AdaSum: An Adaptive Model for Summarization

Jin Zhang
zhangjin@software.ict.ac.cn

Xueqi Cheng
cxq@ict.ac.cn

Gaowei Wu
wgw@ict.ac.cn

Hongbo Xu
hbxu@software.ict.ac.cn

Institute of Computing Technology, Chinese Academy of Sciences
NO. 6, South Road, Kexueyuan, Zhongguancun,
Beijing, P.R.China

ABSTRACT

Topic representation mismatch is a key problem in topic-oriented summarization for the specified topic is usually too short to understand/interpret. This paper proposes a novel adaptive model for summarization, AdaSum, under the assumption that the summary and the topic representation can be mutually boosted. AdaSum aims to simultaneously optimize the topic representation and extract effective summaries. This model employs a mutual boosting process to minimize the topic representation mismatch for base summarizers. Furthermore, a linear combination of base summarizers is proposed to further reduce the topic representation mismatch from the diversity of base summarizers with a general learning framework. We prove that the training process of AdaSum can enhance the performance measure used. Experimental results on DUC 2007 dataset show that AdaSum significantly outperforms the baseline methods for summarization (e.g. MRP, LexRank, and GSPS).

Categories and Subject Descriptors

H.3.1. [**Content Analysis and Indexing**]: Abstracting Methods; I.2.7. [**Natural Language Processing**]: Text Analysis

General Terms

Algorithms, Performance, Experimentation.

Keywords

Topic-oriented Summarization, Topic Representation, Boosting, AdaSum

1. INTRODUCTION

Automated summarization usually deals with concatenating text-span excerpts (i.e., sentences, paragraphs, etc.) into a human understandable document summary and it dates back to the 1950's [17]. Along with the fast growth of information amounts on the internet, a recent need for multi-document summarization has also been exerted and thus several algorithms have been proposed. Most of the researchers have concentrated on the topic-oriented sentence-extraction summarization methods [19, 14, 18]. Most of former approaches are based on the assumption that specified topics are reliable enough to be used as "gold standards" for automatic summarization. However, the specified topic is usually too simple to understand/interpret, and there exists topic representation mismatch between the specified topic and the expected topic.

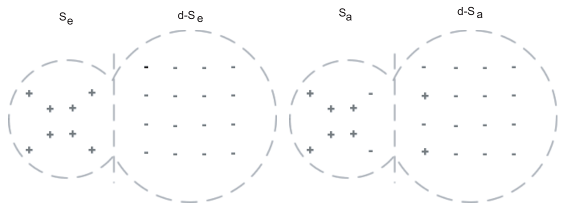
To address this challenge, several methods for topic representation have been developed and applied to topic-oriented summarization [16, 2, 11]. In order to alleviate the topic representation mismatch problem in summarization, most previous approaches recognized related terms by term clustering. As first proposed in [16], Lin et al. represented the topic of a document (or a document collection) by using a set of terms, known as topic signatures (TS). Barzilay et al. [2] proposed the use of content models to capture constraints on topic selection and organization for texts in a particular domain. In recent research, Harabagiu and Lacatusu [11] represented topics as a structure of themes, where a theme is defined as a cluster of sentences from different documents with similar semantic information.

In this paper, we aim to develop a novel summarization algorithm that can optimize any performance measures used through topic representation optimization. Inspired by the work of AdaBoost for classification [8] and AdaRank for information retrieval [22], we propose an adaptive model for topic-oriented multi-document summarization, referred to as AdaSum. Normally, the generated summary represents the estimation of the expected topic in the hypothesis space (document collections), we assume that the summary and the topic representation can be mutually boosted. In order to minimize the expected risk, AdaSum utilizes a linear combination of base summarizers as its model. During the process of summarization, it repeats the process of re-weighting the generated summaries, creating a base summarizer, and calculating a weight for the summary.

We show that AdaSum algorithm can optimize topic representation and generate more effective summaries. The proposed approach is evaluated on the DUC 2007 dataset and the results show that the effectiveness of the proposed approach for both topic representation and document summarization. With comparison to the baseline summarizers, AdaSum shows significantly improvement in performance measures. Moreover, in comparison with the top performing state-of-the-art systems in DUC 2007, our proposed model is also competitive.

The remainder of the paper is organized as follows. Section 2 motivates the need for topic interpretation and representation with a mutual boosting process. The proposed AdaSum is described in

CIKM'08, October 26–30, 2008, Napa Valley, California, USA.



(a) An expected summary (b) An automatic summary

Figure 1: Algorithm bias of summarization for topic representation mismatch, where $\mathbf{d}$ is the document collection to be summarized, S_e is the expected summary of $\mathbf{d}$, and S_a is an automatic summary of a base summarizer. And symbols plus $(+)$ and minus $(-)$ stand for the target sentences with summary state and the another target sentences with non-summary state respectively.

detail in Section 3. We set up the experiments in Section 4 and give the results and discussions in Section 5. After a summary of related work in Section 6, we conclude this paper and give future work in Section 7.

2. FORMALISM PRELIMINARIES

As a particular kind of multi-document summarization, topic-oriented multi-document summarization aims to produce a summary with respect to a specified topic. Topic-oriented multi-document summarization is faced with more challenges than general multi-document summarization, that the information in the generated summary must represent the specified topic. Unfortunately, the specified topic is usually too simple to understand, and this problem causes the topic representation mismatch in summarization, the same as in IR and QA. For this reason, many current summarization algorithms can only partially represent aspects of the expected topic. Thus, there usually exists bias between the expected topic and the estimated topic. In this paper, we call the above bias as algorithm bias, for the mismatch is caused by the diversity of summarization algorithms.

For topic-oriented summarization, the expected summary should represent all aspects of the expected topic. It is a condensed representation for the expected topic on the summary space. Given a document collection $\mathbf{d}$ and the original topic t_0, the expected topic, t_e, can be represented by the expectation on the summary space, S_e. Then, the task of summarization is to generate automatic summaries with less difference against the expected summary (or manual summary). Suppose there are two kinds of states, where one kind corresponds to the summary state and the other corresponds to non-summary state. Let $\Gamma^+(s)$ be the set of sentences with summary state, and $\Gamma^-(s)$ be the set of sentences with non-summary state. Figure 1(a) is an expectation summary for the specified topic, symbols plus $(+)$ and minus $(-)$ stand for target sentences with summary state and sentences with non-summary state. S_e is the set of sentences with summary state, and $\mathbf{d} - S_e$ is the set of sentences with non-summary state in the document collection $\mathbf{d}$. As Figure 1(a) shows, the expected summary should include all the sentences from $\Gamma^+(s)$ with excluding every sentence from $\Gamma^-(s)$. Unfortunately, with the topic representation mismatch in current summarization algorithms, the derived summary could inevitably contain sentences from $\Gamma^-(s)$. For example, as shows in Figure 1(b), there are two sentences with non-summary state in the estimated summary S_a, while two sentences with summary state is misplaced in $\mathbf{d} - S_a$.

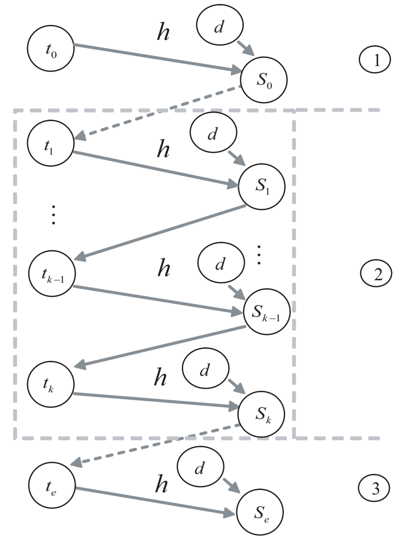


Figure 2: The process of mutual boosting for summarization, where $\mathbf{d}$ is the document collection to be summarized, t_0 and t_e are the original specified topic and the expected topic respectively. In order to alleviate the bias between t_0 and t_e, a feedback process is proposed to optimize the topic representation, where h is the base summarizer.

In order to alleviate the topic mismatch problem for summarization, we need effective summarization methods to represent as many aspects of the expected topic as possible. That is, in order to optimize the summarization performance, it is essential to minimize the expected bias between the estimated topic and the expected topic. For topic-oriented document summarization, the expected summary should be consistent with the topic representation of the specified topic. Thus, there exists a mutual reinforcement between the specified topic and the derived summary. With the mutual reinforcement between the expected topic and the derived summaries, we can assume that the summary and the topic representation can be mutually boosted, where the derived summary depends on the expected topic and the expected topic can be represented with the derived summary. More precisely, we can optimize the estimation of topic representation with a mutual boosting process in the hypothesis space.

3. ADAPTIVE MODEL FOR SUMMARIZATION

In this paper, we aim to develop a novel summarization algorithm that can directly optimize any performance measures used in document summarization. Intuitively, a good topic representation is benefit to derive a high performance summary, and a high performance summary can more effectively represent the expected topic. Therefore, there exists a reinforcement between the estimated topic and the derived summary. And with making use of this reinforcement, estimated topics and derived summaries can be mutually boosted.

As Figure 2 shows, a novel adaptive summarization model can be formulated as a process of mutual boosting, where the k^{th} estimated topic t_k is related to the estimated summary S_{k-1} corresponding to its previous topic t_{k-1} and the base summarizer h[1].

[1]The estimated summary $S_{k-1} = h(t_{k-1}, \mathbf{d})$.

Thus, the expected topic t_e can be represented by k rounds' feedback of estimated summaries, and the expected summary S_e can be approximated, relative to t_e (the third part in Figure 2). Normally, as a form of feedback, the topic representation can be optimized with the topic mismatch as small as possible. Consequently, the performance of base summarizer can be improved with the adaptive process for topic representation.

However, as for the diversity of base summarizers, a single base summarizer usually biases to represent some aspects of the topic representation. In order to further improve the diversity of base summarizers, a linear combination of base summarizers can be employed. In that, the process in the second part of Figure 2 can also be formulated as a linear combination of adaptive models for topic representation, where each adaptive model for topic representation corresponds to a base summarizer. Inspired by the boosting algorithms for classification and IR, AdaBoost and AdaRank, we have proposed a novel adaptive model (AdaSum) for topic-oriented summarization. AdaSum utilizes a linear combination of base summarizers as its model. As to minimize the expected bias between the expected topic and the estimated topic, a general learning framework is employed based on the performance measures of summarization.

Consequently, the process of summarization consists of two steps: (1) a mutual boosting process is employed to derive a new summarizer which can represent the expected topic as good as possible, so as to optimize the topic representation and minimize the expected bias between the estimated topic and the expected topic (the second part in Figure 2); (2) a linear combination of base summarizers is employed to obtain a new summarizer which can boost the performance of derived summaries with a general learning framework.

In the following subsections, we present the design for two steps of our approach, namely adaptive model for topic representation and general learning framework for summarization.

3.1 Adaptive Model for Topic Representation

With the assumption of mutual boosting between topics and summaries, we first utilize an adaptive boosting process for improving summarization performance with topic representation optimization. The goal is to find the optimized topic representation t_k so that the estimated topic is as much as possible similar to the expected topic representation. It is difficult to directly measure the similarity between the estimated topic and the expected topic, because the expected topic is inconvenient to obtain directly. We instead attempt to maximize the similarity between the estimated topic and the original topic in Equation 1.

$$t_k = \arg\max_t \mathrm{sim}(\phi(h(t, \mathbf{d})), \phi(t_0)) \tag{1}$$

where t is the estimated topic, t_k is the optimized topic, t_0 is the original topic for the document collection $\mathbf{d}$, h is a base summarizer, and $\phi(\cdot)$ is the topic representation (e.g. terms' distribution of topics).

The adaptive model for topic representation is showed in Figure 3, where t_0 is the original topic for the document collection $\mathbf{d}$, and t_e is the corresponding target expected topic. Given a specified base summarizer h, with the adaptive boosting process for topic representation, an estimated topic t_i is derived from the previous topic t_{i-1} and its corresponding estimated summary $h(t_{i-1}, \mathbf{d})$. Then, the adaptive boosting process can be formulated a linear combination as follows:

$$\phi(t_i) = (1 - \lambda_i)\phi(t_{i-1}) + \lambda_i \phi(h(t_{i-1}, \mathbf{d})) \tag{2}$$

where $\lambda_i = \mathrm{sim}(\phi(h(t_{i-1}, \mathbf{d})), \phi(t_0)) - \mathrm{sim}(\phi(t_{i-1}), \phi(t_0)) (i > 1)$ and $\lambda_1 = \mathrm{sim}(\phi(h(t_0, \mathbf{d})), \phi(t_0))$. λ_i is used to evaluate the similarity improvement between the estimated topic and the original topic. Specifically, when there is no improvement in similarity between the estimated topic and the original topic, the adaptive boosting process is automatically terminated.

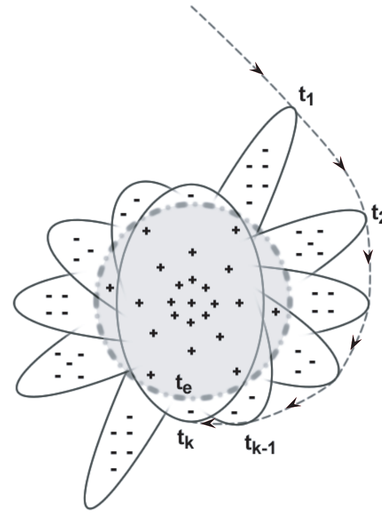


Figure 3: The adaptive model for topic representation, where the dashed circle is the target expected topic for t_0, and $t_1, t_2, ..., t_{k-1}, t_k$ are the estimated topics. The estimated topic for the i^th round is obtain from the previous estimated topic t_{i-1} and the base summarizer h. With the adaptive boosting process for topic representation, the estimated topic has minimum difference to the expected summary.

3.2 Learning Framework for Summarization

In order to obtain a new summarizer to boost the performance of derived summaries, we define the general framework of learning for summarization.

In summarizing (testing), given a document collection $\mathbf{d}$ and topic t, the system returns a set of sentences, which are top sentences in the document collection $\mathbf{d}$ according to descending order of the relevance scores to the topic t. The relevance scores are calculated with a summarizing function (model). In learning (training), a number of document collections with specified topics and their corresponding summarizations (sets of retrieved sentences from document collections) are given. The objective of learning is to construct a summarizing function which achieves the best results in summarizing of the training data in the sense of minimization of a loss function. Ideally the loss function is defined on the basis of the performance measures used in testing.

In training, a set of document collections $\mathbf{D} = \{\mathbf{d}_1, \mathbf{d}_2, \cdots, \mathbf{d}_m\}$ is given. $\mathbf{d}_i$ can be described as a set of sentences, i.e. $\mathbf{d}_i = \{s_{i1}, s_{i2}, \cdots, s_{i,n(\mathbf{d}_i)}\}$, where s_{ij} denotes the j^{th} sentence in $\mathbf{d}_i$, $n(\mathbf{d}_i)$ denotes the size of document collection $\mathbf{d}_i$. Each document collection $\mathbf{d}_i$ is associated with a specified topic t^i and a list of labels $\mathbf{y}_i = \{y_{i1}, y_{i2}, \cdots, y_{i,n(\mathbf{d}_i)}\}$, where $y_{ij} \in \mathcal{Y}$ denotes the summary state of sentence s_{ij}, i.e., whether s_{ij} belongs to the manual summary given by humans. Let $\mathcal{Y} = \{0, 1\}$, then $y_{ij} = 1$ stands for the target sentence s_{ij} is with summary state, otherwise with non-summary state.

A feature vector $\mathbf{x}_{ij} = \mathbf{\Psi}(s_{ij}, t^i) \in \mathcal{X}$ is created from each sentence-topic pair (s_{ij}, t^i), $i = 1, 2, \ldots, m$, $j = 1, 2, \ldots, n(\mathbf{d}_i)$. Thus, the training set can be represented as $S = \{(\mathbf{d}_i, t^i, \mathbf{y}_i)\}_{i=1}^m$.

The objective of learning is to create a summarizing function $f : \mathcal{X} \mapsto \mathcal{R}$, such that, for each document collection $\mathbf{d}$ and its corresponding specified topic t, the sentences in $\mathbf{d}$ can be assigned relevance scores using the function f and be ranked according to the scores. Then top sentences are picked out as topic-oriented summarization for $\mathbf{d}$ and t. We specially define a permutation mapping $\pi(\mathbf{d}_i, t^i, f)$ for document collection $\mathbf{d}_i$, the corresponding topic t^i, and the summarizing function. Let $\mathbf{d}_i = \{s_{i1}, \dots, s_{i,n(\mathbf{d}_i)}\}$ be identified by the indexing set $I = \{1, \dots, n(\mathbf{d}_i)\}$, then $\pi(\mathbf{d}_i, t^i, f)$ is defined as a mapping from I to $\mathcal{Y}^{n(\mathbf{d}_i)}$. We use $\pi_i(j)$, the j^{th} element of $\pi(\mathbf{d}_i, t^i, f)$, to denote the summary state of item j (i.e. s_{ij}). So the learning process comes down to be that of minimizing the loss function which presents the disagreement between $\pi(\mathbf{d}_i, t^i, f)$ and the list of summary state $\mathbf{y}_i$, for all of the document collections with their corresponding topic.

In summarization, some performance measures are used to evaluate the "goodness" of a summarizer or a summarizing function. Conventionally, the performance of a summarizer is evaluated by the manual summary. Normally, a manual summary based measure can be defined, as in ROUGE [15]. In this paper, we utilize a general function $E(\pi(\mathbf{d}_i, t^i, f), \mathbf{y}_i) \in [-1, +1]$ to represent the performance measures. The first argument of E is the permutation mapping π created using the summarizing function f on $\mathbf{d}_i$. The second argument is the list of summary state $\mathbf{y}_i$ given by humans based on the specified topic. E measures the agreement between π and $\mathbf{y}_i$, i.e., the derived summary and the manual summary.

As an example of performance measures, we present the definitions of MAP in summarization. Given a document collection $\mathbf{d}_i$, the corresponding list of summary states $\mathbf{y}_i$, and a permutation mapping π_i on $\mathbf{d}_i$, average precision for summarization of $\mathbf{d}_i$ is defined as:

$$E_{MAP}(\pi_i, \mathbf{y}_i) = \frac{\sum_{j=1}^{n(\mathbf{d}_i)} \pi_i(j) \cdot y_{ij}}{n(\mathbf{d}_i)} \quad (3)$$

where $\pi_i = \pi(\mathbf{d}_i, t^i, f)$, $\pi_i(j)$ denotes the summary state of s_{ij}.

The score evaluated by ROUGE is also a popular performance measure for summarization, which is described in section 4.2 in detail.

In learning framework for summarization, the loss function L can be defined on the basis of such general summarization performance measures, signed as $L(E)$.

In this paper, we define the summarization model as a linear combination of base summarizing functions (base summarizers):

$$f(\mathbf{x}) = \sum_{k=1}^{K} \alpha_k h_k(\mathbf{x}), \quad (4)$$

where $h_k(\mathbf{x})$ is a base summarizer, α_k is its weight, and K is the number of base summarizers. Thus, the hypothesis space for the learning process can be defined as:

$$\mathcal{H} = \left\{ \sum_{k=1}^{K} \alpha_k h_k \middle| K \in \mathcal{N},\ \alpha_k \in \mathcal{R},\ h_k : \mathcal{X} \mapsto \mathcal{Y} \right\}$$

The learning process for summarization is a process of choosing an appropriate function f from hypothesis space $\mathcal{H}$ (a given set of functions). The goal is to find $f(\mathbf{x})$ so that the expected loss of f is as small as possible, i.e.

$$\begin{aligned} f &= \arg\min_{f \in \mathcal{H}} \mathbb{E}\Big(L\Big(E\big(\pi(\mathbf{d}, t, f), \mathbf{y}\big)\Big)\Big) \\ &= \arg\max_{f \in \mathcal{H}} \mathbb{E}\Big(E\big(\pi(\mathbf{d}, t, f), \mathbf{y}\big)\Big) \end{aligned}$$

where $\mathbb{E}(\cdot)$ denotes the expectation of a random variable.

Given a training set $\big\{(\mathbf{d}_i, t^i, \mathbf{y}_i)\big\}_{i=1}^{m}$, we can consider finding $f(\mathbf{x})$ in $\mathcal{H}$ that minimizes the empirical loss:

$$\begin{aligned} f &= \arg\min_{f \in \mathcal{H}} \sum_{i=1}^{m} L\Big(E\big(\pi(\mathbf{d}_i, t^i, f), \mathbf{y}_i\big)\Big) \\ &= \arg\max_{f \in \mathcal{H}} \sum_{i=1}^{m} E\big(\pi(\mathbf{d}_i, t^i, f), \mathbf{y}_i\big) \end{aligned}$$

3.3 Algorithm

With the formal description of learning framework given above, we propose an algorithm inspired by boosting methods, which optimizes a loss function based on the summarization performance measures. The new summarization algorithm is referred to as "AdaSum" and described in Figure 4.

AdaSum takes a training set $S = \big\{(\mathbf{d}_i, t^i, \mathbf{y}_i)\big\}_{i=1}^{m}$ as input, the performance measure function E and the number of iteration K as parameters. AdaSum runs Krounds and at each round it creates a base summarizer $h_k (k = 1, \dots, K)$. Finally, it outputs a summarizing function f by linearly combining the base summarizers.

As can be seen from the algorithm, in each iteration round, AdaSum maintains a distribution of weights over the document collections in the training set, which is denoted as P_k at round k. AdaSum sets all of the initial weights equally so that $P_1(i) = 1/m$. At each round, it increase the weights of those data that are not learned well in f_k, the model created so far. As a result, the learning at the next round will be focused on the creation of a base summarizer that can work on those "hard" data.

At each round, a base summarizer h_k is construction based on training data with weight distribution P_k. The goodness of a base summarizer is measured by the performance measure E weighted by P_k:

$$\sum_{i=1}^{m} P_k(i) E(\pi(\mathbf{d}_i, t^i, h_k), \mathbf{y}_i).$$

Once a base summarizer h_k is built, AdaSum chooses a weight $\alpha_k > 0$ for the base summarizer. Intuitively, α_k measures the importance of h_k.

A summarizing model f_k is created at each round by linearly combining the base summarizers constructed so far $h_1, ..., h_k$ with weights $\alpha_1, ..., \alpha_k$. f_k is then used for updating the distribution P_{k+1}.

AdaSum is a simple yet powerful method. More importantly, it is a method that can be justified from the theoretical viewpoint, as discussed below. In addition, AdaSum has several other advantages when compared with the existing summarization methods such as Mutual Reinforcement Principle (MPR) [23], LexRank [6] and GSPS [24]. Firstly, AdaSum can incorporate any performance measures, provided that the measure is topic based and in the range of $[-1, +1]$. In contrast, the existing methods only minimize loss functions that are loosely related to the manual summary based measures [16]. Moreover, AdaSum employs a more reasonable framework for performing the summarizing task than the existing methods. The adaptive learning process of AdaSum is a novel viewpoint for multi-document summarization.

Input: $S = \{\mathbf{d}_i, t^i, \mathbf{y}_i\}_{i=1}^m$, and parameters E and K
Initialize $P_1(i) = 1/m$
Output: Output summarizing model: f
for $k \leftarrow 1$ **to** K **do**

- Create base summarizer h_k with weighted distribution P_k on S
- Choose α_k

$$\alpha_k = \frac{1}{2} \cdot \ln \frac{1 + E(\pi(\mathbf{d}_i, t^i, h_k), \mathbf{y}_i)}{1 - E(\pi(\mathbf{d}_i, t^i, h_k), \mathbf{y}_i)}$$

- Create f_k

$$f_k(\mathbf{x}) = \sum_{l=1}^{k} \alpha_l h_l(\mathbf{x})$$

- Update P_{k+1}

$$P_{k+1} = \frac{\exp\{-E(\pi(\mathbf{d}_i, t^i, h_k), \mathbf{y}_i)\}}{\sum_{j=1}^{m} \exp\{E(\pi(\mathbf{d}_i, t^i, h_k), \mathbf{y}_i)\}}$$

end

Figure 4: The AdaSum algorithm.

3.4 Construction of Feature Space and Base Summarizer

We consider an efficient implementation for drawing out the features of data $\{(\mathbf{d}_i, t^i)\}$ to construct the feature space $\mathcal{X}$.

A feature vector $\mathbf{x} = \mathbf{\Psi}(s, t)$ is created from each sentence-topic pair (s, t). All of $\mathbf{x}$ span the feature space $\mathcal{X}$. Each feature (each element of $\mathbf{\Psi}$) stands for some relation between sentence and topic, e.g. similarity, distance and correlation. In summarization, a summarizer assigns a relevance score to every sentence in document collection, this score can also be viewed as one feature in $\mathcal{X}$.

In this paper, we utilize the adaptive model proposed in section 3.1 to create each feature in like manner. Some well-known algorithm (systems) can be improved in the adaptive model, their outputs form the features in $\mathcal{X}$.

Based on the construction of feature space, we choose the feature as base summarizer, which has the optimal weighted performance among all of the features in $\mathcal{X}$:

$$\max_k \sum_{i=1}^{m} P(i) E(\pi(\mathbf{d}_i, t^i, x_k), \mathbf{y}_i),$$

where $P(i)$ denotes the weight of training instance $(\mathbf{d}_i, t^i, \mathbf{y}_i)$.

Obtaining base summarizers by this way, the learning process turns out to be that of repeatedly selecting features and linearly combining the selected features. Note that features which are not selected in the training phase will have a weight of zero.

The same ideas are also used in our experiments in section 5.

3.5 Algorithm Theoretical Analysis

The algorithm AdaSum is equivalent to forward stagewise additive modeling [12] using the exponential loss function

$$\begin{aligned} L(\mathbf{y}, f) &= L\Big(E\big(\pi(\mathbf{d}, t, f), \mathbf{y}\big)\Big) \\ &= \exp\Big(-E\big(\pi(\mathbf{d}, t, f), \mathbf{y}\big)\Big). \end{aligned} \tag{5}$$

A direct and effective loss measure for multi-document summarization is hard to find, and the existing measures are the manual summary based. Ideally, we want to maximize the summarization accuracy in terms of a performance measure on the training data:

$$\max_{f \in \mathcal{H}} \left\{ \sum_{i=1}^{m} E(\pi(\mathbf{d}_i, t^i, f), \mathbf{y}_i) \right\}$$

This is equivalent to minimize the loss

$$\min_{f \in \mathcal{H}} \left\{ \sum_{i=1}^{m} \Big(1 - E(\pi(\mathbf{d}_i, t^i, f), \mathbf{y}_i)\Big) \right\} \tag{6}$$

Because E may be a noncontinuous function and thus be difficult to handle, we instead attempt to minimize the convex upper bound of the loss in (6):

$$\min_{f \in \mathcal{H}} \left\{ \sum_{i=1}^{m} \exp\big(-E(\pi(\mathbf{d}_i, t^i, f), \mathbf{y}_i)\big) \right\} \tag{7}$$

i.e. we utilize the exponential loss function (5) in learning process. In this paper, we consider an additive model (4). Then the optimization of (7) can practise the gradient-descent paradigm in function space. One must solve

$$(\alpha_k, h_k) = \arg\min_{\alpha, h} \sum_{i=1}^{m} \exp\big(-E(\pi(\mathbf{d}_i, t^i, f_{k-1} + \alpha h), \mathbf{y}_i)\big) \tag{8}$$

for the summarizer h_k and corresponding coefficient α_k to be added at each step.

Following the above idea, we get the algorithm in Figure 4.

4. EVALUATION SUMMARIZATION

In order to evaluate our proposed approach, we select three methods, MRP, LexRank and GSPS as the base summarizers in the experiments, for they are the state-of-art summarization methods. Furthermore, we compare AdaSum with the DUC 2007 top performing systems. We use the ROUGE metrics on DUC 2007 data sets for comparison. And the ROUGE score of the start-of-the-art systems came from Hoa's overview of DUC 2007 [4].

4.1 DUC Task Description

Every year, Document Understanding Conferences (DUC[2]) evaluates competing research group's summarization systems on a set of summarization tasks. In DUC 2005, DUC 2006 and DUC 2007, the task is to produce summaries of document collections in response to short topic statements that define what the summaries should address. The summaries are limited to 250 words in length. The DUC 2007 task was a complex question-focused summarization task that required summaries to piece together information from multiple documents to answer a question or set of questions as posed in a DUC topic. NIST[3] Assessors developed a total of 45 DUC topics to be used as test data. For each topic, the assessor selected 25 related documents from the Associated Press, New York Times and Xinhua text collection and formulated a DUC topic statement, which was a request for information that could be answered using the selected documents. The topic statement could be in the form of a question or set of related questions and could include background information that the assessor thought could help clarify his/her information need. The assessor also indicated the "granularity" of the desired response for each DUC topic. That is, they indicated whether they wanted the answer to their question(s) to name specific events, people, places, etc., or whether they

[2]Document Understanding Conferences, http://duc.nist.gov/
[3]National Institute of Standard and Technology, http://www.nist.gov/

wanted a general, high-level answer. Only one value of granularity was given for each topic, since the goal was not to measure the effect of different granularity on system performance for a given topic, but to provide additional information about the user's preferences to both human and automatic summarization.

4.2 ROUGE

Automatic text summarization has drawn a lot of interest in the NLP and IR communities in the past years. Recently, a series of government-sponsored evaluation efforts in text summarization have taken place in both the United States and Japan. The most famous DUC evaluation is organized yearly to compare the summaries created by systems with those created by humans. Following the recent adoption of automatic evaluation techniques (such as BLEU/NIST) by the machine translation community, a similar set of evaluation metrics - known as ROUGE - were introduced for both single and multi-document summarization. ROUGE includes four automatic evaluation methods that measure the similarity between summaries: ROUGE-N, ROUGE-L, ROUGE-W, and ROUGE-S. Formally, ROUGE-N measures the n-gram recall between a candidate summary and a set of reference summaries. ROUGE-N is computed as follows:

$$\text{ROUGE-N} = \frac{\sum_{S\in\{Ref.Summ.\}}\sum_{gram_n\in S} Count_{match}(gram_n)}{\sum_{S\in\{Ref.Summ.\}}\sum_{gram_n\in S} Count(gram_n)} \tag{9}$$

ROUGE-L uses the longest common subsequence (LCS) metric to evaluate summaries. In this method, a sequence $R = [r_1, r_2, ..., r_n]$ is considered to be a subsequence of another sequence $S = [s_1, s_2, ..., s_n]$ if there exists a strictly-increasing sequence of indices for S (i.e. $I = [i_1, i_2, ..., i_k]$ such that for all $j = 1, 2, ...k, s_{ij} = z_j$. The longest common subsequence for R and S can be defined as the sequence common to both R and S with the greatest length. ROUGE-L is based on the assumption that pairs of summaries with longer LCS scores will be more similar than those summaries with shorter LCS scores. To capture this generalization, if we assume that summary sentence X and Y can be represented as a sequence of words, an LCS-based F-measure can be calculated to estimate the similarity between a reference summary X of length m and a candidate summary Y of length n as follows:

$$R_{lcs} = \frac{LCS(X,Y)}{m}, P_{lcs} = \frac{LCS(X,Y)}{n}$$

$$F_{lcs} = \frac{(1+\beta^2)R_{lcs}P_{lcs}}{R_{lcs}+\beta^2 P_{lcs}} \tag{10}$$

Here, $LCS(X,Y)$ is equal to the length of the LCS of X and Y and $\beta = P_{lcs}/R_{lcs}$. LCS can be also used to compute an F-measure for an entire summary, not just a single sentence. The summary-based LCS F-measure can be computed as follows:

$$R_{lcs} = \frac{\sum_{i=1}^{u} LCS\cup(r_i,C)}{m}, P_{lcs} = \frac{\sum_{i=1}^{u} LCS\cup(r_i,C)}{n}$$

$$F_{lcs} = \frac{(1+\beta^2)R_{lcs}P_{lcs}}{R_{lcs}+\beta^2 P_{lcs}} \tag{11}$$

where u represents a reference summary of m words, C represents a candidate summary of n words, and $LCS\cup(r_i,C)$ is the LCS score of the union LCS between r_i and the candidate summary C.

5. EXPERIMENTAL RESULTS AND DISCUSSION

In this section we experimentally evaluate the quality of AdaSum performed on DUC 2007 dataset. Two different sets of experiments are presented. The first one focuses on the effectiveness of the topic representation optimization and the topic expansion during the feedback process. The latter experiments focuse on evaluating the summary performance on a real document collection.

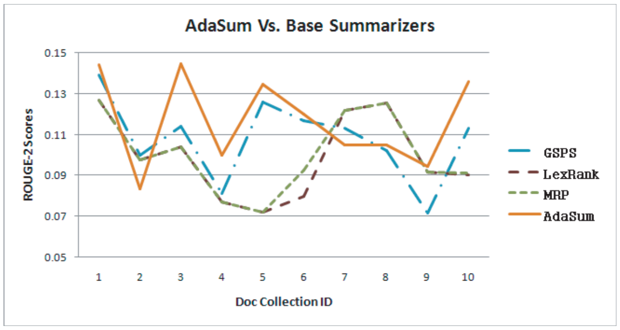
5.1 Topic Representation Optimization

In order to study how to understand the expected topic, we use the first ten document collections from the large-scale common data set evaluation conducted within the DUC 2007. For each collection, the input for summarization is available, along with a specified topic for the input and the document collections to be summarized that year. Each of the inputs containes around 25 documents and the summaries were 250 words long. As the following description shows, with the process of feedback, a short topic included few terms is expanded to a long related terms list.

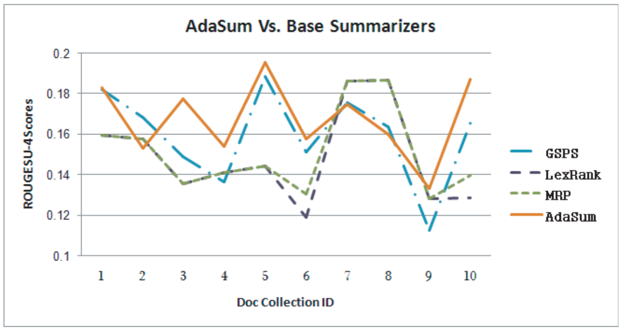
Table 1: Topic representation optimization of AdaSum, where the first and second columns are the identity number and corresponding title of selected topics, and the last column is the term distribution of optimized topic representation with our AdaSum.

Topic ID	Original Topic	Optimized Topic
D0701A	Southern Poverty Law Center	**poverty southern law center organization morris co-founder dees nonprofit** successfully obtain represent previously sue intend
D0702A	art and music in public schools	**school public dance** involve city year york **music student program art** new indispensable week six
D0703A	steps toward introduction of the Euro	january **launch single reserve** replace **foreign new billion** issue report tuesday know wim **bank currency**
D0704A	Amnesty International	kenyan **international country accuse government amnesty human** incite calculate **right charge** against seek reinstate disagreement
D0705A	Basque separatism	**independence separatist freedom kill arm acronym homeland basque** nearly language **campaign people spanish** name discussion
D0706B	Burma government change 1988	rule know **military myanmar party** also member **guise burma jail various arrest suu kyi parliament**
D0707B	Turkey and the European Union	full say **foreign european minister** new **union membership today effort pangalos perspective** vocation monday country
D0708B	world-wide chronic potable water shortages	report **people city province shortage face plague demand daily** move vietnam cause monday **potable** currently
D0709B	Angelina Jolie	**jolie angelina** father say fact brother work possibility **role** choose responsibility haven compliment secure earth
D0710C	Israel/Mossad "The Cyprus Affair"	**act against behalf cyprus turkey** two **suspicion** say **netanyahu arrest spy man israeli statement official**

Table 1 illustrates the topic representation optimization with our proposed feedback process in this paper. Results from the third column are the term distribution of estimated topics for the final estimated summary, and the terms are generated with feedback ranking. The bold terms are important for topic identification. Intu-



(a) AdaSum vs. Base Summarizers on ROUGE-2



(b) AdaSum vs. Base Summarizers on ROUGE-SU4

Figure 5: AdaSum vs. base summarizers performance comparison, where AdaSum can significantly improve the ROUGE-2 scores and ROUGE-SU4 scores.

itively, the derived term distribution in the third column can optimize the original topic represented by the specified topic in the second column.

The success of AdaSum in this task demonstrates its flexibility and effectiveness, and indicates that our proposed model is sufficiently expressive to represent more important topic properties. Thus, with topic representation optimization, the derived summary with topic representation could be more relative to the specified topic, and which is likely to reduce the topic representation mismatch between the original topic and the expected topic.

5.2 Summarization Performance

Table 2: Performance comparison with AdaSum and base summarizers. With comparison to the average scores on ROUGE-2 and ROUGE-SU4 scores for MRP, LexRank and GSPS, AdaSum can achieve significant improvement, and the last column is the improvement of AdaSum on both ROUGE scores.

Summarizer	ROUGE-2	ROUGE-SU4	AdaSum Improvement
AdaSum	**0.1168**	**0.1676**	(—,—)
MRP	0.1000	0.1509	(**+16.8%**, **+11.0%**)
LexRank	0.0987	0.1487	(**+18.4%**, **+12.7%**)
GSPS	0.1077	0.1592	(**+8.5%**, **+5.2%**)

In this section, we study the performance of AdaSum on the dataset of DUC 2007. Firstly, we choose the first ten document collections to evaluate the summarization performance achieved by AdaSum, and the summary performance for this ten collections is evaluated by ROUGE-2 and ROUGE-SU4 metrics. We utilize three base summarizers as features (MRP, LexRank and GSPS). Table 2 shows the average recall achieved by AdaSum and the base summarizers on DUC 2007 dataset respectively, where every test set is relative to a specified topic. The 2nd column and the 3rd column in Table 2 show the average recall on ROUGE-2 and ROUGE-SU4 respectively, and the last column represents the improvement of AdaSum against the base summarizers on both ROUGE scores.

From the results in Table 2, we can see that AdaSum improves the summarization performance on both ROUGE scores. In comparison with the scores of base summarizers, AdaSum can obtain significant improvement on both ROUGE scores. To illustrate the performance, we plot the curve of ROUGE-2 scores and ROUGE-SU4 scores in Figure 5. Looking at the results, AdaSum is effective to improve the summarization performance. Moreover, we investigate the summarization performance of AdaSum on the whole 45 document collections of DUC 2007, and the results are shown in Table 3. With comparison to three base summarizers, AdaSum can gain improvement about 23.0%, 18.4% and 5.6% on ROUGE-2 metric. Table 2 and Table 3 prove our proposed model is effectiveness for topic-oriented summarization. These results confirm our hypothesis about the benefit of adaptive summarization model, which also confirm that our proposed approach can significant improve the performance of summarization.

Table 3: Performance comparison with AdaSum and base summarizers on DUC 2007 dataset. With comparison to the average scores on ROUGE-2 and ROUGE-SU4 scores for MRP, LexRank and GSPS, AdaSum can also achieve significant improvement, and the last column is the improvement of AdaSum on both ROUGE scores.

Summarizer	ROUGE-2	ROUGE-SU4	AdaSum Improvement
AdaSum	**0.1172**	**0.1692**	(—,—)
MRP	0.0953	0.1493	(**+23.0%**, **+13.3%**)
LexRank	0.0963	0.1493	(**+18.4%**, **+13.3%**)
GSPS	0.1110	0.1638	(**+5.6%**, **+3.3%**)

We also investigate the significant performance of our proposed model with significant test experiments. The paired samples t-tests experiments on ROUGE scores in Table 4 show these differences on both ROUGE scores to be statistically significant at $p < 0.05$ both for individual judges and average scores across all judges, where six pairs are the ROUGE-2 scores and ROUGE-SU4 scores among AdaSum and three base summarizers respectively. The results of Sig. (1-tailed) indicate that our AdaSum performs statistically better than every base summarizer at $p < 0.05$.

In order to further confirm our hypothesis, we conduct the third experiment. We compare the summary performance of our proposed approach with the top performing systems in DUC 2007. With comparison to the average scores of ROUGE-2 on the first ten document collections, AdaSum can outperform the 2nd rank system, Microsoft. Although AdaSum can not outperform the best system, the difference is slight. Furthermore, in comparison with the average scores of ROUGE-2 on the whole 45 document collections of DUC 2007, AdaSum can obtain the 4th place. The results confirm that AdaSum performs well as compared to the state-of-the-art systems competing in DUC. In this paper, we choose MRP, LexRank and GSPS as the base summarizers, thus the results mostly tend to confirm the superiority of our proposed approach. Ideally, if given more empirical base summarizers and further optimization for algorithms, our proposed approach could achieve more significant results.

Table 4: **Paired Samples t-tests of ROUGE Scores among AdaSum and base summarizers on ROUGE-2 and ROUGE SU4, where Pair 1, Pair 2 and Pair 3 are the pairs of ROUGE-2 scores among AdaSum and three base summarizers (MRP, LexRank and GSPS) on the first 10 document collections of DUC 2007 respectively, and Pair 4, Pair 5 and Pair 6 are the pairs of the corresponding ROUGE-SU4 scores.**

	Paired Differences				t	df	$Sig.(1-tailed)$
			95% Confidence Interval of the Difference				
	Mean	Std. Deviation	Lower	Upper			
R2-AdaSum vs. R2-MRP	0.0168	0.0284	-0.0036	0.0371	1.8632	9	0.0477
R2-AdaSum vs. R2-LexRank	0.0181	0.0293	-0.0029	0.0391	1.9538	9	0.0412
R2-AdaSum vs. R2-GSPS	0.0091	0.0148	-0.0014	0.0197	1.9524	9	0.0413
SU4-AdaSum vs. SU4-MRP	0.0167	0.0263	-0.0021	0.0355	2.0056	9	0.0379
SU4-AdaSum vs. SU4-LexRank	0.0189	0.0285	-0.0015	0.0393	2.0930	9	0.0329
SU4-AdaSum vs. SU4-GSPS	0.0084	0.0137	-0.0014	0.0181	1.9295	9	0.0429

6. RELATED WORK

Automatic summarization is the process of automatically creating a compressed version of a given text so that it can provide useful information to the user. Summarization techniques leverage a wide range of Natural Language Processing (NLP) and discourse information. Some focus primarily on techniques that have been developed in Information Retrieval [10], while most try to leverage both IR approaches and some aspects of NLP [13]. In the previous research, most efforts on extractive summarization have been concentrated on statistical and NLP approaches. For example, the diversity of concepts covered by a document [3] has been first explored by Carbonell and Goldstein in 1998. They proposed to use Maximal Marginal Relevance (MMR), which selects summary sentences that are both relevant to the user query and least similar to the previously chosen ones. In [23], a mutual reinforcement principle (MRP) is employed to iteratively extract key phrases and sentences from a document. Most recently, graph-based ranking methods have been proposed for document summarization. Radev et al. [20] described an extractive multi-document summarizer which extracts a summary from multiple documents based on the document cluster centroids. Also, Erkan et al. [6] proposed a stochastic graph-based method (LexRank) for computing relative importance of textual units for NLP, where the sentence importance is measured by the concept of eigenvector centrality in a graph representation of sentences. In [24], a graph-based sub-topic partition algorithm is proposed by ranking sentence importance with a "personalized" LexRank and removing redundancy with subtopic partition, where the global features are taken as the "personalized" vector for LexRank.

In what follows, we review the work that are closely related to our approach. There are two topics which are related to our current work - boosting in classification and AdaRank in IR.

6.1 Boosting

Boosting is a general technique for improving the accuracies of machine learning algorithms. The basic idea of boosting is to repeatedly construct "weak learners" by re-weighting training data and form an ensemble of weak learners such that the total performance of the ensemble is "boosted". Freund and Schapire have proposed the first well-known boosting algorithm called AdaBoost (Adaptive Boosting) [8], which is designed for binary classification (0-1 prediction). Later, Schapire and Singer have introduced a generalized version of AdaBoost in which weak learners can give confidence scores in their predictions rather than make 0-1 decisions [21]. Extensions have been made to deal with the problems of multi-class classification [8, 9], regression [5], and ranking [7]. In fact, AdaBoost is an algorithm that ingeniously constructs a linear model by minimizing the "exponential loss function" with respect to the training data.

6.2 AdaRank

The key problem for document retrieval is ranking, specifically, how to create the ranking model (function) that can sort documents based on their relevance to the given query. It is a common practice in IR to tune the parameters of a ranking model using some labeled data and one performance measure [1]. Xu and Li have proposed a learning algorithm within the framework of boosting called AdaRank [22], which can minimize a loss function directly defined on the basis of general IR performance measures, and the optimization of loss function is based on queries. The objective of AdaRank is to construct a ranking function which achieves the best results in ranking of the training data in the sense of minimization of a loss function. Ideally the loss function is defined on the basis of the performance measure used in testing. AdaSum can be viewed as a machine learning method for loss function optimization of summaries performance measures, based on a different approach.

7. CONCLUSIONS AND FUTURE WORK

In this paper, we propose a novel adaptive model for summarization, AdaSum, under the assumption that the summary and the topic representation can be mutually boosted. AdaSum aims to simultaneously optimize the topic representation and extract effective summaries. In order to alleviate the topic mismatch problem in summarization and minimize the expected risk between specified topic and expected topic, AdaSum utilizes a linear combination of base summarizers as its model. Moreover, a general learning framework is proposed to construct a new summarizer with a linear combination of base summarizers. The model can naturally make full use of the reinforcement between the topic and the derived summaries by employing a boosting approach to optimize the topic representation and generate effective summaries. We prove that the training process of AdaSum can exactly enhance the performance measure used. Experimental results on DUC 2007 dataset show that AdaSum significantly outperforms the baseline methods of MRP, LexRank, and GSPS. Compared to the top systems in DUC 2007, our proposed approach is also very competitive.

The study has two main contributions: (1) a mutual boosting process is employed to optimize the topic representation and minimize the expected bias between the estimated topic and the expected topic, (2) inspired by the work of AdaBoost and AdaRank, a linear combination of base summarizers is employed to obtain a new summarizer which can boost the performance of derived summaries with a general learning framework. In addition, our proposed model offers other advantages: ease of implementation, theoretical soundness, efficiency in training, and high performance in summarization. To some extent, AdaSum can be viewed as a model of learning to summarize.

As for future research, we plan to exploit more effective measures to evaluate the similarity between the expected topic and the estimated topic for the optimization of topic representation. And introducing more empirical base summarizers for the general learning framework can further optimize the performance of our proposed model. Moreover, we would like to adapt our propose model to task-focused summarization problems, such as web search result snippets.

8. ACKNOWLEDGMENTS

This research is supported by The 973 National Basic Research Program of China (2004CB318109) and The 863 National High-Tech Research and Development Plan of China (2006AA01Z452 & 2007AA01Z441). We would like to thank Professor Jian-Yun Nie for helpful discussions and invaluable suggestions. Finally, We would like to thank the anonymous reviewers for their insightful comments.

9. REFERENCES

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

[2] R. Barzilay and L. Lee. Catching the drift: Probabilistic content models, with applications to generation and summarization. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 113–120, 2004.

[3] J. G. Carbonell and J. Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 335–336, 1998.

[4] H. T. Dang. Overview of duc 2007. In *Proceedings of Document Understanding Conference, Rochester, New York, USA*, 2007.

[5] N. Duffy and D. P. Helmbold. Boosting methods for regression. *Machine Learning*, 47(2-3):153–200, 2002.

[6] G. Erkan and D. R. Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22:457–479, 2004.

[7] Y. Freund, R. D. Iyer, R. E. Schapire, and Y. Singer. An efficient boosting algorithm for combining preferences. *Journal of Machine Learning Research*, 4:933–969, 2003.

[8] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997.

[9] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: a statistical view of boosting. *The Annals of Statistics*, 28(2):337–374, 2000.

[10] J. Goldstein, V. Mittal, J. Carbonell, and M. Kantrowitz. Multi-document summarization by sentence extraction, 2000.

[11] S. M. Harabagiu and V. F. Lacatusu. Topic themes for multi-document summarization. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 202–209, 2005.

[12] T. Hastie, R. Tibshirani, and J. H. Friedman. *The Elements of Statistical Learning*. Springer, 2001.

[13] E. Hovy and C. Lin. Automated text summarization in summarist, 1997.

[14] K. Knight and D. Marcu. Statistics-based summarization - step one: Sentence compression. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*, pages 703–710, 2000.

[15] C.-Y. Lin. Rouge: a package for automatic evaluation of summaries. In *Proceedings of WAS, Barcelona, Spain*, 2004.

[16] C.-Y. Lin and E. H. Hovy. The automated acquisition of topic signatures for text summarization. In *Proceedings of the 18th conference on Computational linguistics*, pages 495–501, 2000.

[17] H. Luhn. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2(2):159–165, 1958.

[18] D. Mallett, J. Elding, and M. A. Nascimento. Information-content based sentence extraction for text summarization. In *Proceedings of the International Conference on Information Technology: Coding and Computing*, pages 214–218, 2004.

[19] I. Mani and M. T. Maybury. *Advances in Automatic Text Summarization*. The MIT Press, 1999.

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

[21] R. E. Schapire and Y. Singer. Improved boosting algorithms using confidence-rated predictions. *Machine Learning*, 37(3):297–336, 1999.

[22] J. Xu and H. Li. Adarank: a boosting algorithm for information retrieval. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 391–398, 2007.

[23] H. Y. Zha. Generic summarization and keyphrase extraction using mutual reinforcement principle and sentence clustering. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 113–120, 2002.

[24] J. Zhang, X. Cheng, and H. Xu. Gspsummary: A graph-based sub-topic partition algorithm for summarization. In *Proceedings of The 4th Asia Information Retrieval Symposium(AIRS2008)*, pages 321–334, 2008.