply choosing the sentence with highest score in every round ignores the relationship among different score lists and the inaccuracy or overlap of the top sentences can lead to the poor effect as well.

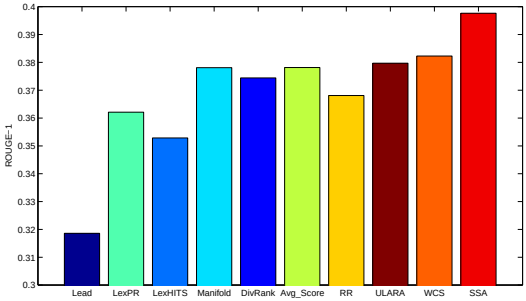

Figure 2: The comparison results of all the methods.

5.3.2 Influence of training data construction schemes

As mentioned in Section 4.1, there are two schemes to construct the training data (i.e., *average* and *maximum*). Average scheme chooses the average probability of an n-gram under all human summaries as the measure and maximum scheme applies the maximum probability value to represent the probability of an n-gram under all human summaries.

From the comparison shown in Table 2, we observe that the maximum scheme performs better than the average one. By analyzing the definition of two schemes, *maximum* tends to assign sentences with a higher value when compared with the *average* scheme and therefore it can choose more potential summary sentences into positive training set and produce better results.

Schemes	ROUGE-1	ROUGE-2	ROUGE-SU4
maximum	0.39766	0.09528	0.13939
average	0.39394	0.09306	0.13859

Table 2: Results of different schemes on DUC2004

5.3.3 Influence of component diversity

In the experimental study, we exploit four different summarization methods which are proven effective in summarization and they rank sentences from different aspects by utilizing diverse strategies. Therefore differences in algorithms and implementation make the ensemble process can comprehensively take into consideration multiple ranking strategies.

The LexPageRank summarization method scores sentences by making use of the voting or recommendations between sentences, and thus the global information of all the sentences can be in full use. LexHITS method assigns each sentence two properties, i.e., hub and authority, which can take into account the mutual relationship between sentences and provide a better view of the relationships embedded in the sentences. Manifold-ranking based method is based on a universal ranking algorithm and can capture the underlying manifold structure of data,

thus some implicit relationships between sentences are exploited. DivRank can be regarded as an expand version of LexPageRank which incorporates a variable to record the number of times of nodes having been visited. Through this improvement, the diversity and prestige of sentences to be chosen can be guaranteed simultaneously.

In this set of experiments, we further investigate the influence of different combinations of component methods[4]. We use the proposed framework to combine any three of all summarization methods and compare the results with the combination scheme using all the four methods. Table 3 shows the comparison results.

Systems	ROUGE-1	ROUGE-2	ROUGE-SU4
LH+MF+DR	0.39394	0.09306	0.13859
LPR+MF+DR	0.39393	0.09397	0.13834
LPR+LH+DR	0.38829	0.08172	0.12807
LPR+LH+MF	0.39383	0.09355	0.13916
All	0.39766	0.09528	0.13939

Table 3: Comparison results of different component combinations on DUC2004

From the table, it can be seen that different component summarization methods have less impact on the results in general and this may due to the small number of testing data. However, from the little fluctuation we can observe that aggregating all the individual methods can perform the best results. It is worth mentioning that the performance of LPR+LH+DR is relatively poor, and this result may owe to losing the Manifold-ranking based method which performs best among all the four methods.

Conclusion and perspectives

In this paper, we propose a supervised aggregation summarization framework by combining the results from four typical multi-document systems including LexPageRank, LexHITS, Manifold-ranking method and DivRank. To evaluate the proposed appraoch, we compare it with several combination methods, e.g., average score, round robin, unsupervised learning algorithm rank aggregation (ULARA) and weighted consensus summarization (WCS). And the experimental results on DUC 2004 data set demonstrate the effectiveness of our proposed framework. In addition, irrespective of the specific individual summarization methods used in this study, the supervised aggregation framework for summarization can also incorporate some more delicate and effective summarizers and generate more promising summary.

In this study, we investigate the supervised summarization aggregation approach to learn weights automatically by incorporating human labeled data. Since labeled sentences would be time-consuming and costly, in the future we will explore semi-supervised method which can decrease the amount of labeled data required. Moreover, more effective individual summarization approaches would be exploited and added into the ensemble method.

Acknowledgments

This work is supported by NSFC under the grant No. 60933004 and 61073082. We also would like to thank the reviewers for their useful comments.

[4]LH, LPR, MF and DR are short for LexHITS, LexPageRank, Manifold and DivRank, respectively.

References

Celikyilmaz, A. and Hakkani-Tur, D. (2010). A hybrid hierarchical model for multi-document summarization. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 815–824. Association for Computational Linguistics.

Chen, S., Wang, F., Song, Y., and Zhang, C. (2011). Semi-supervised ranking aggregation. *Information Processing & Management*, 47(3):415–425.

Erkan, G. and Radev, D. (2004). Lexpagerank: Prestige in multi-document text summarization. In *Proceedings of EMNLP*, volume 4.

Haghighi, A. and Vanderwende, L. (2009). Exploring content models for multi-document summarization. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 362–370. Association for Computational Linguistics.

Hahn, U. and Mani, I. (2000). The challenges of automatic summarization. *Computer*, 33(11):29–36.

Hoi, S. and Jin, R. (2008). Semi-supervised ensemble ranking. In *Proceedings of the 23rd national conference on Artificial intelligence*, volume 2.

Jin, F., Huang, M., and Zhu, X. (2010). A comparative study on ranking and selection strategies for multi-document summarization. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 525–533. Association for Computational Linguistics.

Joachims, T. (2002). Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 133–142. ACM.

Kleinberg, J. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5):604–632.

Klementiev, A., Roth, D., and Small, K. (2007). An unsupervised learning algorithm for rank aggregation. *Machine Learning: ECML 2007*, pages 616–623.

Li, H. (2011). Learning to rank for information retrieval and natural language processing. *Synthesis Lectures on Human Language Technologies*, 4(1):1–113.

Li, L., Zhou, K., Xue, G., Zha, H., and Yu, Y. (2009). Enhancing diversity, coverage and balance for summarization through structure learning. In *Proceedings of the 18th international conference on World wide web*, pages 71–80. ACM.

Lin, C. and Hovy, E. (2002). From single to multi-document summarization: A prototype system and its evaluation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 457–464. Association for Computational Linguistics.

Lin, C. and Hovy, E. (2003). Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 71–78. Association for Computational Linguistics.

Liu, Y., Liu, T., Qin, T., Ma, Z., and Li, H. (2007). Supervised rank aggregation. In *Proceedings of the 16th international conference on World Wide Web*, pages 481–490. ACM.

Mei, Q., Guo, J., and Radev, D. (2010). Divrank: the interplay of prestige and diversity in information networks. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1009–1018. ACM.

Mihalcea, R. and Tarau, P. (2004). Textrank: Bringing order into texts. In *Proceedings of EMNLP*, volume 4. Barcelona: ACL.

Nenkova, A., Vanderwende, L., and McKeown, K. (2006). A compositional context sensitive multi-document summarizer: exploring the factors that influence summarization. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 573–580. ACM.

Ouyang, Y., Li, S., and Li, W. (2007). Developing learning strategies for topic-based summarization. In *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 79–86. ACM.

Ouyang, Y., Li, W., Lu, Q., and Zhang, R. (2010). A study on position information in document summarization. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 919–927. Association for Computational Linguistics.

Page, L., Brin, S., Motwani, R., and Winograd, T. (1999). The pagerank citation ranking: Bringing order to the web.

Radev, D., Jing, H., Stys, M., and Tam, D. (2004). Centroid-based summarization of multiple documents. *Information Processing & Management*, 40(6):919–938.

Shen, C. and Li, T. (2011). Learning to rank for query-focused multi-document summarization. In *Data Mining (ICDM), 2011 IEEE 11th International Conference on*, pages 626–634. IEEE.

Shen, D., Sun, J., Li, H., Yang, Q., and Chen, Z. (2007). Document summarization using conditional random fields. In *Proceedings of IJCAI*, volume 7, pages 2862–2867.

Svore, K., Vanderwende, L., and Burges, C. (2007). Enhancing single-document summarization by combining ranknet and third-party sources. In *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*, pages 448–457.

Tang, J., Yao, L., and Chen, D. (2009). Multi-topic based query-oriented summarization. In *Proceedings of SDM*, volume 9.

Van, M. and Erp, S. (2000). L.: Variants of the borda count method for combining ranked classifier hypotheses. In *in the Seventh International Workshop on Frontiers in Handwriting Recognition*, pages 443–452.

Wan, X. and Xiao, J. (2009). Graph-based multi-modality learning for topic-focused multi-document summarization. In *Proceedings of the 21st international jont conference on Artifical intelligence (IJCAI-09)*, pages 1586–1591.

Wan, X. and Yang, J. (2008). Multi-document summarization using cluster-based link analysis. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 299–306. ACM.

Wan, X., Yang, J., and Xiao, J. (2007). Manifold-ranking based topic-focused multi-document summarization. In *Proceedings of IJCAI*, volume 7, pages 2903–2908.

Wang, D. and Li, T. (2010). Many are better than one: improving multi-document summarization via weighted consensus. In *Proceeding of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 809–810. ACM.

Wang, D. and Li, T. (2011). Weighted consensus multi-document summarization. *Information Processing & Management*.

Wei, F., Li, W., and He, Y. (2009). Co-feedback ranking for query-focused summarization. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 117–120. Association for Computational Linguistics.

Wong, K., Wu, M., and Li, W. (2008). Extractive summarization using supervised and semi-supervised learning. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 985–992. Association for Computational Linguistics.

Zhou, D., Weston, J., Gretton, A., Bousquet, O., and Scholkopf, B. (2004). Ranking on data manifolds. *Advances in neural information processing systems*, 16:169–176.