Expert Systems with Applications 39 (2012) 12460–12473

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

# GenDocSum + MCLR: Generic document summarization based on maximum coverage and less redundancy

Rasim M. Alguliev, Ramiz M. Aliguliyev *, Makrufa S. Hajirahimova

*Institute of Information Technology, Azerbaijan National Academy of Sciences, 9, B. Vahabzade Street, Baku AZ1141, Azerbaijan*

ARTICLE INFO

*Keywords:*
Generic document summarization
Maximum coverage
Less redundancy
Optimization model
Differential evolution algorithm

ABSTRACT

With the rapid growth of information on the Internet and electronic government recently, automatic multi-document summarization has become an important task. Multi-document summarization is an optimization problem requiring simultaneous optimization of more than one objective function. In this study, when building summaries from multiple documents, we attempt to balance two objectives, content coverage and redundancy. Our goal is to investigate three fundamental aspects of the problem, i.e. designing an optimization model, solving the optimization problem and finding the solution to the best summary. We model multi-document summarization as a Quadratic Boolean Programing (QBP) problem where the objective function is a weighted combination of the content coverage and redundancy objectives. The objective function measures the possible summaries based on the identified salient sentences and overlap information between selected sentences. An innovative aspect of our model lies in its ability to remove redundancy while selecting representative sentences. The QBP problem has been solved by using a binary differential evolution algorithm. Evaluation of the model has been performed on the DUC2002, DUC2004 and DUC2006 data sets. We have evaluated our model automatically using ROUGE toolkit and reported the significance of our results through 95% confidence intervals. The experimental results show that the optimization-based approach for document summarization is truly a promising research direction.



## 1. Introduction

Interest in text summarization started with advent of on-line publishing and the increased impact of the Internet and electronic government (e-government) services (Hung, Tang, Chang, & Ke, 2009). With the growing popularity of the Internet and e-government services (for example, electronic document management systems) a huge amount of electronic documents are available online. The increasing amount of electronic documents has led to information overload. In this case, the user due to the large amount of information does not read many relevant and interesting documents. Text summarization is an issue to attack the information overload problem. Text summarization refers to the process of taking a textual document, extracting content from it, and presenting the most important content to the user in a condensed form and in a manner sensitive to the user's or application needs. To achieve this goal, text summarization systems should identify the most salient information in a document and convey it in less space than the original document. Therefore, the text summarization has been used as the useful tools in order to help users efficiently find useful information from immense amount of information (Ko & Seo, 2008; Mani & Maybury, 1999; Ouyang, Li, Li, & Lu, 2011). Text summarization helps to simplify information search and reduce the search time by pointing the most relevant information that allows users to quickly comprehend the information in a large document (Kutlu, Cigir, & Cicekli, 2010; Mani & Maybury, 1999; Wan & Xiao, 2010).

Automatically generating summaries from large text corpora has long been studied in both information retrieval and natural language processing (NLP). There are several types of text summarization tasks. Document summaries can be classified into different types according to different dimensions. For example, extractive summarization can be either generic or query-relevant. Generic document summarization should reflect the major content of the documents without any additional information and prior knowledge. Query-oriented document summarization should focus on the information expressed in the given queries, i.e. the summaries must be biased to the given queries (Kutlu et al., 2010; Mani & Maybury, 1999; Tang, Yao, & Chen, 2009; Wan & Xiao, 2010).

Depending on the number of documents to be summarized, the summary can be a single-document or a multi-document (Wan & Xiao, 2010). Single-document summarization can only condense one document into a shorter representation, whereas multi-document summarization can condense a set of documents into

* Corresponding author.
*E-mail addresses:* r.aliguliyev@gmail.com, a.ramiz@science.az (R.M. Aliguliyev).

0957-4174/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2012.04.067

a summary. Multi-document summarization can be considered as an extension of single-document summarization and used for precisely describing the information contained in a cluster of documents and facilitate users to understand the document cluster. Since it combines and integrates the information across documents, it performs knowledge synthesis and knowledge discovery, and can be used for knowledge acquisition. It differs from single document summarization in that it is important to identify differences and similarities across documents (Kutlu et al., 2010; Mani & Maybury, 1999; Ouyang et al., 2011).

Automatic document summarization methods can be divided into two categories: supervised and unsupervised methods. Supervised methods are based on algorithms that use a large amount of human-made summaries, and as a result, are most useful for documents that are relevant to the summarizer model. Thus, they do not necessarily produce a satisfactory summary for documents that are not similar to the model. In addition, when users change the purpose of summarization or the characteristics of documents, it becomes necessary to reconstruct the training data or retrain the model. Unsupervised methods do not require training data such as human-made summaries to train the summarizer (Kutlu et al., 2010; Mani & Maybury, 1999).

There are two types of summarization: extractive summarization and abstractive summarization. Extractive summarization selects the important sentences from the original documents to form a summary, while abstractive summarization paraphrases the corpus using novel sentences. Extractive summarization usually ranks the sentences in the documents according to their scores calculated by a set of predefined features, such as term frequency-inverse sentence frequency (TF-ISF), sentence or term position, and number of keywords. Abstractive summarization usually involves information fusion, sentence compression and reformulation. Although an abstractive summary could be more concise, it requires deep NLP techniques (Kutlu et al., 2010; McDonald, 2007; Ouyang et al., 2011; Wang, Li, & Weise, 2010). Therefore, extractive summaries are more feasible and practical. In this paper, we focus on extractive multi-document summarization.

Extractive document summarization clearly entails selecting the most salient information and putting it together in a coherent summary. The summary consists of multiple separately extracted sentences from different documents. Obviously, each of the selected sentences should individually be important. However, when many of the competing sentences are included in the summary, the issue of information overlap between parts of the output comes up, and a mechanism for addressing redundancy is needed (Chali, Hasan, & Joty, 2011). Therefore, when many of the competing sentences are available, given summary length limit, the strategy of selecting best summary rather than selecting best sentences becomes evidently important. Selecting the best summary is a global optimization problem in comparison with the procedure of selecting the best sentences (Huang, He, Wei, & Li, 2010). For content selection, document summarization includes how to identify the important content, remove the redundant content and keep the high content coverage. For linguistic quality, how to keep the content to be coherent and fluent is very significant (He, Qin, & Liu, 2012). In addition, it is known that coverage and redundancy two main criteria that decide the quality of summary. In this paper, we propose a new multi-document summarization approach, called MCLR (maximum coverage and less redundancy), via sentence extraction to simultaneously deal with these two concerns during sentence selection. We model multi-document summarization as a Quadratic Boolean Programing (QBP) problem where objective function is a weighted combination of the content coverage and redundancy objectives. The model employs two levels of analysis: first level, every sentence is scored according to the features it covers and second level, when, before being added to the final summary, the sentences deemed to be important are compared to each other and only those that are not too similar to other candidates are included in the final summary. We create a modified differential evolution algorithm to solve the optimization problem. We evaluate the proposed model on the DUC2002, DUC2004, and DUC2006 data sets (Document Understanding Conference) and show that the resulting summaries compare favorably on ROUGE metrics with those by existing state-of-the-art summarization systems.

The rest of paper is organized as follows. Section 2 discusses the related work on extractive multi-document summarization. In Section 3, we explain the proposed optimization model for multi-document summarization. Next, in Section 4, we give details of binary differential evolution algorithm which has been used to solve the optimization problem. The numerical experiments and results are given in Section 5. Section 6 addresses the conclusions and future work.

## 2. Related work

A variety of multi-document summarization methods have been developed in the literature. Most of the researchers have concentrated on not sentence-generation summarization methods but sentence-extraction methods in order to create document summary (Ko & Seo, 2008). To date, various extraction-based methods have been proposed for document summarization. The extraction-based document summarization method ranks sentences by their scores and selects ones with the highest scores as summaries (Takamura & Okumura, 2009). Different approaches employ different methods for estimating the importance of sentences. The work of Ouyang et al. (2011) studies how to apply regression models to the sentence-ranking problem in query-focused multi-document summarization. They implement the regression models using Support Vector Regression (SVR). SVR is the regression type of Support Vector Machines and is capable of building state-of-the-art optimum approximation functions.

The major challenge in multi-document summarization is that a document set may contain diverse information which is either related or unrelated to the main central topic. The selection of the distinct ideas included in the document is called diversity-based selection. Diversity is important to control the redundancy in summarized text and produce a more appropriate summary. Many approaches have been proposed for text summarization based on diversity. The pioneer work for diversity-based text summarization is maximal marginal relevance (MMR), introduced by Carbonell and Goldstein (1998), where a greedy algorithm selects the most relevant sentences, and at the same time avoids redundancy by removing sentences that are too similar to already selected ones. MMR maximizes marginal relevance in retrieval and summarization. In the MMR technique, a sentence has high marginal relevance if it is relevant to the query and contains minimal similarity to previously selected sentences. One major problem of MMR is that it is non-optimal because the decision is made based on the scores at the current iteration. In the work proposed by McDonald (2007), the summarization task was defined as a global inference problem which attempted to optimize three properties jointly, i.e., relevance, redundancy and length. The objective function is similar to MMR. An interactive extension of the MMR algorithm for query-focused summarization is proposed in Lin, Madnani, and Dorr (2010) where at each step it interactively asks the user to select the best sentence for inclusion in the summary. That is, instead of the system automatically selecting the candidate with the highest score, it presents the user with a ranked list of candidates for selection. Swarm diversity-based method (Binwahlan, Salim, & Suanmali, 2010) is an integration of the two

methods MMI (Maximal Marginal Importance) diversity-based text summarization (Binwahlan, Salim, & Suanmali, 2009) and swarm-based text summarization.

Currently, the most widely used summarization methods are clustering based. Clustering has been used as an effective tool for finding the diversity among the sentences. Clustering-based summarization methods usually perform various clustering techniques on the term-sentence matrices formed from the documents. After the sentences are grouped into different clusters, a centroid score is assigned to each sentence based on the average cosine similarity between the sentence and the rest of the sentences in the same cluster (He et al., 2012). Finally, the sentences with the highest scores in each cluster are selected to form the summary (Wang & Li, 2010). The papers (Alguliev & Aliguliyev, 2008; Alguliev & Aliguliyev, 2009; Alguliev, Aliguliyev, Hajirahimova, & Mehdiyev, 2011; Aliguliyev, 2009) present methods in which sentences are initially clustered, and then representative sentences are chosen from each cluster to be included into the summary. Here, the clustering stage minimizes redundancy (i.e., maximizes diversity) and the representative sentence selection maximizes relevance. Specifically, they rank all sentences according to their scores and select the top ranked sentences as candidate sentences. Intuitively, a sentence close to the cluster center is more representative and can be selected to represent the cluster. In Aliguliyev (2010), a new sentence extractive technique is developed. This technique calculates the average weight of a sentence with respect to cluster which it is assigned to. The weight of the sentence is calculated by a recursive formula. In Binwahlan et al. (2009), two ways were used for finding the diversity: the first one is a preliminary way where the document sentences are clustered based on the similarity and all resulting clusters are presented as a tree containing a binary tree for each group of similar sentences. The second way is to apply the proposed method on each branch in the tree to select one sentence as summary sentence. The clustering algorithm and binary tree were used as helping factor for finding the most distinct topics in the text. This approach firstly clusters the sentences and uses the obtained sentence clusters to generate a summary. Current document clustering methods usually represent documents as a term-document matrix and perform clustering algorithms on it. Although these clustering methods can group the documents satisfactorily, it is still hard for people to capture the meanings of the documents since there is no satisfactory interpretation for each document cluster. Besides, Wang, Zhu, Li, Chi, and Gong (2011) proposed a model to simultaneously cluster and summarize documents. Nonnegative factorization was performed on the term-document matrix using the term-sentence matrix as the base so that the document-topic and sentence-topic matrices could be constructed, from which the document clusters and the corresponding summary sentences were generated simultaneously. The paper (Lee, Park, Ahn, & Kim, 2009) presents a novel generic document summarization method using the negative matrix factorization (NMF). NMF is employed to decompose a given non-negative matrix into a multiplication of a non-negative semantic feature matrix, and a non-negative semantic variable matrix. This method has the following advantages: NMF selects more meaningful sentences than the LSA-related methods, because it can use more intuitively interpretable semantic features and grasp the innate structure of documents. The LSA-related methods (Gong & Liu, 2001) represent a sentence by means of a linear combination of semantic features. Wang, Li, Zhu, and Ding (2008) proposed a new framework based on sentence-level semantic analysis (SLSS) and symmetric non-negative matrix factorization (SNMF). First, it calculates the sentence-sentence similarities using semantic analysis and constructs the similarity matrix. Then symmetric matrix factorization which has been shown to be equivalent to normalized spectral clustering, is used to group sentences into clusters. Finally, the most informative sentences are selected from each cluster to form the summary. Cai and Li (2011) develop a new summarization approach which can simultaneously cluster and rank sentences by investigating the spectral characteristics of the similarity network which is constructed upon the document(s). Different from other existing clustering-based summarization approaches, this approach explores the "clustering structure" of sentences before the actual clustering algorithm is performed. The special clustering structure, called the structure of beams, is discovered by analyzing the spectral characteristics of the sentence similarity network.

Typically, sentence features, such as, position, length, keyword frequency, title-keyword, syntactic criteria, and indicator phrase, etc. define the relevance of each sentence. Each feature of a sentence may be weighted according to its importance in the application domain and the sum of the weighted features is the measure of its relevance. Centroid (Radev, Jing, Stys, & Tam, 2004) represents a non-optimization approach that evaluates terms and select sentences based on term importance. Huang, Yang, and Kuo (2009) investigate sentence features from a concept-level space and apply a fuzzy-rough hybrid scheme to define a sentence relevance measure. The method CN-Summ Antiqueira, Oliveira, Costa, and Nunes, 2009 uses a simple network of sentences that requires only surface text pre-processing, thus allowing assessing extracts obtained with no sophisticated linguistic knowledge. Bhattacharya, Ha-Thuc, and Srinivasan (2011) present MeSH-based methods for extracting core portions of full text documents. Specifically, they create a reduced version of each full text document that contains only its important portions. This reduced version may be viewed as a 'summary' but their interest is not to generate a human readable summary, rather it is to have an intermediate representation that may later be used for algorithmic functions such as to serve text retrieval and information extraction.

In extractive document summarization, finding an optimal summary can be viewed as a combinatorial optimization problem which is NP-hard to solve. There are a few papers exploring an optimization approach to document summarization. The potential of optimization based document summarization models has not been well explored to date. This is partially due to the difficulty to formulate the criteria used for objective assessment. As far as we know, the idea of optimizing summarization was mentioned in Filatova and Hatzivassiloglou (2004). They represented documents in a two dimensional space of textual and conceptual units with an associated mapping between them, and proposed a formal model that simultaneously selected important text units and minimized information overlap between them. The selection of the best textual units was regarded as an optimization problem over a general scoring function that maximized the distinct conceptual units. Huang et al. (2010) consider document summarization as a multi-objective optimization problem involving four objective functions, namely information coverage, significance, redundancy and text coherence. These functions measure the possible summaries based on the identified core terms and main topics. To eliminate redundancy, they used spectral clustering and classified each sentence into groups, each of which consists of semantically related sentences. The importance of a sentence within documents is defined using the Markov random walk model. Wang, Zhu, Li, and Gong (2009) proposed a Bayesian sentence-based topic model (BSTM) for multi-document summarization by making use of both the word-document and word-sentence associations. The BSTM models the probability distributions of selecting sentences given topics and provides a principled way for the summarization task. In Takamura and Okumura (2009), text summarization formalized as a budgeted median problem. This model covers the whole document cluster through sentence assignment. An advantage of this method is that it can incorporate asymmetric relations between

sentences in a natural manner. MCMR (Maximum Coverage and Minimum Redundant) (Alguliev et al., 2011) is an optimization-based approach which models text summarization as a linear integer-programing problem. This model generally attempts to optimize relevance and redundancy simultaneously.

## 3. Modeling document summarization as an optimization problem

Given a corpus $\mathbf{D} = \{d_1, d_2, \ldots, d_N\}$ of topic-related documents, where $N$ is the number of documents. We represent the corpus as the set of sentences $\mathbf{S} = \{s_1, s_2, \ldots, s_n\}$ from all the documents in the corpus $\mathbf{D}$, where $s_i$ denotes $i$th sentence in $\mathbf{S}$, $n$ is the number of sentences in the document corpus.

### 3.1. Sentence representation and similarity measure

In text mining, a textual unit is represented by the weights of the words that it contains, ignoring the order of the words and any punctuation. Formally, in a collection, a sentence is represented by a vector that is defined as a bag-of-words $s_i = \{\omega_{i1}, \omega_{i2}, \ldots, \omega_{im}\} \in R^m$, where $\omega_{ik}$ is the weight of the $k$th word in the $i$th sentence and $m$ is the number of words actually appearing in the collection.

Here, the weight $\omega_{ik}$ associated with $k$th word in the sentence $s_i$ is calculated using the tf-isf (term frequency–inverse sentence frequency) scheme:

$$\omega_{ik} = tf_{ik} \times isf_k, \tag{1}$$

where $tf_{ik}$ is the number of occurrences of the $k$th word in sentence $s_i$, and $isf_k = \log(n/n_k)$, $n_k$ is the number of sentences containing the $k$th word.

Text similarity measures play an important role in text-related research and applications, in particular, in areas such as text summarization (Aliguliyev, 2009), document clustering (Aliguliyev, 2009), textual knowledge representation and knowledge discovery (Islam & Inkpen, 2008), and information retrieval (Tsai, Tang, & Chan, 2010). Existing methods for computing text similarity have focused mainly on large documents. We focus on computing the similarity between two sentences. Recently, with the development of NLP applications, the need for an effective and accurate method to compute the similarity between two short or sentence-length text snippets has been identified (Islam & Inkpen, 2008; Wenyin, Quan, Feng, & Qiu, 2010). Similarity of short text snippets has applications in various computational areas. For example, in text mining short text similarity can be applied as a measure to discover knowledge from textual databases.

In this study, the similarity between two sentences $s_i$ and $s_j$ is defined by the following formula (Wenyin et al., 2010):

$$sim(s_i, s_j) = \sum_{l=1}^{m} \left[ \left( \sum_{k=1}^{m} \omega_{ik} p_{kl} \right) \left( \sum_{k=1}^{m} \omega_{jk} p_{kl} \right) \right], \tag{2}$$

where $p_{kl}$ is the similarity between $k$th and $l$th words (Alguliev & Aliguliyev, 2009):

$$p_{kl} = \exp(-\mathrm{NGD}_{kl}). \tag{3}$$

In Eq. (3), $\mathrm{NGD}_{kl}$ is the Normalized Google Distance between $k$th and $l$th words (Alguliev & Aliguliyev, 2009; Aliguliyev, 2009):

$$\mathrm{NGD}_{kl} = \frac{\max\{\log(n_k), \log(n_l)\} - \log(n_{kl})}{\log n - \min\{\log(n_k), \log(n_l)\}}, \tag{4}$$

where $n_{kl}$ denotes the number of sentences in the collection containing both $k$th and $l$th words.

### 3.2. Optimization model

Most of the state-of-art summarization systems are based on extracting the most salient and non-redundant sentences to composite the final summaries. Upon this extractive summarization framework, sentences first evaluated and ranked according to certain criteria and measures, and then the most significant ones are extracted from the original documents to generate a summary automatically. Without doubt, each of the selected sentences included in the summary should be individually important. However, this does not guarantee they collectively produce the best summary. For example, if the selected sentences overlap a lot with each other, such a summary is definitely not desired.

Document summarization, especially multi-document summarization in essence is a multi-objective optimization problem. It requires the simultaneous optimization of more than one objective function. A particular challenge for multi-document summarization is that a document set might contain diverse information either related or unrelated to the main topic. Hence, we need effective summarization methods to analyze the information stored in different documents and extract the globally important information to reflect the main topic. Another challenge for multi-document summarization is that the information stored in different documents inevitably overlaps with each other, and hence we need effective summarization methods to merge information stored in different documents, and if possible, contrast their differences (Huang et al., 2010). In this study, when building summaries from multiple documents, our approach generally attempts to optimize two objectives:

- *Content coverage:* The content coverage means that the generated summary should cover all subtopics as much as possible. It concerns the extent to which the information provided in original documents is included in the generated summary. A good summary should maximize this goal to its best.
- *Redundancy*: It is expected that the redundant or the duplicate information contained in the generated summary be minimized.

Optimizing these properties jointly is a challenging task. This is because the inclusion of relevant sentences relies on not only properties of the sentences themselves, but also properties of every other sentence in the summary. Unlike single-document summarization, redundancy is particularly important since it is likely that sentences from different documents will convey the same information (Huang et al., 2010).

We define two objective functions:

(1) $f_{\mathrm{cov}}(s_i)$: The content coverage of sentence $s_i$ participating in the summary:

$$f_{\mathrm{cov}}(s_i) = sim(s_i, O), \quad i = 1, \ldots n, \tag{5}$$

where $O$ is the center of the collection $\mathbf{S} = \{s_1, s_2, \ldots, s_n\}$.

It is known that (Radev et al., 2004) the center $O$ reflects the main content of collection $\mathbf{S} = \{s_1, s_2, \ldots, s_n\}$. Therefore, Eq. (5) evaluates the importance of sentence $s_i$ by measuring its similarity to the center $O$. $k$th coordinate $o_k$ of the center $O = [o_1, o_2, \ldots, o_m]$ is calculated as: $o_k = \frac{1}{n}\sum_{i=1}^{n} w_{ij}$, $k = 1, \ldots, m$. Higher value of $f_{\mathrm{cov}}(s_i)$ corresponds to higher content coverage of sentence $s_i$.

(2) $f_{\mathrm{red}}(s_i, s_j)$: The redundancy between sentences $s_i$ and $s_j$:

$$f_{\mathrm{red}}(s_i, s_j) = 1 - sim(s_i, s_j), \quad i \neq j = 1, \ldots n. \tag{6}$$

Higher value of $f_{\mathrm{red}}(s_i, s_j)$ corresponds to lower overlap in content between sentences $s_i$ and $s_j$, i.e. higher value of objective (6) provides minimum redundancy (high diversity) in the summary.

Let $x_i$ be binary variable, $x_i = 1$ when $s_i$ is selected; otherwise, $x_i = 0$. Since, a high quality summary should maximize the content coverage of the given document set, while minimize the redundancy, then text summarization problem can then be formalized as the following optimization problem:
maximize

$$f(\mathrm{X}) = w \cdot f_{\mathrm{cov}}(\mathrm{X}) + (1-w) \cdot f_{\mathrm{red}}(\mathrm{X}) = w \cdot \sum_{i=1}^{n-1} sim(s_i, O)x_i + (1-w) \cdot \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} (1 - sim(s_i, s_j))x_i x_j \quad (7)$$

subject to

$$\sum_{i=1}^{n} l_i x_i \leqslant L, \quad (8)$$

$$x_i \in \{0,1\}, \quad \forall i, \quad (9)$$

where $L$ is the given summary length limitation, $l_i$ indicates the length of sentence $s_i$. The number of words or in bytes measures the lengths of summary and sentence.

The optimization model (7)–(9) balances content coverage and diversity of the summary. In this model a parameter $w$ is used to combine the two objectives (5) and (6) into a scalar (7). $w$ is the weighting parameter, specifying the relative contributions of the $f_{\mathrm{cov}}(\cdot)$ and $f_{\mathrm{red}}(\cdot)$ functions to the hybrid function $f(\cdot)$. When $w = 1$, this model gives preference to sentences that are maximally relevant to the content of the document regardless of the diversity. In contrast, when $w = 0$, the model gives preference to sentences that diverge from the others regardless of the content coverage. By varying the parameter value $w$ and solving a sequence of Quadratic Boolean programing (QBP) problems (7)–(9) (for each $w$) the efficient summaries from the maximum content coverage summary ($w = 1$) to the high diversity summary ($w = 0$) can be found. If $w = 0.5$ the $f_{\mathrm{cov}}(\cdot)$ and $f_{\mathrm{red}}(\cdot)$ functions are assumed to be equally important. In our study we set $w = 0.75$. The impact of using different $w$'s is further studied in the second set of experiments reported in Section 5.5.

Now our objective is to find the binary assignment $\mathrm{X} = [x_i]$ with the best content coverage and least redundancy such that the summary length is at most $L$. The basic step of multi-document summarization is to extract a candidate sentence. If the length $L$ is the number of terms in the summary, the cost of each candidate sentence is the number of terms within it. The cardinality constraint (8) guarantees that the summary is bounded in length. The integrality constraint on $x_i$ (9) is automatically satisfied in the problem above.

## 4. Binary differential evolution algorithm

In solving optimization problems with a high-dimensional search space, the classical optimization algorithms do not provide a suitable solution because the search space increases exponentially with the problem size, therefore solving these problems using exact techniques is not practical. Over the last decades, there has been a growing interest in algorithms inspired by the behaviors of natural phenomena. It is shown by many researchers that these algorithms are well suited to solve complex computational problems such as optimization of objective functions (Rashedi, Nezamabadi-pour, & Saryazdi, 2009; Zielinski, Peters, & Laur, 2005), pattern recognition (Das & Suganthan, 2011; Das & Sil, 2010), document clustering (Aliguliyev, 2009), text summarization (Alguliev & Aliguliyev, 2009; Aliguliyev, 2009) and dynamic economic dispatch (Lu, Zhou, Qin, Li, & Zhang, 2010).

In our study, the optimization problem (7)–(9) was solved using a differential evolution (DE) (Aliguliyev, 2009; Das & Suganthan, 2011; Das & Sil, 2010). The execution of the DE is similar to other evolutionary algorithms like genetic algorithms or evolution strategies.

### 4.1. Population initialization

The evolutionary algorithms differ mainly in the representation of parameters and in the evolutionary operators. The classical DE is a population-based global optimization that uses a real-coded representation. Like to other evolutionary algorithms, DE also starts with a population of $N_{pop}$ individuals $\mathbf{P} = [X_1, X_2, \ldots, X_{N_{pop}}]$, where individual $X_p = [x_{p,1}, x_{p,2}, \ldots, x_{p,n}]$ ($p = 1, 2, \ldots, N_{pop}$) is an $n$-dimensional vector with parameter values determined randomly and uniformly between predefined search ranges $[X_{\min}, X_{\max}]$, where $X_{\min} = [x_{\min,1}, x_{\min,2}, \ldots, x_{\min,n}]$ and $X_{\max} = [x_{\max,1}, x_{\max,2}, \ldots, x_{\max,n}]$. Then mutation and crossover operators are employed to generate new candidate vectors, and a selection scheme is applied to determine whether the offspring or the parent survives to the next generation. The above process is repeated until a termination criterion is reached.

### 4.2. Mutation

A mutant vector, denoted as $Y_p(t) = [y_{p,1}(t), y_{p,2}(t), \ldots, y_{p,n}(t)]$ ($p = 1, 2, \ldots, N_{pop}$) is generated by using a mutation operator. The different mutation strategies are developed in literature (Aliguliyev, 2009; Das & Suganthan, 2011; Das & Sil, 2010; Lu et al., 2010; Zhang, Luo, & Wang, 2008). In this study, for each target vector $X_p(t)$ randomly choose two other vectors $X_{p1}(t)$ and $X_{p2}(t)$ from the same generation. Then it calculates the weighting combination of the differences $(X_p(t) - X_{p1}(t))$, $(X_p(t) - X_{p2}(t))$ and creates a mutant vector $Y_p(t)$ by adding the result to the best current solution $X^{best}(t)$. Thus, for the $i$th component of the mutant vector $Y_p(t)$ we obtain:

$$y_{p,i}(t) = x_i^{best}(t) + (\lambda_p - \lambda_{p1})(x_{p,i}(t) - x_{p1,i}(t)) + (\lambda_p - \lambda_{p2})(x_{p,i}(t) - x_{p2,i}(t)). \quad (10)$$

The coefficients $\lambda_p$, $\lambda_{p1}$, and $\lambda_{p2}$ we define as follows:

$$\lambda_p = \frac{|f(X_p)|}{|f(X_p)| + |f(X_{p1})| + |f(X_{p2})|},$$
$$\lambda_{p1} = \frac{|f(X_{p1})|}{|f(X_p)| + |f(X_{p1})| + |f(X_{p2})|},$$
$$\lambda_{p2} = \frac{|f(X_{p2})|}{|f(X_p)| + |f(X_{p1})| + |f(X_{p2})|}, \quad (11)$$

where $f(\cdot)$ is the objective function (7).

### 4.3. Crossover

After the mutation phase, a crossover operator is applied to each mutant vector and its corresponding target vector to yield a trial vector. A crossover operation comes into play after generating the mutant vector to enhance the potential diversity of the population. The mutant vector $Y_p(t)$ exchanges its components with the target vector $X_a(t)$ under this operation to form the trial vector $Z_p(t) = [z_{p,1}(t), z_{p,2}(t), \ldots, z_{p,n}(t)]$. Two commonly used crossover operations are the binomial crossover and the exponential crossover. In our study, the binomial crossover is employed and executed as follows:

$$z_{p,i}(t) = \begin{cases} y_{p,i}(t), & \text{if } \mathrm{rand}_{p,i} \leqslant CR \text{ or } i = i_{rand} \\ x_{p,i}(t), & \text{otherwise} \end{cases}, \quad (12)$$

where the index $i_{rand}$ refers to a randomly chosen integer in the set $\{1, 2, \ldots, n\}$ which is used to ensure that at least one component of the trial vector, $Z_p(t)$, differs from its target vector, $X_p(t)$. $CR$ is the

real-valued crossover rate in the range [0,1] which is set by the user and remains constant during the search process; $rand_{p,i}$ is the uniformly distributed random number within the range (0,1) chosen once for each $i$th component of the $p$th parameter vector, $i \in \{1, 2\ldots,n\}$, $p \in \{1, 2\ldots,N_{pop}\}$.

### 4.4. Selection

To keep the population size constant over subsequent generations, the next step of the algorithm calls for selection to determine whether the target or the trial vector survives to the next generation, i.e., at $t+1$. The selection operation is described as:

$$X_p(t+1) = \begin{cases} Z_p(t), & \text{if } f(Z_p(t)) \geqslant f(X_p(t)) \\ X_p(t), & \text{otherwise} \end{cases}. \quad (13)$$

Therefore, if the new trial vector yields an equal or higher value of the objective function, it replaces the corresponding target vector in the next generation; otherwise the target vector is retained in the population. Hence, the population either gets better (with respect to the maximization of the objective function) or remains the same in fitness status, but never deteriorates.

### 4.5. Discretization

Binary DE is the modified version of DE which operates in binary search spaces. In the binary DE, the real value of genes is converted to the binary space by the rule:

$$x_{p,i}(t+1) = \begin{cases} 1, & \text{if } rand_{p,i} < sigm(x_{p,i}(t+1)) \\ 0, & \text{otherwise} \end{cases}, \quad (14)$$

where, as before, $rand_{p,i}$ is a uniformly distributed random number lying between 0 and 1 which is called anew for each $i$th component of the $p$th parameter vector and $sigm(z)$ is the sigmoid function:

$$sigm(z) = \frac{1}{1+\exp(-z)}. \quad (15)$$

Using this transformation from the real-coded representation we obtain the binary-coded representation, $x_{p,i}(t) \in \{0,1\}$. Where the $x_{p,i}(t) = 1$ indicates that the $i$th sentence is selected to be included in the summary, otherwise, the $i$th sentence is not be selected.

### 4.6. Termination criterion

Mutation, crossover and selection operations continue until some termination criterion is reached. The termination criterion can be defined in a few ways like: (1) by a fixed number of iterations $t_{max}$, with a suitably large value of $t_{max}$ depending upon the complexity of the objective function; (2) when best fitness of the population does not change appreciably over successive iterations; (3) by a specified CPU time limit; and alternatively (4) attaining a pre-specified objective function value (Das & Suganthan, 2011; Das & Sil, 2010). According to our previous successful experience (Alguliev et al., 2011; Aliguliyev, 2010), in this paper we use the first one as the termination criteria, i.e., the algorithm terminates when the maximum number of generations $t_{max}$ is achieved.

### 4.7. Framework of the binary DE algorithm

Based on the above initialization, mutation, crossover, selection and discretization operations the framework of the binary DE algorithm can be summarized as:

**Step 1**: *Initialization*. Set the generation number $t = 0$, and randomly initialize a population of $N_{pop}$ target vectors, $\mathbf{P} = [X_1(t), X_2(t), \ldots, X_{N_{pop}}(t)]$, with $X_p = [x_{p,1}(t), x_{p,2}(t)\ldots, x_{p,n}(t)]$ uniformly distributed in the range $[X_{min}, X_{max}]$, $p = 1, 2\ldots, N_{pop}$.

**Step 2**: *Discretization*. Transform real-coded vectors to binary-coded vectors using Eq. (14).

**Step 3**: *Evaluation*. Evaluate each vector in $\mathbf{P} = [X_1, X_2, \ldots, X_{N_{pop}}]$ and select the vector with current best solution.

**Step 4**: *Mutation*. Generate a mutant vector $Y_p(t) = [y_{p,1}(t), y_{p,2}(t)\ldots, y_{p,n}(t)]$ for target vector $X_p(t)$ by using mutation operator (10), $p = 1, 2\ldots, N_{pop}$.

**Step 5**: *Crossover*. Generate a trial vector $Z_p(t)$ for target vector $X_p(t)$ by applying crossover operator (12) on $Y_p(t)$ and $X_p(t)$, $p = 1, 2\ldots, N_{pop}$.

**Step 6**: *Selection*. Evaluate each $Z_p(t)$ and determine the members of the target population of the next generation by using the selection scheme (13), $p = 1, 2\ldots, N_{pop}$.

**Step 7**: *Discretization*. Discretizate a new trial vector $X_p(t+1)$ by using Eq. (14), $p = 1, 2\ldots, N_{pop}$.

**Step 8**: *Stopping*. Repeat steps 2–7 until a user-specified maximum number $t_{max}$ of fitness calculation is reached.

**Step 9**: *Output*. Report the summary obtained by the best vector $X^{best}(t)$ as the final solution at maximum number of iteration.

### 4.8. Runtime complexity analysis

In this section, we analyze the time complexity of the proposed algorithm. Runtime-complexity analysis of the population-based stochastic search techniques like DE is a critical issue by its own right (Das & Suganthan, 2011; Wang et al., 2010; Zielinski et al., 2005). Das and Suganthan (2011) note that the average runtime of a standard DE algorithm usually depends on the population size, length of the vector and its stopping criterion. The authors pointed out that in each generation of DE a loop over $N_{pop}$ is conducted, containing a loop over $n$.

Assuming $N_{pop}$ and $n$ are the population size and the length of each vector in the DE, respectively, the time complexity of one generation for DE can be estimated as follows:

1. Time required for initialization of the individual is proportional to the length of the individual. As the length of the vector is equal to $n$, the time complexity of population initialization is $O(N_{pop} \times n)$.
2. Since the mutation and crossover operations are performed at the component level for each DE vector, the number of fundamental operations in DE is proportional to the total number of loops. Thus, mutation and crossover require $O(N_{pop} \times n)$ time each.
3. The time complexity for selection is $O(N_{pop})$.
4. Fitness computation is composed of three steps:
   - Complexity of computing similarity of $n$ sentences to the center of document collection is $O(N_{pop} \times n \times m)$.
   - For updating the centers (of summaries) total complexity is $O(N_{pop} \times m)$.
   - Complexity of computing similarity between $n$ sentences is $O(N_{pop} \times m \times n^2)$.

Therefore, the fitness evaluation has total complexity $O(N_{pop} \times m \times n^2)$.

Thus summing up the above complexities, total time complexity becomes $O(N_{pop} \times m \times n^2)$ per generation. For maximum $t_{max}$ number of generations total complexity becomes $O(N_{pop} \times m \times n^2 \times t_{max})$.

## 5. Experimental results

In this section, we report our experimental results. We conducted four experiments to evaluate the performance of the proposed method. In the first experiment, in order to evaluate the effectiveness of the proposed method MCLR, its performance has been compared with the nine document summarization methods. Moreover, MCLR has been compared with the average results of the best team of DUC (Document Understanding Conference) summarization track. In the second experiment, we studied the influence of weighting parameter $w$ on the proposed method. In the third experiment, we compared the efficiency of the methods. Finally, in the fourth experiment, we tested the statistical significance of the summarization results.

### 5.1. Data sets

The DUC (Document Understanding Conference), since 2008 known as TAC (Text Analysis Conference), has been the major forum for comparing summarization systems on a shared test set. Every year several summarization topics are released and the summaries produced by participants are evaluated, both automatically and manually. To evaluate the summarization results empirically, we use the DUC2002, DUC2004 and DUC2006 data sets, all of which are open benchmark data sets from DUC for generic automatic summarization evaluation. Table 1 gives a brief description of the data sets, in which data source indicates where the documents are obtained. For example, DUC2002 data come from the Text REtrieval Conference (TREC), and DUC2004 data are from the Topic Detection and Tracking (TDT) research.

### 5.2. Preprocessing

In the preprocessing step of generating document summaries, each document is decomposed into individual sentences using NLTK toolkit (NLTK toolkit), all stopwords are removed by using stopwords list provided in English stoplist (English stoplist)and word stemming is performed by Porter's algorithm (Porter stemming algorithm). A term-frequency vector for each sentence in the document is then constructed by Eq. (1).

### 5.3. Evaluation metrics

We carried out automatic evaluation of our summaries using ROUGE (Lin & Hovy, 2003) toolkit (i.e. ROUGE-1.5.5 in this study) for evaluation. ROUGE is a widely accepted metric for automatic evaluation of summarization tasks (DUC 2004–2007 and TAC 2008–2010). It measures summary quality by counting overlapping units such as the N-gram (ROUGE-N), word sequences (ROUGE-L and ROUGE-W) and word pairs (ROUGE-S and ROUGE-SU) between the candidate summary and the reference summary. ROUGE toolkit reports separate scores for N-grams (N = 1–4), longest common subsequence, weighted longest common subsequence co-occurrences and skip bigram co-occurrences. We showed four of the ROUGE metrics in the experimental results: ROUGE-1 (unigram), ROUGE-2 (bigram), ROUGE-L (longest common subsequence) and ROUGE-SU (skip bigram). Unigram and bigram statistics have been shown to have the highest correlation with human assessments (Lin & Hovy, 2003).

**Table 1**
Characteristics of the data sets.

| | DUC2002 | DUC2004 | DUC2006 |
|---|---|---|---|
| Number of clusters | 59 | 50 | 50 |
| Number of documents in each cluster | ~10 | 10 | 25 |
| Number of documents | 567 | 500 | 1250 |
| Data source | TREC | TDT | AQUAINT |
| Summary length | 200 words | 665 bytes | 250 words |

The ROUGE-N measure compares N-grams of two summaries, and counts the number of matches. This measure is computed as Lin and Hovy (2003):

$$\mathrm{ROUGE-N} = \frac{\sum_{S\in Summ_{ref}}\sum_{\text{N-gram}\in S} Count_{match}(\text{N-gram})}{\sum_{S\in Summ_{ref}}\sum_{\text{N-gram}\in S} Count(\text{N-gram})}, \tag{16}$$

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

ROUGE-L computes the ratio between the length of the summaries' longest common subsequence (LCS) and the length of the reference summary, as defined by Eq. (17):

$$P_{LCS}(R,S) = \frac{LCS(R,S)}{|S|}, \quad R_{LCS}(R,S) = \frac{LCS(R,S)}{|R|},$$
$$F_{LCS}(R,S) = \frac{(1+\beta^2)P_{LCS}(R,S)R_{LCS}(R,S)}{\beta^2 P_{LCS}(R,S) + R_{LCS}(R,S)}, \tag{17}$$

where $|R|$ and $|S|$ is the length of the reference $R$ and candidate $S$ sentence summaries, respectively. $LCS(R,S)$ is the length of a LCS of $R$ and $S$. $P_{LCS}(R,S)$ is the precision of $LCS(R,S)$, $R_{LCS}(R,S)$ is the recall of $LCS(R,S)$, and $\beta = P_{LCS}(R,S)/R_{LCS}(R,S)$.

Lin (Lin & Hovy, 2003) implemented two extensions to ROUGE-N: skip-bigram co-occurrence (ROUGE-S) and skip-bigram co-occurrence averaged with unigram co-occurrence (ROUGE-SU). The way ROUGE-S is calculated identical to ROUGE-2, except that skip bigrams are defined as subsequences rather than the regular definition of bigrams as substrings. Skip-bigram (skip-bigram is any pair of words in their sentence order, allowing for arbitrary gaps) co-occurrence statistics, ROUGE-S, measure the similarity of a pair of summaries based on how many skip-bigrams they have in common:

$$P_{SKIP2}(R,S) = \frac{SKIP2(R,S)}{C(|S|,2)},$$
$$R_{SKIP2}(R,S) = \frac{SKIP2(R,S)}{C(|R|,2)},$$
$$F_{SKIP2}(R,S) = \frac{(1+\beta^2)P_{SKIP2}(R,S)R_{SKIP2}(R,S)}{\beta^2 P_{SKIP2}(R,S) + R_{SKIP2}(R,S)}, \tag{18}$$

where $SKIP2(R,S)$ is the number of skip-bigram matches between $R$ and $S$,$\beta$ is the relative importance of $P_{SKIP2}(R,S)$ and $R_{SKIP2}(R,S)$, $P_{SKIP2}(R,S)$ being the precision of $SKIP2(R,S)$ and $R_{SKIP2}(R,S)$ the recall of $SKIP2(R,S)$. $C(\cdot,\cdot)$ is the combination function. One potential problem for ROUGE-S is that it does not give any credit to a candidate sentence if the sentence does not have any word pair co-occurring with its references. To accommodate this, ROUGE-S is extended with the addition of unigram as counting unit. The extended version is called ROUGE-SU that is a weighted average between ROUGE-S and ROUGE-1.

### 5.4. Experiment 1: performance comparison

First, we compare the implemented summarization method MCLR with the other summarization systems to examine the effectiveness of the method MCLR for summarization performance improvement. We implement the following most widely used document summarization methods as the baseline systems to compare with our proposed method MCLR.

1. *Random*: Randomly selects sentences for each topic.
2. *Centroid*: The centroid-based summarization usually includes the sentences of the highest similarities with all the other sentences in the documents into the summary, which is good since these sentences deliver the majority of information contained in the documents, however the redundancy needs to be further removed and the subtopics in the documents are hard to detect. The method applies MEAD algorithm (Radev et al., 2004) which extracts the most important sentences from a set of sentences based on the linear combination of three features, namely, the centroid score, the position score and the overlap-with-first score.
3. *LexRank*: The graph-based methods such as LexRank apply graph analysis and take the influence of other sentences into consideration, which provides a better view of the relationships embedded in the sentences. LexRank first builds a graph of all candidate sentences where nodes are the sentences and the edges are the cosine similarity values. Two candidate sentences are connected with an edge if the similarity between them is above a threshold. The system finds the most central sentences of the graph by performing a random walk on it (Erkan & Radev (2004)).
4. *LSA*: The method performs latent semantic analysis on terms by sentences matrix to select sentences having the greatest combined weights across all important topics (Gong & Liu, 2001).
5. *NMF*: The method performs non-negative matrix factorization on terms by sentences matrix and then ranks the sentences by their weighted scores (Lee et al., 2009). NMF can be viewed as a clustering method, which has many nice properties and advantages. Intuitively, this method clusters these sentences and chooses the most representative ones from each cluster to form the summary.
6. *KM*: The method calculates sentence similarity matrix using cosine similarity, performs *k*-means algorithm to clustering the sentences, and chooses the center sentences in each clusters (Wang et al., 2009).
7. *FGB*: This model uses both the term-document and term-sentence matrices to simultaneously cluster and summarize the documents. The model translates the clustering summarization problem into minimizing the Kullback–Leibler divergence between the given documents and model reconstructed terms. The minimization process results two matrices that represent the probabilities of the documents and sentences given clusters (topics). The document clusters are generated by assigning each document to the topic with the highest probability, and the summary is formed with the sentences with the high probability in each topic (Wang et al., 2011).
8. *BSTM*: The method by using both the term-document and term-sentence associations explicitly models the probability distributions of selecting sentences given topics and provides a principled way for the summarization task. An efficient variational Bayesian algorithm is derived for estimating model parameters. BSTM is similar to the FGB summarization since they are all based on sentence-based topic model. BSTM model is also related to 3-factor NMF model (Wang et al., 2009).
9. *SNMF + SLSS*: This summarization framework based on sentence-level semantic analysis (SLSS) and symmetric non-negative matrix factorization (SNMF). SLSS is able to capture the semantic relationships between sentences and SNMF can divide the sentences into groups for extraction (Wang et al., 2008).

These summarization methods are selected as the representatives of the most widely used types of summarization methods, and they are fundamentally different in both algorithm design and implementation, which makes them diverse and complimentary with each other.

**Table 2**
Overall performance comparison on DUC2002 data.

| Methods | ROUGE-1 | ROUGE-2 | ROUGE-L | ROUGE-SU |
|---|---|---|---|---|
| DUCbest | **0.4987 (1)** | **0.2523 (1)** | **0.4680 (1)** | **0.2841 (1)** |
| MCLR | 0.4965 (2) | 0.2493 (3) | 0.4632 (3) | 0.2799 (3) |
| Random | 0.3846 (11) | 0.1169 (11) | 0.3722 (11) | 0.1806 (11) |
| Centroid | 0.4538 (7) | 0.1918 (7) | 0.4324 (7) | 0.2363 (7) |
| LexRank | 0.4796 (6) | 0.2295 (6) | 0.4433 (6) | 0.2620 (6) |
| LSA | 0.4308 (10) | 0.1502 (10) | 0.4051 (9) | 0.2023 (9) |
| NMF | 0.4458 (8) | 0.1628 (8) | 0.4151 (8) | 0.2169 (8) |
| KM | 0.4316 (9) | 0.1514 (9) | 0.4038 (10) | 0.2014 (10) |
| FGB | 0.4851 (5) | 0.2410 (5) | 0.4508 (5) | 0.2686 (5) |
| BSTM | 0.4881 (4) | 0.2457 (4) | 0.4552 (4) | 0.2702 (4) |
| SNMF + SLSS | 0.4956 (3) | 0.2501 (2) | 0.4665 (2) | 0.2826 (2) |

**Table 3**
Overall performance comparison on DUC2004 data.

| Methods | ROUGE-1 | ROUGE-2 | ROUGE-L | ROUGE-SU |
|---|---|---|---|---|
| DUCbest | 0.3822 (5) | 0.0922 (3) | 0.3869 (4) | 0.1323 (3) |
| MCLR | **0.3946 (1)** | 0.0928 (2) | 0.3913 (2) | 0.1339 (2) |
| Random | 0.3187 (11) | 0.0635 (11) | 0.3452 (11) | 0.1178 (11) |
| Centroid | 0.3673 (8) | 0.0738 (7) | 0.3618 (8) | 0.1251 (8) |
| LexRank | 0.3784 (6) | 0.0857 (5) | 0.3753 (6) | 0.1310 (5) |
| LSA | 0.3415 (10) | 0.0654 (10) | 0.3497 (10) | 0.1195 (10) |
| NMF | 0.3675 (7) | 0.0726 (8) | 0.3675 (7) | 0.1292 (7) |
| KM | 0.3487 (9) | 0.0694 (9) | 0.3588 (9) | 0.1212 (9) |
| FGB | 0.3872 (4) | 0.0812 (6) | 0.3842 (5) | 0.1296 (6) |
| BSTM | 0.3907 (3) | 0.0901 (4) | 0.3880 (3) | 0.1322 (4) |
| SNMF + SLSS | 0.3942 (2) | **0.0937 (1)** | **0.3921 (1)** | **0.1343 (1)** |

The DE algorithm is stochastic in nature. Hence, it has been run several times. All of the results reported here are averaged over 20 runs. The parameters of the binary DE are set as follows: the population size, $N_{pop} = 50$; the number of iteration, $t_{\max} = 1000$; the crossover rate $cr = 0.65$.

Tables 2–4 show the ROUGE scores of different methods using DUC2002, DUC2004 and DUC2006 data sets, respectively. The higher the ROUGE scores, the better summarization performance. The bolded results highlight the best results in this set of experiments. The number in parentheses in each table slot shows the ranking of each method on a specific data set.

We observe that our MCLR model achieves high ROUGE scores and outperforms most of the baseline systems (except the best team in DUC2002 and the method SNMF + SLSS on DUC2004 and DUC2006). As seen from the results, on DUC2004 and DUC2006 the ROUGE-1 scores of our method MCLR are higher than the DUCbest and the SNMF + SLSS method and competitive with the best team from DUC2002. More importantly, our MCLR model, in terms of all ROUGE scores, outperforms the DUCbest in DUC2006 significantly. The ROUGE-2, ROUGE-L, and ROUGE-SU scores of MCLR in all data sets are competitive with the SNMF + SLSS method. It is

**Table 4**
Overall performance comparison on DUC2006 data.

| Methods | ROUGE-1 | ROUGE-2 | ROUGE-L | ROUGE-SU |
|---|---|---|---|---|
| DUCbest | 0.3796 (4) | 0.0754 (4) | 0.3476 (4) | 0.1321 (4) |
| MCLR | **0.3975 (1)** | 0.0850 (2) | 0.3674 (2) | 0.1385 (2) |
| Random | 0.3175 (11) | 0.0489 (11) | 0.2938 (11) | 0.1008 (11) |
| Centroid | 0.3639 (7) | 0.0726 (7) | 0.3425 (7) | 0.1262 (7) |
| LexRank | 0.3666 (6) | 0.0733 (6) | 0.3442 (6) | 0.1288 (6) |
| LSA | 0.3308 (9) | 0.0502 (10) | 0.3051 (9) | 0.1023 (10) |
| NMF | 0.3237 (10) | 0.0550 (9) | 0.3006 (10) | 0.1061 (9) |
| KM | 0.3637 (8) | 0.0618 (8) | 0.3411 (8) | 0.1250 (8) |
| FGB | 0.3713 (5) | 0.0748 (5) | 0.3450 (5) | 0.1308 (5) |
| BSTM | 0.3898 (3) | 0.0848 (3) | 0.3525 (3) | 0.1365 (3) |
| SNMF + SLSS | 0.3955 (2) | **0.0855 (1)** | **0.3680 (1)** | **0.1398 (1)** |

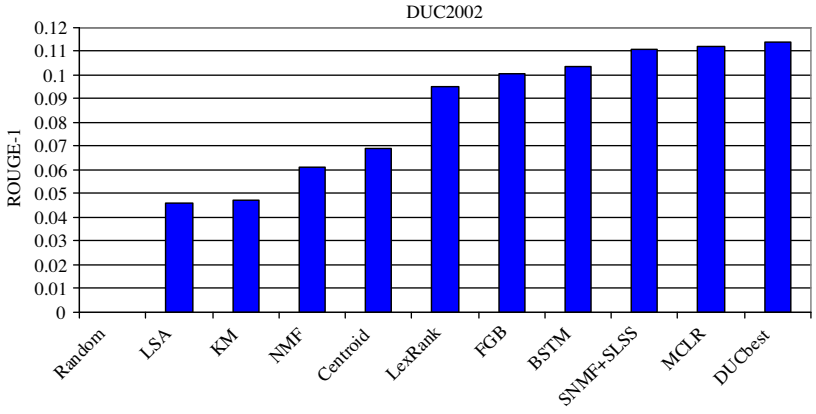


**Fig. 1.** Comparison of the methods using DUC2002.

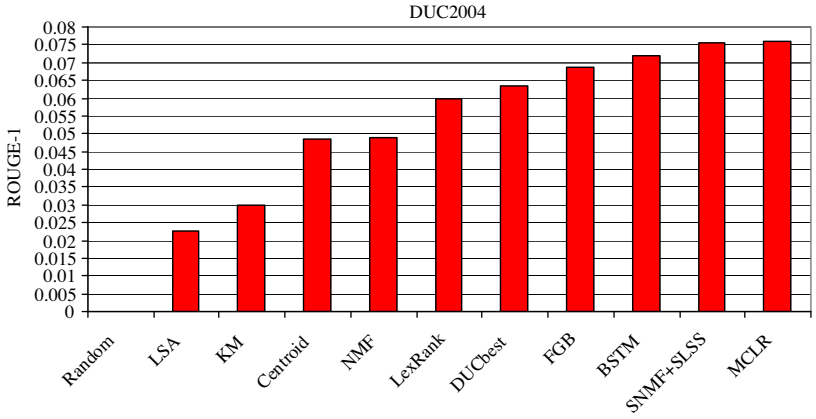


**Fig. 2.** Comparison of the methods using DUC2004.

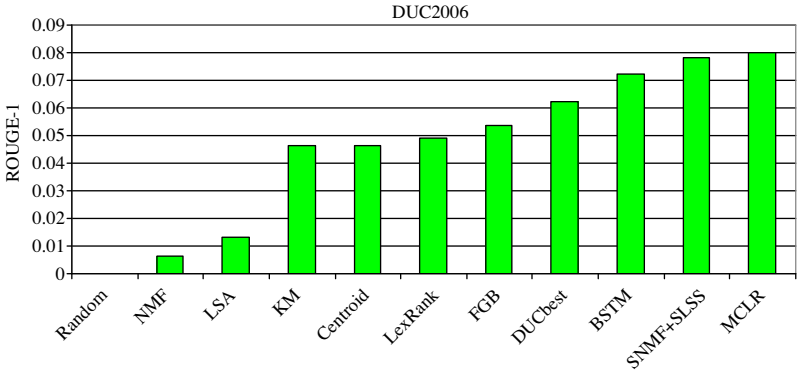


**Fig. 3.** Comparison of the methods using DUC2006.

necessary note that the good performance of the SNMF + SLSS benefits from the sentence-level semantic understanding, the clustering over symmetric similarity matrix by the SNMF algorithm, and the within-cluster sentence selection using both internal (e.g., the computed similarity between sentences) and external information (e.g., the given topic information). We also note that our MCLR model does not make use of any external information; while the SNMF + SLSS usually depends on some external knowledge, for example, SNMF + SLSS employs WordNet for discovering semantic similarity between words. Although we can spend more efforts on the preprocessing or language-processing step, our goal here is to demonstrate the effectiveness of optimization-based document summarization approach and hence we do not utilize advanced NLP techniques any external information for preprocessing. The good results of the best team come from the fact that they extract the topic information of the document set in an ad hoc manner, and utilize advanced NLP techniques to resolve pronouns and other anaphoric expressions which is not applied in other implemented methods.

To visually illustrate the comparison, we use Figs. 1–3. We subtract the Random score from the scores of all the other methods in these figures so that the difference can be observed more clearly.

**Table 5**
The resultant rank of the methods.

| Methods | $R_r$ = | | | | | | | | | | | Resultant rank |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | |
| SNMF + SLSS | 6 | 5 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 11.4 |
| MCLR | 2 | 7 | 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 10.8 |
| DUCbest | 4 | 0 | 2 | 5 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 9.9 |
| BSTM | 0 | 0 | 6 | 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 9.3 |
| FGB | 0 | 0 | 0 | 1 | 9 | 2 | 0 | 0 | 0 | 0 | 0 | 7.5 |
| LexRank | 0 | 0 | 0 | 0 | 2 | 10 | 0 | 0 | 0 | 0 | 0 | 6.7 |
| Centroid | 0 | 0 | 0 | 0 | 0 | 0 | 9 | 3 | 0 | 0 | 0 | 5.2 |
| NMF | 0 | 0 | 0 | 0 | 0 | 0 | 3 | 5 | 2 | 2 | 0 | 4.1 |
| KM | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 | 6 | 2 | 0 | 3.5 |
| LSA | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 | 8 | 0 | 2.5 |
| Random | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 12 | 1.1 |

As we have similar conclusion on different ROUGE scores, we only show the ROUGE-1 results in these figures.

To obtain the resulting ranks of the methods we transformed Tables 2–4 into another one, shown in Table 5. The resultant rank in Table 5 (last column) was computed according to the following formula (Aliguliyev, 2009):

$$\text{Rank}(method) = \sum_{r=1}^{11} \frac{(11 - r + 1)R_r}{11}, \tag{19}$$

where $R_r$ denotes the number of times the method appears in the $r$th rank.

Table 5 demonstrates overall comparison of the summarization methods. From the results, we observe the following:

- Random method provides the worst performance, as expected.
- The new methods SNMF + SLSS, BSTM and FGB, proposed in recent years, greatly improve the summarization results by using various advanced techniques such as semantic analysis and document structure.
- MCLR achieves high performance and outperforms most of the baseline systems, and is comparable with newly developed method SNMF + SLSS and the bets DUC participant.
- The widely used clustering-based summarization method NMF can improve important sentence selection.
- The FGB method outperforms most of the baseline systems. This is because, as stated in Wang et al. (2011), in FGB model, the factorization results contain both the sentence-topic matrix, from which it chooses the sentences with the highest probabilities in each topic to form the summaries and the document-topic matrix, from which it can get the document clusters. Since the topics are generated from both the document side and the sentence side, the document-level and the sentence-level information will influence each other. Therefore, the sentences used for document summarization are not treated independently, as do many of the existing methods.
- BSTM outperforms other NMF-based methods, FGB and NMF, since the document-topic allocation is marginalized out in BSTM and the marginalization increases the stability of the estimation of the sentence-topic parameters.
- NMF shows the best performance than LSA method, because it uses more intuitively interpretable semantic features and grasp the innate structure of documents. NMF's use of semantic features allows it to identify subtopics of documents more successfully than the LSA method.
- LexRank outperforms Centroid. This is because LexRank ranks the sentence using eigenvector centrality that implicitly accounts for information subsumption among all sentences.
- The Centroid system outperforms clustering-based summarization NMF and KM methods. This is mainly because the Centroid based algorithm takes into account positional value and first-sentence overlap that are not used in clustering-based summarization.
- The SNMF + SLSS method outperforms other NMF-based methods, BSTM, FGB, and NMF because it takes into account semantic relationships between sentences.
- On large data set DUC2006, the method KM shows the best results than the method NMF. On the contrary, on small data sets (DUC2002 and DUC2004) the method NMF outperforms KM.

The experimental results indicate that the optimization-based approach for document summarization is truly a promising research direction. It is valuable to note that a real optimization based summarization method is different from the existing non-optimization based methods in two noteworthy aspects. First, it ranks summaries instead of ranking individual sentences. Second, though ignored in the previous literature, the approach to rank summaries should not directly rely on the approach to rank sentences. Otherwise, the optimization solutions will degenerate to the traditional non-optimization based (e.g. MMR like) methods.

As seen from the results, shown in Tables 2–4, improvement of the method SNMF + SLSS and DUCbest compared with the MCLR method are slightly. For example, compared with the MCLR method on DUC2006 data set the SNMF + SLSS method improves the performance by 0.59%, 0.16% and 0.94% in terms ROUGE-2, ROUGE-L and ROUGE-SU metrics, respectively. Under our analysis with a modification of DE algorithm, a weighting of sentences and

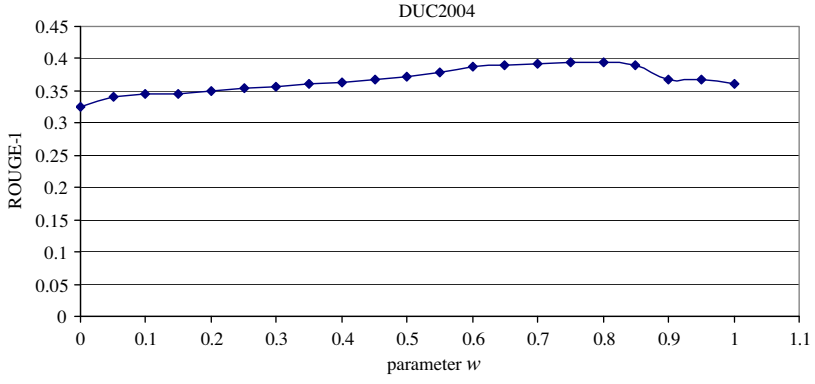


**Fig. 4.** ROUGE-1 performance of hybrid method $f$ vs. $w$ on DUC2004.

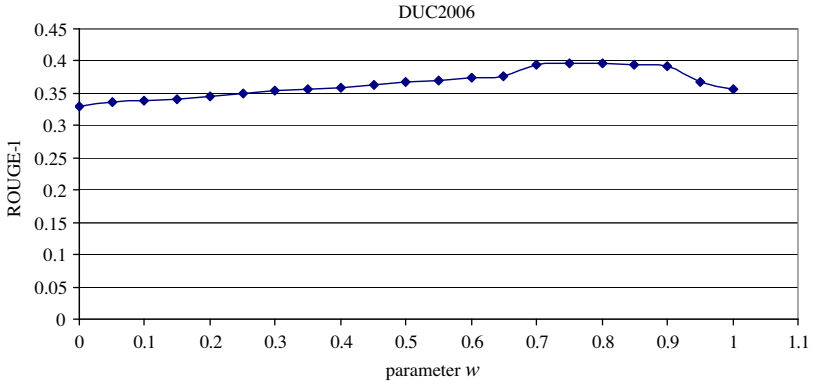


**Fig. 5.** ROUGE-1 performance of hybrid method $f$ vs. $w$ on DUC2006.

**Table 6**
Comparison of the methods on time spent.

| Methods | DUC2002 (min) | DUC2004 (min) | DUC2006 (min) |
|---|---|---|---|
| Centroid | 32.7 | 28.8 | 97.4 |
| LexRank | 17.2 | 16.1 | 29.3 |
| LSA | 21.3 | 20.2 | 45.6 |
| NMF | 32.4 | 30.9 | 58.3 |
| KM | 15.4 | 13.6 | 23.3 |
| FGB | 33.9 | 31.7 | 59.6 |
| BSTM | 36.7 | 33.1 | 58.5 |
| SNMF + SLSS | 37.4 | 33.2 | 60.4 |
| MCLR | 14.9 | 13.3 | 22.7 |

a suitable choice of similarity measure, we without employing any external knowledge can improve the results of our method. We plan to conduct this research direction in future work.

### 5.5. Experiment 2: discussion on parameter w

In order to investigate how the relative contributions from the content coverage objective function $f_{cov}(\cdot)$ (5) and the redundancy objective function $f_{red}(\cdot)$ (6) influence the summarization performance of the objective function $f(\cdot)$ (7), Figs. 4 and 5 show the ROUGE-1 values of the method $f(\cdot)$ with respect to different values of the combining weight $w$ on DUC2004 and DUC2006 data sets, respectively. Here, $w$ is adjusted from 0 to 1 in every 0.1 interval and the results show that combining both $f_{cov}$ and $f_{red}$ objectives leads to better performance. As shown in Figs. 4 and 5, when $w$ is greater than 0, better summarization performance were observed, compared to that with $w = 0$ (method $f_{red}$). We observe that when $w$ is 0.75, the performance is the best. We see also that the method $f_{cov}$ (corresponds to $w = 1$) shows the best result, than the method $f_{red}$ (corresponds to $w = 0$).

### 5.6. Efficiency

The efficiency of the algorithm computation is an important factor. Our evaluations are performed by Delphi 7 on a Server running Windows Vista with two Dual-Core Intel Xeon CPU 4 GHz and 4 Gb memory. Table 6 shows the comparison in terms of CPU time spent by each method.

From the experimental results shown in Table 6, we clearly observe that (1) Centroid method performs very slowly on large document corpus DUC2006; (2) the methods NMF, FGB, BSTM, and SNMF + SLSS spend almost equal CPU time on all data sets; (3) the results demonstrate the high efficiency of MCLR. What is more, the computational speed of the method MCLR, measured by CPU time, is distinctly faster than that of SNMF + SLSS, despite of the fact that the method MCLR concedes to the method SNMF + SLSS in terms of ROUGE values; (4) Comparing MCLR to KM and LSA on large data set DUC2006, we find that MCLR is as fast as KM and much faster than LSA.

### 5.7. Statistical significance test

In order to statistically compare the performance of MCLR with other summarization methods, we use a non-parametric statistical significance test, called Wilcoxon's matched-pairs signed rank based statistical test, to determine the significance of our results. The statistical significance test for independent samples has been conducted at the 5% significance level of the summarization results (Hollander & Wolfe, 1999). Ten groups, corresponding to the ten methods: 1. MCLR, 2. Random, 3. Centroid, 4. LexRank, 5. LSA, 6. NMF, 7. KM, 8. FGB, 9. BSTM, 10. SNMF + SLSS, have been created for each data set. Two groups are compared at a time one corresponding to MCLR method and the other corresponding to some other method considered in this paper. Each group consists of

**Table 7**
Median values of ROUGE-1 scores and standard deviation over 20 consecutive runs of methods.

| Methods | DUC2002 | | | DUC2004 | | | DUC2006 | | |
|---|---|---|---|---|---|---|---|---|---|
| | Median | 95% CI | Stdv. | Median | 95% CI | Stdv. | Median | 95% CI | Stdv. |
| MCLR | **0.4968** | **[0.4962, 0.4973]** | **1.2e−3** | **0.3969** | **[0.3961, 0.3976]** | **1.5e−3** | **0.3974** | **[0.3959, 0.3976]** | **1.8e−3** |
| Random | 0.3911 | [0.3876, 0.3957] | 8.7e−3 | 0.3224 | [0.3148, 0.3242] | 1.0e−2 | 0.3183 | [0.3118, 0.3217] | 1.1e−2 |
| Centroid | 0.4539 | [0.4509, 0.4558] | 5.2e−3 | 0.3672 | [0.3651, 0.3714] | 6.8e−3 | 0.3648 | [0.3609, 0.3677] | 7.3e−3 |
| LexRank | 0.4813 | [0.4785, 0.4838] | 5.6e−3 | 0.3776 | [0.3749, 0.3810] | 6.6e−3 | 0.3617 | [0.3614, 0.3678] | 6.8e−3 |
| LSA | 0.4282 | [0.4256, 0.4297] | 4.4e−3 | 0.3421 | [0.3392, 0.3445] | 5.7e−3 | 0.3303 | [0.3279, 0.3335] | 6.0e−3 |
| NMF | 0.4430 | [0.4417, 0.4481] | 6.8e−3 | 0.3699 | [0.3652, 0.3716] | 6.8e−3 | 0.3203 | [0.3199, 0.3255] | 6.0e−3 |
| KM | 0.4288 | [0.4277, 0.4339] | 6.6e−3 | 0.3456 | [0.3434, 0.3486] | 5.5e−3 | 0.3632 | [0.3609, 0.3657] | 5.1e−3 |
| FGB | 0.4887 | [0.4850, 0.4898] | 5.1e−3 | 0.3832 | [0.3814, 0.3866] | 5.5e−3 | 0.3684 | [0.3676, 0.3734] | 6.2e−3 |
| BSTM | 0.4892 | [0.4866, 0.4921] | 5.8e−3 | 0.3882 | [0.3863, 0.3928 | 7.0e−3 | 0.3924 | [0.3879, 0.3940] | 6.5e−3 |
| SNMF + SLSS | 0.4949 | [0.4939, 0.4960] | 2.3e−3 | 0.3934 | [0.3924, 0.3953] | 3.2e−3 | 0.3917 | [0.3896, 0.3938] | 4.6e−3 |

**Table 8**
Median values of ROUGE-2 scores and standard deviation over 20 consecutive runs of methods.

| Methods | DUC2002 | | | DUC2004 | | | DUC2006 | | |
|---|---|---|---|---|---|---|---|---|---|
| | Median | 95% CI | Stdv. | Median | 95% CI | Stdv. | Median | 95% CI | Stdv. |
| MCLR | 0.2494 | **[0.2485,0.2502]** | **1.1e−3** | 0.0927 | **[0.0912,0.0929]** | **1.8e−3** | 0.0852 | **[0.0848,0.0857]** | **9.7e−4** |
| Random | 0.1148 | [0.1109,0.1215] | 1.1e−2 | 0.0648 | [0.0626,0.0688] | 6.7e−3 | 0.0493 | [0.0469] | 5.1e−3 |
| Centroid | 0.1917 | [0.1896,0.1944] | 5.2e−3 | 0.0729 | [0.0713,0.0748] | 3.8e−3 | 0.0722 | [0.0713,0.0734] | 2.2e−3 |
| LexRank | 0.2329 | [0.2301,0.2353] | 5.5e−3 | 0.0839 | [0.0835,0.0858] | 2.2e−3 | 0.0739 | [0.0727,0.0750] | 2.5e−3 |
| LSA | 0.1532 | [0.1503,0.1559] | 6.1e−3 | 0.0654 | [0.0644,0.0676] | 3.4e−3 | 0.0507 | [0.0483,0.0524] | 4.4e−3 |
| NMF | 0.1619 | [0.1596,0.1656] | 6.4e−3 | 0.0743 | [0.0723,0.0752] | 3.1e−3 | 0.0568 | [0.0550,0.0572] | 2.4e−3 |
| KM | 0.1520 | [0.1493,0.1555] | 6.6e−3 | 0.0692 | [0.0664,0.0707] | 4.6e−3 | 0.0615 | [0.0607,0.0637] | 3.3e−3 |
| FGB | 0.2396 | [0.2357,0.2436] | 8.5e−3 | 0.0835 | [0.0815,0.0852] | 3.9e−3 | 0.0734 | [0.0724,0.0752] | 3.1e−3 |
| BSTM | 0.2455 | [0.2433,0.2483] | 5.4e−3 | 0.0903 | [0.0889,0.0914] | 2.6e−3 | 0.0838 | [0.0825,0.0847] | 2.4e−3 |
| SNMF + SLSS | **0.2514** | [0.2498,0.2517] | 2.1e−3 | **0.0946** | [0.0932,0.0955] | 2.5e−3 | **0.0869** | [0.0857,0.0871] | 1.5e−3 |

**Table 9**
*P-values* produced by Wilcoxon's matched-pairs signed rank test by comparing MCLR with other methods.

| Data set | Random | Centroid | LexRank | LSA | NMF | KM | FGB | BSTM | SNMF + SLSS |
|---|---|---|---|---|---|---|---|---|---|
| | Comparing medians of ROUGE-1 metric of MCLR with other methods | | | | | | | | |
| DUC2002 | <0.0001 | <0.0001 | <0.0001 | <0.0001 | < 0.0001 | <0.0001 | <0.0001 | 0.0002 | 0.0025 |
| DUC2004 | <0.0001 | <0.0001 | <0.0001 | <0.0001 | < 0.0001 | <0.0001 | <0.0001 | 0.0005 | 0.0021 |
| DUC2006 | <0.0001 | <0.0001 | <0.0001 | <0.0001 | < 0.0001 | <0.0001 | <0.0001 | 0.0015 | 0.0010 |
| | Comparing medians of ROUGE-1 metric of MCLR with other methods | | | | | | | | |
| DUC2002 | <0.0001 | <0.0001 | <0.0001 | <0.0001 | < 0.0001 | <0.0001 | 0.0003 | 0.0047 | 0.0039 |
| DUC2004 | <0.0001 | <0.0001 | <0.0001 | <0.0001 | < 0.0001 | <0.0001 | <0.0001 | 0.0026 | 0.0042 |
| DUC2006 | <0.0001 | <0.0001 | <0.0001 | <0.0001 | < 0.0001 | <0.0001 | <0.0001 | 0.0042 | 0.0024 |

the ROUGE-1 and ROUGE-2 scores for the data sets produced by 20 consecutive runs of the corresponding method. The median values, 95% confidence interval (CI), and standard deviation (Stdv.) of ROUGE-1 and ROUGE-2 scores of each method for all the data sets are shown in Tables 7 and 8, respectively.

As is evident from Table 7 the median values of ROUGE-1 for MCLR method on all data sets are better than that for the other methods, whereas, from Table 8 we observe that the median values of ROUGE-2 for MCLR method on all data sets are better than that for the other methods except the method SNMF + SLSS. To establish that this goodness is statistically significant, Table 9 reports the *P-values* produced by Wilcoxon's matched-pairs signed rank test for comparison of two groups (one group corresponding to MCLR and another group corresponding to some other algorithm) at a time (GraphPad Software). As a null hypothesis, it is assumed that there are no significant differences between the median values of two groups. Whereas, the alternative hypothesis is that there is significant difference in the median values of the two groups. It is clear from Table 9 that *P-values* are much less than 0.05 (5% significance level). For example, the Wilcoxon's matched-pairs signed rank test between the algorithms MCLR and SNMF + SLSS for DUC2002 provides a *P-value* of 0.0025 (ROUGE-1), which is very small. This is strong evidence against the null hypothesis, indicating that the better median values of the performance metrics produced by MCLR is statistically significant and has not occurred by chance. Similar results are obtained for all other data sets and for all other methods compared to MCLR method, establishing the significant superiority of the proposed technique.

From the statistical results, we observe that our MCLR model significantly outperforms the other baseline summarization methods. A visual comparison of statistical significance is provided in Figs. 6 and 7. Figs. 6 and 7 show the median values and the change of ROUGE-1 and ROUGE-2 scores obtained by each method on the benchmark data sets, respectively. It can be observed that the change of ROUGE-1 and ROUGE-2 scores of MCLR is noticeably better than that of other methods. In addition, according to the statistical significance test, MCLR is more stable than the other methods. For convincing, we address the readers to pay an attention to the values of the standard deviation (Stdv.) and confidence intervals (95% CI) reported in Tables 7 and 8. The best results are shown in bold.

## 6. Conclusion and future work

With the explosive growth of the volume and complexity of document data (e.g., news, blogs, web pages) on the Internet and electronic government multi-document summarization provides a useful solution for understanding documents and reducing information overload. Thus, multi-document summarization has attracted much attention in recent years, and many applications have been developed. Multi-document summarization aims to generate a compressed summary by extracting the major information in a collection of documents sharing the same or similar topics. In multi-document summarization, the risk of extracting two sentences conveying the same information is greater than in a single-document summarization problematic. Moreover, identifying redundancy is a critical task, as information appearing several times in different documents can be qualified as important. Hence, we need effective summarization methods to analyze and extract the important information. A good summary is expected to preserve the topic information contained in the documents as much as possible, and at the same time to contain as little redundancy as possible, known as information richness and diversity, respectively. The requirement raises a fundamental problem: how important will a selected summary be to represent the whole documents?

This paper discusses work on multi-document summarization to create a generic extractive summary of multiple documents on the same (or related) topic. The proposed approach adopts a broadly used summarization model – sentence extraction – to extract important sentences and compose them into a summary. This

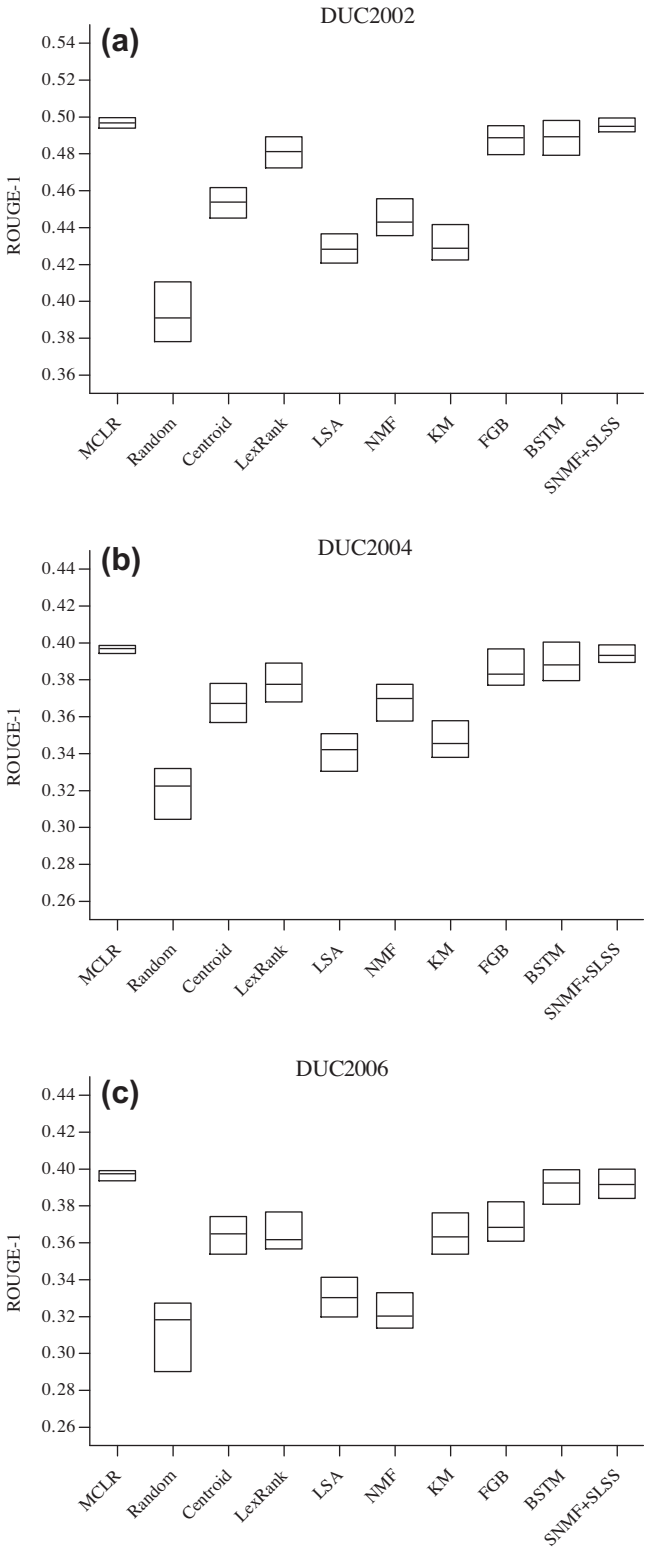


**Fig. 6.** Change of ROUGE-1 for different summarization method on (a) DUC2002, (b) DUC2004 and (c) DUC2006.

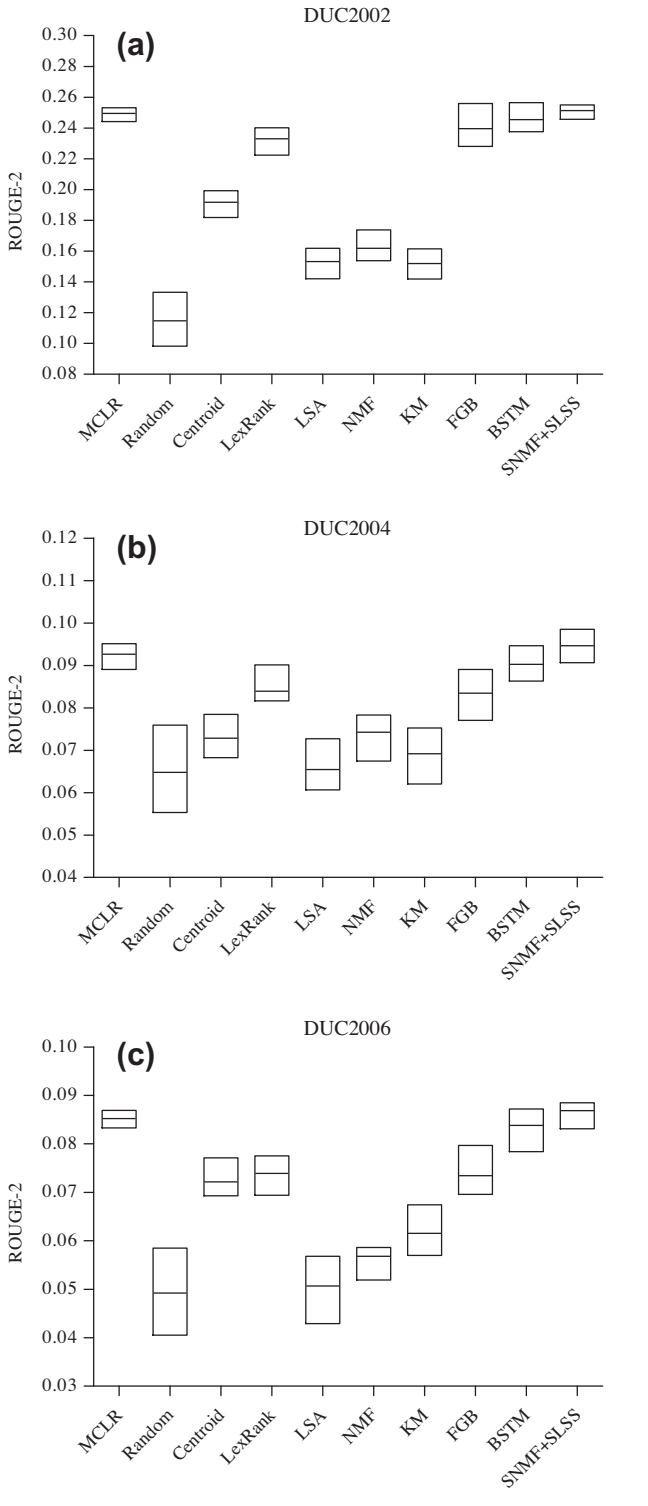


**Fig. 7.** Change of ROUGE-2 for different summarization methods on (a) DUC2002, (b) DUC2004 and (c) DUC2006.

approach divides the multi-document summarization task into three subtasks: (1) evaluating sentences according to their importance of being part in the summary by calculating their similarity to the center of sentences collection, (2) eliminating redundancy while extracting the most important sentences, and (3) organizing extracted sentences into a summary.

We present a multi-document summarization model which extracts key sentences from given documents while reducing redundant information in the summaries. The model is represented as a QBP problem that was solved by using a binary differential evolution algorithm. We showed that the resulting summarization system based on the proposed optimization approach is competitive on the DUC2002, DUC2004 and DUC2006 data sets. The

experimental results provide strong evidence that our method is a viable method for document summarization.

In future work, we will further improve our approach mainly in three ways: firstly, weighting of sentences will be studied; then other modification of DE algorithm will be developed in order to find the best summary more effectively. In future, we plan also to experiment our approach with the different similarity measures to check how it performs.

## References


Alguliev, R. M., & Aliguliyev, R. M. (2008). Automatic text documents summarization through sentences clustering. *Journal of Automation and Information Sciences, 40*(9), 53–63.

Alguliev, R. M., & Aliguliyev, R. M. (2009). Evolutionary algorithm for extractive text summarization. *Intelligent Information Management, 1*(2), 128–138.

Alguliev, R. M., Aliguliyev, R. M., Hajirahimova, M. S., & Mehdiyev, C. A. (2011). MCMR: maximum coverage and minimum redundant text summarization model. *Expert Systems with Applications, 38*(12), 14514–14522.

Aliguliyev, R. M. (2010). Clustering techniques and discrete particle swarm optimization algorithm for multi-document summarization. *Computational Intelligence, 26*(4), 420–448.

Aliguliyev, R. M. (2009). Performance evaluation of density-based clustering methods. *Information Sciences, 179*(20), 3583–3602.

Aliguliyev, R. M. (2009). A new sentence similarity measure and sentence based extractive technique for automatic text summarization. *Expert Systems with Applications, 36*(4), 7764–7772.

Antiqueira, L., Oliveira, O. N., Costa, L. F., & Nunes, M. G. V. (2009). A complex network approach to text summarization. *Information Sciences, 179*(5), 584–599.

Bhattacharya, S., Ha-Thuc, V., & Srinivasan, P. (2011). MeSH: A window into full text for document summarization. *Bioinformatics, 27*(13), i120–i128.

Binwahlan, M. S., Salim, N., & Suanmali, L. (2010). Fuzzy swarm diversity hybrid model for text summarization. *Information Processing & Management, 46*(5), 571–588.

Binwahlan, M. S., Salim, N., & Suanmali, L. (2009). MMI diversity based text summarization. *International Journal of Computer Science and Security, 3*(1), 23–33.

Cai, X., & Li, W. (2011). A spectral analysis approach to document summarization: Clustering and ranking sentences simultaneously. *Information Sciences, 181*(18), 3816–3827.

Carbonell, J., & Goldstein, J. (1998). The use of MMR, diversity-based re-ranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on research and development in information retrieval, Melbourne, Australia, August 24–28* (pp. 335–336).

Chali, Y., Hasan, S. A., & Joty, S. R. (2011). Improving graph-based random walks for complex question answering using syntactic, shallow semantic and extended string subsequence kernels. *Information Processing & Management, 47*(6), 843–855.

Das, S., & Suganthan, P. N. (2011). Differential evolution: A survey of the state-of-the-art. *IEEE Transactions on Evolutionary Computation, 15*(1), 4–31.

Das, S., & Sil, S. (2010). Kernel-induced fuzzy clustering of image pixels with an improved differential evolution algorithm. *Information Sciences, 180*(8), 1237–1256.

Erkan, G., & Radev, D. (2004). Lexrank: Graph-based centrality as salience in text summarization. *Journal of Artificial Intelligence Research, 22*, 457–479.

Filatova, E., & Hatzivassiloglou, V. (2004). A formal model for information selection in multi-sentence text extraction. In *Proceedings of the 20th international conference on computational linguistics (COLING'04), Geneva, Switzerland, August 23–27* (pp. 397–403).

Gong, Y., & Liu, X. (2001). Generic text summarization using relevance measure and latent semantic analysis. In *Proceedings of the 24th annual international ACM SIGIR conference on research and development in information retrieval, New Orleans, USA, September 9–12* (pp. 19–25).

He, R., Qin, B., & Liu, T. (2012). A novel approach to update summarization using evolutionary manifold-ranking and spectral clustering. *Expert Systems with Applications, 39*(3), 2375–2384.

Hollander, M., & Wolfe, D. A. (1999). *Nonparametric statistical methods* (2nd ed.). Wiley-Interscience (p.787).

Huang, L., He, Y., Wei, F., & Li, W. (2010). Modeling document summarization as multi-objective optimization. In *Proceedings of the third international symposium on intelligent information technology and security informatics, Jinggangshan, China, April 02–04* (pp. 382–386).

Huang, H.-H., Yang, H.-C. & Kuo, Y.-H. (2009). A fuzzy-rough hybrid approach to multi-document extractive summarization. In *Proceedings of the 2009 ninth international conference on hybrid intelligent systems, Shenyang, China, August 12–14* (pp. 168–173).

Hung, S.-Y., Tang, K.-Z., Chang, C.-M., & Ke, C.-D. (2009). User acceptance of intergovernmental services: An example of electronic document management system. *Government Information Quarterly, 26*(2), 387–397.

Islam, A., & Inkpen, D. (2008). Semantic text similarity using corpus-based word similarity and string similarity. *ACM Transactions on Knowledge Discovery from Data, 2*(2), 10:1–10:25.

Ko, Y., & Seo, J. (2008). An effective sentence-extraction technique using contextual information and statistical approaches for text summarization. *Pattern Recognition Letters, 29*(9), 1366–1371.

Kutlu, M., Cigir, C., & Cicekli, I. (2010). Generic text summarization for Turkish. *The Computer Journal, 53*(8), 1315–1323.

Lee, J.-H., Park, S., Ahn, C.-M., & Kim, D. (2009). Automatic generic document summarization based on non-negative matrix factorization. *Information Processing & Management, 45*(1), 20–34.

C.-Y., Lin, & Hovy, E. (2003). Automatic evaluation of summaries using N-gram co-occurrence statistics. *Proceedings of the 2003 conference of the north american chapter of the association for computational linguistics on human language technology* (1, pp. 71–78). Morristown, NJ, USA: Association for Computational Linguistics.

Lin, J., Madnani, N., & Dorr, B. (2010). Putting the user in the loop: Interactive maximal marginal relevance for query-focused summarization. In *Proceedings of the 11th annual conference of the north american chapter of the association for computational linguistics, Los Angeles, USA, June 1–6* (pp. 305–308).

Lu, Y., Zhou, J., Qin, H., Li, Y., & Zhang, Y. (2010). An adaptive hybrid differential evolution algorithm for dynamic economic dispatch with valve-point effects. *Expert Systems with Applications, 37*(7), 4842–4849.

Mani, I., & Maybury, M. T. (1999). *Advances in automatic text summarization*. Cambridge: MIT Press.

McDonald, R. (2007). A study of global inference algorithms in multi-document summarization. In *Proceedings of 29th European conference on IR research*. *LNCS* (25, pp. 557–564). Rome, Italy: Springer-Verlag.

Ouyang, Y., Li, W., Li, S., & Lu, Q. (2011). Applying regression models to query-focused multi-document summarization. *Information Processing & Management, 47*(2), 227–237.

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

Rashedi, E., Nezamabadi-pour, H., & Saryazdi, S. (2009). GSA: A gravitational search algorithm. *Information Sciences, 179*(13), 2232–2248.

Takamura, H., & Okumura, M. (2009). Text summarization model based on the budgeted median problem. In *Proceedings of the 18th ACM international conference on information and knowledge management, Hong Kong, China, November 2–6* (pp. 1589–1592).

Tang, J., Yao, L., & Chen, D. (2009). Multi-topic based query-oriented summarization. In *Proceedings of the 9th SIAM international conference on data mining, Nevada, USA, April 30–May 2* (pp. 1148–1159).

Tsai, F. S., Tang, W., & Chan, K. L. (2010). Evaluation of novelty metrics for sentence-level novelty mining. *Information Sciences, 180*(12), 2359–2374.

Wan, X., & Xiao, J. (2010). Exploiting neighborhood knowledge for single document summarization and keyphrase extraction. *ACM Transactions on Information Systems, 28*(2), 8:1–8:3.

Wang, D., & Li, T. (2010). Document update summarization using incremental hierarchical clustering. In *Proceedings of the ACM 19th conference on information and knowledge management, Toronto, Canada, October 26–30* (pp. 279–287).

Wang, Y., Li, B., & Weise, T. (2010). Estimation of distribution and differential evolution cooperation for large scale economic load dispatch optimization of power systems. *Information Sciences, 180*(12), 2405–2420.

Wang, D., Li, T., Zhu, S., & Ding, C. (2008). Multi-document summarization via sentence-level semantic analysis and symmetric matrix factorization. In *Proceedings of the 31st annual international ACM SIGIR conference on research and development in information retrieval, Singapore, July 20–24* (pp. 307–314).

Wang, D., Zhu, S., Li, T., & Gong, Y. (2009). Multi-document summarization using sentence-based topic models. In *Proceedings of the ACL-IJCNLP 2009 conference short papers, Singapore, August 04* (pp. 297–300).

Wang, D., Zhu, S., Li, T., Chi, Y., & Gong, Y. (2011). Integrating document clustering and multidocument summarization. *ACM Transactions on Knowledge Discovery from Data, 5*(3), 14:1–14:26.

Wenyin, L., Quan, X., Feng, M., & Qiu, B. (2010). A short text modeling method combining semantic and statistical information. *Information Sciences, 180*(20), 4031–4041.

Zhang, M., Luo, W., & Wang, X. (2008). Differential evolution with dynamic stochastic selection for constrained optimization. *Information Sciences, 178*(15), 3043–3074.

Zielinski, K., Peters, D., & Laur, R. (2005). Runtime analysis regarding stopping criteria for differential evolution and particle swarm optimization. In *Proceedings of the 1st international conference on experiments/process/system modelling/simulation/optimization, Athens, Greece, July 6–9*.

Document Understanding Conference. <http://duc.nist.gov>.

English stoplist. <ftp://ftp.cs.cornell.edu/pub/smart/english.stop>.

GraphPad Software. <http://www.graphpad.com/>.

Natural Language Toolkit. <http://nltk.org>.

Porter stemming algorithm. <http://www.tartarus.org/martin/PorterStemmer/>.