Appl Intell
DOI 10.1007/s10489-012-0336-1

Multi-document summarization via submodularity

Jingxuan Li · Lei Li · Tao Li

Abstract Multi-document summarization is becoming an important issue in the Information Retrieval community. It aims to distill the most important information from a set of documents to generate a compressed summary. Given a set of documents as input, most of existing multi-document summarization approaches utilize different sentence selection techniques to extract a set of sentences from the document set as the summary. The submodularity hidden in the *term coverage* and the *textual-unit similarity* motivates us to incorporate this property into our solution to multi-document summarization tasks. In this paper, we propose a new principled and versatile framework for different multi-document summarization tasks using submodular functions (Nemhauser et al. in Math. Prog. 14(1):265–294, 1978) based on the term coverage and the textual-unit similarity which can be efficiently optimized through the improved greedy algorithm. We show that four known summarization tasks, including generic, query-focused, update, and comparative summarization, can be modeled as different variations derived from the proposed framework. Experiments on benchmark summarization data sets (e.g., DUC04-06, TAC08, TDT2 corpora) are conducted to demonstrate the efficacy and effectiveness of our proposed framework for the general multi-document summarization tasks.

J. Li · L. Li · T. Li (✉)
School of Computing and Information Sciences, Florida International University, 11200 SW 8th St, Miami, FL 33199, USA
e-mail: taoli@cs.fiu.edu

J. Li
e-mail: jli003@cs.fiu.edu

L. Li
e-mail: lli003@cs.fiu.edu

Published online: 09 February 2012

Keywords Multi-document summarization · Submodularity · Greedy algorithm

1 Introduction

Multi-document summarization, as a fundamental and effective pattern for the document understanding and organization, enables better information services by creating concise and informative reports for a large collection of documents. It is useful for many real world applications. For example, Chen and Liu [1] aimed at tracking user-interested news events from a large pool of news. In this case, multi-document summarization can be applied to summarize those news events. A huge amount of available online textual documents in the field of biomedicine leads to great difficulties for building question answering, or information retrieval systems [5]. Luckily, multi-document summarization can assist extracting the essential information from those documents and hereby benefit those systems.

Notice that the generated summary is either generic where the important information contained in input documents without any particular information needs is extracted, or query/topic-focused in which it is produced in response to a user query [10, 16]. Recently, new summarization tasks such as the update summarization, [3] and the comparative summarization [28] have also been proposed. The update summarization aims to generate short summaries of recent documents to capture new information different from earlier documents, and the comparative summarization focuses on summarizing the differences between comparable document groups.

In this paper, we propose a new principled, and versatile framework for MULTI-DOCUMENT SUMMARIZATION using the SUBMODULAR FUNCTION (MSSF). Many known

summarization tasks described above, including generic, query-focused, update, and comparative summarization, can be modeled as different variations derived from the proposed framework. The framework provides an elegant basis to establish connections between various summarization tasks while highlighting their differences.

In our summarization framework, the multi-document summarization problem is first mapped to the *budgeted maximum coverage problem* [11] which often arises in circuit layout, job scheduling, facility location, and other areas. Then the submodularity underlying the *term coverage* and the *textual-unit similarity* is taken into consideration for the greedy summarization algorithm, and is shown to have the ability of addressing the multi-document summarization problem. We further take advantage of the submodularity to modify the general greedy algorithm and finally adopt this modified version to improve the efficiency of our framework for different multi-document summarization tasks. Our work is closely related to Lin et al. [15], but different from their work which mainly resolves the generic summarization problem using the textual-unit (e.g., sentence) similarity, our work demonstrates advantages from three aspects:

1. proposes a new principled and versatile framework to address different summarization problems;
2. utilizes the improved greedy algorithm proposed by Minoux, [17] which provides higher efficiency than the general one as the backbone of the computation;
3. considers the term-coverage-based submodular function which shows the performance improvement over the textual-unit-similarity based one.

A preliminary study of the work has appeared as a 2-page poster at 2011 ACM SIGIR International Conference [13]. In this journal submission, we have added more analysis and discussion of the proposed framework, included more details of the improved greedy algorithm, and provided comprehensive experimental results. The rest of the paper is organized as follows. In Sect. 2, we review the related work about different multi-document summarization tasks and the submodular function. After introducing the original greedy algorithm and the modified version in Sect. 3, we propose the submodular function based framework for multi-document summarization, and model four aforementioned summarization tasks including generic, query-focused, update, and comparative summarization in Sect. 4. Section 5 presents experimental results of our framework on well-accepted summarization data sets. Finally Sect. 6 concludes the paper.

2 Related work

Generic summarization For generic summarization, a saliency score is usually assigned to each sentence, and then sentences are ranked according to the saliency score. Scores are usually calculated based on a combination of statistical, and linguistic features. MEAD [21], a well-known toolkit for document summarization, is an implementation of the centroid-based method in which sentence scores are computed based on sentence-level and inter-sentence features. In addition, there are some other approaches, including the probabilistic model [23], non-negative matrix factorization based model [27] and graph-based model [6, 26].

Lin et al. [15] propose attacking the generic multi-document summarization problem via submodular function. Our work shares the similar idea with theirs. However, their method only uses textual-unit-similarity (e.g., sentence-similarity) based submodular function, while ours also considers term-coverage based submodular functions which are more convincing under specific scenarios. Moreover, we also propose a principled and versatile framework which shows the capability to deal with many other summarization tasks besides the generic one. Last but not least, our method is more efficient due to the improved greedy algorithm.

Query-focused summarization In query-focused summarization, the information of the given topic or query should be incorporated into summarizers and sentences suiting the user's declared information need should be extracted. Many methods for the generic summarization can be extended to incorporate the query information [22, 29]. Wan et al. [25] make full use of both the relationship among all the sentences in the documents and the relationship between the given query and the sentences by manifold ranking. Probability models have also been proposed with different assumptions on the generation process of documents and queries [4, 9, 24].

Update and comparative summarization Update summarization was introduced in Document Understanding Conference (DUC) 2007 [2] and was a main task of the summarization track in Text Analysis Conference (TAC) 2008 [3]. It is required to summarize a set of documents under the assumption that the reader has already read and summarized the first set of documents as the main summary. To produce the update summary, some strategies are required to avoid the redundant information which has already been covered by the main summary. One of the most frequently used methods for removing the redundancy is Maximal Marginal Relevance (MMR) [8]. Comparative document summarization is proposed by Wang et al. [28] to summarize differences between comparable document groups. A sentence selection approach is proposed in [28] to accurately discriminate the documents in different groups modeled by the conditional entropy.

Submodularity In general, Submodularity is a diminishing returns policy, it shows that adding an element to a smaller set contributes more than adding it to a larger set, and is naturally to be used for efficiently finding out the optimal solution (in our case, the summarization). The formal definition of Submodularity is given as follows.

Let E be a finite set and f be a real valued nondecreasing function defined on the subsets of E that satisfies

$$f(S) + f(T) \leq f(S \cup T) + f(S \cap T), \tag{1}$$

where S and T are both subsets of E, such a function f is called **submodular** function [20]. A key observation is that submodular functions are closed under nonnegative linear combinations [12].

Particularly, several works contribute to maximizing the submodular function. For example, [7, 19] attacked the general unit cost submodular function maximization problem. They showed that for the monotonic increasing submodular function the greedy algorithm could achieve an approximation factor $(1 - 1/e)$. Khuller et al. [11] presented an algorithm that achieves an approximation factor $(1 - 1/e)$ for the budgeted submodular function maximization problem.

3 Algorithm using submodular function

3.1 Why submodularity?

The connection between the submodularity and the multi-document summarization cannot be easily identified. To clarify this, an alternative property of submodularity named as *decreasing marginal value* is given by:

$$f(T \cup \{\varsigma\}) - f(T) \leq f(S \cup \{\varsigma\}) - f(S), \tag{2}$$

where $S \subseteq T$, S and T are two subsets of E, and $\varsigma \in E \setminus T$. Intuitively, through this property, by adding one element to a larger set T, the value increment of f can never be larger than that by adding one element to a smaller set S. This intuitive diminishing property exists in different areas, e.g., in the social network, adding one new friend cannot increase more social influence for a more social group than for a less social group [12].

The *budgeted maximum coverage problem* is then described as: given a set of elements E where each element is associated with an influence and a cost defined over a domain of these elements and a budget B, the goal is to find out a subset of E which has the largest possible influence while the total cost does not exceed B. This problem is NP-hard [11]. However, [11] proposed a greedy algorithm which picks up the element that increases the largest possible influence within the cost limit each time and it guarantees the influence of the result subset is $(1 - 1/e)$-approximation. Submodularity resides in each "pick up" step.

Based on the submodular function and the budgeted maximum coverage problem, we can derive the answer to the question: why do we use submodularity for the multi-document summarization task? Let us delve into multi-document summarization task from two directions: the first one is the *term coverage*, and the second one is the *textual-unit similarity*.

3.1.1 Term coverage

A pool of sentences is formed for the given document set. The problem is how to pick up the most representative sentences from that pool as the summary of this document set[1] within the budget. Suppose the budget is the number of terms, the action of adding one candidate sentence is associated with its *summarization quality increase* (i.e., the overall quality increase incurred by the terms in this candidate sentence while not in the already picked sentences.) and *cost*. (i.e., the number of terms in this candidate sentence.) The quality of the current generated summary S over the document set is hereby defined as

$$f(S) = \#(\{t | t \text{ is term of } S\}), \tag{3}$$

which denotes the cardinality of the term set of S. Accordingly, the quality increase incurred by adding a candidate sentence can be defined by

$$I(\varsigma) = \#(\{t_1 | t_1 \text{ is term of } \varsigma\} \setminus \{t_2 | t_2 \text{ is term of } S\}), \tag{4}$$

where ς is the candidate sentence.

It does make sense that f holds the submodular property since the quality increase given by a candidate sentence based on a larger set of already picked sentences is smaller than that based on a smaller set. One common practice in defining f is to assign the weight (we treat the term frequency as the weight in this paper) to each term in the document set. Then the definition of $f(S)$ is given by

$$f(S) = \sum_{t \in S} w_t, \tag{5}$$

where w_t is the weight of term t.

Accordingly, the definition of the quality increase incurred by adding a new sentence ς to the current generated summary S is

$$I(\varsigma) = \sum_{t \in \varsigma \setminus S} w_t, \tag{6}$$

Intuitively, the candidate sentence which provides more quality increase should be picked as the new sentence to form the summary and the length of the final summary is fixed. Hence, we can treat multi-document summarization as a budgeted maximum coverage problem.

[1] Here, the number of the extracted sentences or the number of words inside these sentences is fixed. We treat it as the budget B.

3.1.2 Textual-unit similarity

If the budget B is the number of terms in the summary, the cost of each candidate sentence is the number of terms within it. A high quality summary should be two-fold: (1) maximizes the information coverage of the given document set; (2) minimizes the redundancy. One of the most popular methods for serving these two purposes is Maximal Marginal Relevance (MMR) [8] which aims to reduce the redundancy and maintain query relevance in retrieved documents at the same time. Hence, a MMR-similar definition for the quality of the current generated summary is given by

$$f(S) = \sum_{s_i \in E \setminus S} \sum_{s_j \in S} \mathrm{sim}(s_i, s_j) - \sum_{s_i, s_j \in S, s_i \neq s_j} \mathrm{sim}(s_i, s_j), \quad (7)$$

where E is the whole sentence set, $\mathrm{sim}(s_i, s_j)$ is the weight between the textual units s_i and s_j (the typical textual unit is sentence). Note that the first component of (7) is for the information coverage and the second component is for the redundancy removal, these two terms carry the same weight. Both information coverage and redundancy terms of $f(S)$ are submodular, thus $f(S)$ is also submodular, since the linear combination of submodular functions is closed. Suppose ς is the candidate sentence, the quality increase is therefore represented as follows:

$$I(\varsigma) = f(S \cup \{\varsigma\}) - f(S). \quad (8)$$

The goal is to generate a summary which provides the largest possible quality within the budget. Hence, the mapping from the multi-document summarization problem to the budgeted maximum coverage problem is straightforward.

According to the above analysis, the multi-document summarization problem can be modeled as a budgeted maximum coverage problem in two different levels—the term coverage and the textual-unit similarity. The general greedy algorithm for the multi-document summarization is presented in Sect. 3.2.

3.2 Algorithm for summarization

The main idea of the greedy algorithm for the multi-document summarization problem is simple: sequentially pick up the sentence which provides the largest quality increase based on the sentences in the current summary until the budget is reached.

As we discussed in Sect. 3.1, there are two ways of defining the specific submodular function for the summarization. The first one is from the term coverage perspective, and the second one is from the textual-unit similarity perspective. Given a document set D, a budget B and the indication of two submodular function types, the greedy algorithm utilizes the appropriate submodular function to generate a summary for D within B. The procedure is shown in Algorithm 1.

Algorithm 1 The greedy algorithm for summarization

Input DocumentSet D, budget B,
SubmodularFunctionType T
if $T =$ "*Term Coverage*" **then**
Summary = Summ-TermCoverage(D, B)
end if
if $T =$ "*Textual-Unit Similarity*" **then**
Summary = Summ-UnitSimilarity(D, B)
end if
Output Summary

3.2.1 Algorithm details

The core components in Algorithm 1 are "Summ-TermCoverage", and "Summ-UnitSimilarity". Most details of these two core components are identical in Algorithm 2 except the definition of the submodular function f as well as the quality increase incurred by adding a new sentence.

Algorithm 2 The core component of the greedy algorithm for summarization

Extract sentence set E from document set D
Initial remaining sentence set R as E
Initial summary S as $\emptyset$
Initial cost C as 0

while Size(R)>0 **do**
$\varsigma \leftarrow$ The sentence which has $\arg\max_{e \in R} \frac{I(e)}{(length(e))^p}$
if $(C \leftarrow C + length(\varsigma)) < B$
$S \leftarrow S \cup \{\varsigma\}$
$R \leftarrow R \setminus \{\varsigma\}$
else
Stop
end if
end while
Return S

The definition of f for "Summ-TermCoverage" is given in (5). In this context, whenever the algorithm picks up a new sentence into the summary, it would somehow strive to choose the longer sentence in the remaining sentence set, since the longer one has more possibility to cover more important terms and provide more quality increase. To avoid the summary containing only long sentences of the document set, we include the length of sentence as denominator

Table 1 A quick summary of the submodular functions for different summarization tasks

Summarization Type	Submodular Function
Generic Summarization	$f(S) = \sum_{t \in S} w_t$
	$f(S) = \sum_{s_i \in D \setminus S} \sum_{s_j \in S} \text{sim}(s_i, s_j) - \sum_{s_i, s_j \in S, s_i \neq s_j} \text{sim}(s_i, s_j)$
Query-focused Summarization	$f(S, q) = f_G + \sum_{s_i \in S} \text{sim}(q, s_i)$
Update Summarization	$f(q, S_1, S_2) = f_G + \sum_{s_i \in S_2} \text{sim}(q, s_i) - \sum_{s_i \in S_2} \sum_{s_j \in S_1} \text{sim}(s_i, s_j)$
Comparative Summarization	$f(S) = f_G - \sum_{s_i \in S} \sum_{s_j \in OtherGroups} \text{sim}(s_i, s_j)$

of the quality increase to weaken such effect. f for "SummUnitSimilarity" is the one defined in (7). Practically, we treat each sentence as the basic textual unit, and represent sentences as term vectors, each entry of which is the weight of term frequency-inverse sentence frequency (TF-ISF) [10]. Then the weight between two sentences is the pairwise cosine similarity. Similarly, we include the length of sentence as the denominator of the quality increase to avoid the bias.

Note that, in the case when the scaling factor of sentence length $p = 1$ and f is a normalized monotonic submodular function, it was proved by [11] that Algorithm 2 achieves a bounded approximation ratio $(1 - 1/e^{\frac{1}{2}})$; in other cases when the number of the sentences in the final summary S is $|S|, 0 \leq p < 1$ and f is a normalized monotonic submodular function, Lin et al. [15] proved that Algorithm 2 guarantees a bounded approximation ratio $(1 - \prod_{n=1}^{|S|}(1 - (c_n/B)^p))$. On one hand, (5) is a normalized monotonic submodular function, therefore, the above theoretical results holds; on the other hand, Lin et al. [15] proved that Algorithm 2 could still solve the summarization problem near-optimally with a high probability even though (7) is not guaranteed monotonic.

3.2.2 Improvements on algorithm

As we can see, each time the greedy algorithm picks up a new sentence, it has to recompute the quality increases considering each of the remaining sentences as the candidate based on the current summary. Suppose the given document set contains a huge number of sentences, the running time would be unacceptable. Hence, in order to apply this method to real world applications, we are wondering if the running time of this algorithm could be reduced.

Inspired by the work of Minoux [17], we further utilize the sumodularity to make modifications to the process of picking up new sentences. This general idea is: once the top sentence in the remaining sentence set R which holds the largest value of $\frac{I(e)}{(length(e))^p}$ based on the current summary, its following sentences can never surmount it, since as the summary enlarges, the value of $\frac{I(e)}{(length(e))^p}$ is getting smaller and smaller. In such case, there is no need to recompute $\frac{I(e)}{(length(e))^p}$ of all remaining sentences as in Algorithm 2, so that the running time is greatly reduced.

From Algorithm 3, one can find out the details of the changes.

Table 2 Notations

Notation	Meaning
D	Document Set
S	Summary
S_1	Summary for D_1
S_2	Summary for D_2
S'	Existing Summary
w_t	Weight of term t
s_i, s_j	Textual unit
sim	Similarity
q	Given query
f_G	General information coverage

4 The summarization framework

Our proposed submodularity-based framework can be modeled to different multi-document summarization tasks, including generic, query-focused, update and comparative summarization. In this section, we formulate each summarization task by defining different submodular functions.

For the generic summarization, we present the submodular function from the two aforementioned aspects: the term coverage and the textual-unit similarity. Table 1 summarizes the submodular functions for different summarization tasks, and Table 2 presents the notations. The general procedure of methods for different summarization tasks is described in Algorithm 1 and Algorithm 3, while the only difference resides in the submodular functions.

4.1 Generic summarization

Given a set of documents, the generic summarization is the task of extracting a set of sentences which can cover the gen-

Algorithm 3 The core component of the improved greedy algorithm

Extract sentence set E from document set D
Initial summary S as $\emptyset$
Initial remaining sentence set R as E
Assign $v_e = \frac{I(e)}{(length(e))^p}$ to each sentence e of R
Initial cost C as 0

while Size(R) > 0 **do**
  while true **do**
    $t = \text{Top}(R)$
    $R \leftarrow$ sort R based on v_e of each sentence e
    $t' = \text{Top}(R)$
    if $t \neq t'$
      Save $v'_t = \frac{I(t')}{(length(t'))^q}$ for t'
    else
      Stop the while
    end if
  end while
  if $(C \leftarrow C + length(t')) < B$
    $S \leftarrow S \cup \{t'\}$
    $R \leftarrow R \setminus \{t'\}$
  else
    Stop
  end if
  $t'' = \text{Top}(R)$
  Save $v''_t = \frac{I(t'')}{(length(t''))^q}$ for t''
end while
Return S

eral ideas of the document set. If there is no length limit to the summary, all the sentences in the whole document set would be the final summary since they cover all the content of the documents. However, such summary results in great difficulty of reading and capturing the general ideas for users; contrarily, it would be better to set a summary length for the summarization task. As discussed in Sect. 3.2, given the length limit to the summary, the generic summarization problem can be resolved by using the submodular function.

The submodular function for the generic summarization is defined as (5) for the term frequency or (7) for the textual-unit similarity. Notice that (5) considers all terms no matter how many times they appear in the document set. In reality, it is possible that the result will involve terms with lower frequency and with no remarkable contribution to the summary. Therefore, we set the threshold λ in the experiment to filter such terms. In other words, if the frequency of a term is less than λ, it will be discarded.

4.2 Query-focused summarization

The query-focused summarization is to generate a short summary based on a given document set and a given query. The generated summary reflects the condensed information related to the given query under the length budget. Different from the generic summarization that generates summaries presenting the general ideas of the document set, the query-focused summarization provides the summary that can satisfy special requirement for users.

Given a document set and a query q, we define the quality function as

$$f(S, q) = f_G + \sum_{s_i \in S} \text{sim}(q, s_i), \tag{9}$$

where the first term represents the general information coverage which could be replaced by (5) or (7), the second term represents the query-focused information coverage. Clearly, this function is a submodular function, since both parts in (9) are submodular, and the linear combination of submodular functions is closed.

4.3 Update summarization

The update Summarization is a form of the multi-document summarization in which we generate a summary of a new document set based on the assumption that the user has already read a given document set. Generally, this summarization task is based on the following scenario: A user is interested in a particular news topic and wants to track its related news as it evolves over time, so he/she subscribes to a news feed that sends his/her relevant articles as they are submitted from various news services. However, either there are so many news articles that he/she cannot keep up with, or he/she has to leave for a while and then wants to catch up. Whenever he/she checks out news of his/her interested topic, it bothers him/her that most articles keep repeating the same information; he/she would like to read summaries that only talk about what's new or different about this topic.[2] We formulate such problem as follows:

Given a query q (represents the user's interested topic) and two sets of documents D_1 (already read articles) and D_2 (new articles), the update summarization aims to generate a summary of D_2 related to the query q, given D_1. First of all, the summary of D_1, referred as to S_1, can be generated. Then, the update summary of D_2 related to q, referred as to S_2 is generated. The main idea of S_2 should be different from the main idea of S_1. Also, S_2 should cover all the aspects of the document set D_2 as many as possible.

[2] http://www.nist.gov/tac/2009/Summarization/update.summ.09.guidelines.html.

Based on this formal definition, we formulate the submodular function for the update summarization as

$$f(q, S_1, S_2) = f_G + \sum_{s_i \in S_2} \text{sim}(q, s_i) - \sum_{s_i \in S_2} \sum_{s_j \in S_1} \text{sim}(s_i, s_j), \quad (10)$$

where S_1 is the existing summary and S_2 is the updated summary. The first term in (10) denotes the general information coverage for new coming document set, the second term denotes the query-focused information coverage, and the last terms denotes the redundancy given S_1.

Since each term is a submodular function, the property of the submodularity holds for the linear combination of these terms. Similar to (7), (10) is not monotonic everywhere, but there is a high probability that a near-optimal solution can be generated.

4.4 Comparative summarization

Given a collection of document groups, the comparative summarization is to generate a short summary delivering the differences of these documents by extracting the most discriminative sentences in each document group. The traditional document summarization aims to cover the majority of the information among document collections, while the comparative summarization is to find differences. We formulate the comparative summarization as follows:

Given N groups of documents $G_1, G_2, \ldots, G_N$, the comparative summarization aims to generate summaries $S_1, S_2, \ldots, S_N$ such that the summaries can represent topics of corresponding groups whereas they are different from each other on the theme level.

We extend our greedy algorithm for the comparative summarization to generate the discriminant summary for each group of documents. The submodular function for the comparative summarization is defined as

$$f(S) = f_G - \sum_{s_i \in S} \sum_{s_j \in OtherGroups} \text{sim}(s_i, s_j), \quad (11)$$

where S is the summary of the current group. The first term represents the general information coverage of current group, while the second term represents the redundancy based on the other groups. Clearly, the linear combination of these terms holds the submodularity property. As before, without the monotonic property, (11) has a high probability to generate a near-optimal solution.

5 Experiments

We have conducted experiments on the four summarization tasks and our proposed method based on the submodular function has outperformed many existing approaches. For the generic summarization, the DUC04 data set is applied. For the query-focused summarization, the DUC05 and the DUC06 data sets are adopted as the experiment data. As for the update summarization task, the experiments are performed on the TAC08 data set. The brief description of the data sets can be found in Table 3. For the comparative summarization, we use the TDT2 corpora to compare the summary generated by different comparative summarization methods. Note that we treat the sentence as the basic textual unit for all experiments those need consider textual-unit similarity.

All the tasks, except the comparative summarization, are evaluated by Recall-Oriented Understudy for Gisting Evaluation (ROUGE)—an evaluation toolkit for document summarization [14] which automatically determines the quality of a summary by comparing it with the human generated summaries through counting the number of their overlapping textual units (e.g., n-gram, word sequences, and etc.). In particular, F-measure scores of ROUGE-2 and ROUGE-SU4 are presented for our experiment. For the comparative summarization, we provide two other approaches for the purpose of comparison. The detailed experimental results are described in the following.

5.1 Generic summarization

For the Generic summarization, we use DUC04 as the experimental data. We evaluate our method (denoted as MSSF) on the generic summarization from two aspects: the term coverage and the textual-unit similarity (in our experiment setup, the sentence is the basic textual-unit).

We observe through the experiment that the summary result generated by our method is the best when the threshold $\lambda = 2$. Consequently, we set λ as 2 when performing comparative experiments with other existing methods. we also

Table 3 Brief description of the data sets

	DUC04	DUC05	DUC06	TAC08 A	TAC08 B
Type of Summarization	Generic	Query-focused	Query-focused	Query-focused	Update
#topics	50	50	50	48	48
#documents per topic	10	25–50	25	10	10
Summary length	665 bytes	250 words	250 words	100 words	100 words

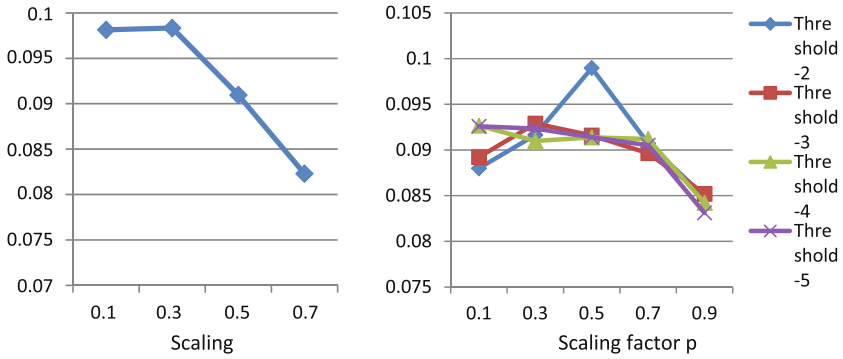

Fig. 1 *Left*: ROUGE-2 for MSSF (Sentence Similarity) using scaling factor 0.1–0.7; *Right*: ROUGE-2 on threshold ranging from 2–5 for MSSF (Term Coverage) using scaling factor 0.1–0.9

Table 4 Results on generic summarization

	ROUGE-2	ROUGE-SU4
DUC Best	0.09216	0.13233
Random	0.06377	0.11779
MMR	0.09144	0.13287
LexPageRank	0.08572	0.13097
Centroid	0.07379	0.12511
LSA	0.06538	0.11946
NMF	0.07261	0.12918
MSSF(Term Coverage)	*0.09897*	*0.13951*
MSSF(Textual-Unit Similarity)	*0.09834*	*0.13901*

conduct experiments to evaluate the sensitivity of the scaling factor p on MSSF (Textual-Unit Similarity) and on MSSF (Term Coverage) using different thresholds. From Fig. 1, we have two observations: (1) Different scaling factors do have different impacts on the result under consistent experiment setting (here, consistent setting could mean MSSF (Textual-Unit Similarity) or the same threshold on MSSF (Term Coverage)); (2) Under different experiment settings, the best results are not always given by the same scaling factor, i.e., when performing MSSF (Term Coverage) using the threshold 2, the best summarization is given by the scaling factor 0.5, while performing MSSF (Term Coverage) using the threshold 3, the scaling factor 0.3 gives the best result.

After clarifying the impact of the scaling factor, we set it as 0.5 for MSSF (Term Coverage) and 0.3 (It shows the best result in Fig. 1 when p is set as 0.3) for MSSF (Sentence Similarity). We implement the following widely used or recent published methods for generic summarization as the baseline systems to compare with our proposed method MSSF: (1) Random: the method randomly selects specific sentences as the summary; (2) Maximum Marginal Relevance (MMR): the method is similar with MSSF (Textual-Unit Similarity) as we mentioned in Sect. 3.1.2. It greedily selects the sentence which maximizes the relevance with the given document set while minimizes the redundancy with the sentences that have already been selected; (3) LexPageRank: the method first constructs a sentence connectivity graph based on the cosine similarity and then selects important sentences based on the concept of eigenvector centrality [6]; (4) Centroid: the method extracts sentences based on the centroid value, the positional value and the first sentence overlap; (5) Latent Semantic Analysis (LSA): the method identifies semantically important sentences by conducting latent semantic analysis; (6) Non-negative Matrix Factorization (NMF): the method performs NMF on the sentence-term matrix and select the high ranked sentences.

From the results showed in Table 4, our method MSSF clearly outperforms the other rivals and is even better than the DUC04 best team work. Note that MSSF (Term Coverage) is slightly better than MSSF (Textual-Unit Similarity) which has the similar submodular function as the work of Lin et al. [15]. Since one sentence of the given document set should be covered by at least one sentence in the summary, not by all summary sentences, sometimes there may exist bias in the first term of submodular function (7). In a word, MSSF (Term Coverage) is more reasonable.

5.2 Query-focused summarization

Main tasks of DUC05 and DUC06 are both the query-focused summarization, and therefore we conduct experiments on these two data sets. In addition to baseline systems, we also compared our system with some widely used and recently published systems: (1) SNMF [27]: calculates sentence-sentence similarities by the sentence level semantic analysis, clusters the sentences via the symmetric non-negative matrix factorization, and extracts the sentences based on the clustering result; (2) Qs-MRF [29]: extends the mutual reinforcement principle between the sentence and the term to the document-sentence-term mutual reinforcement chain, and uses the query-sensitive similarity to measure the affinity between the pair of texts; (3) Wiki [18]: uses Wikipedia as the external knowledge to expand the query and builds the connection between the query and the sentences in documents.

The empirical result are reported in Table 5. The results show that on DUC05, our method outperforms the other systems except Qs-MRF and Wiki; on DUC06, our method

Table 5 Results on query-focused summarization

	DUC05		DUC06	
	ROUGE-2	ROUGE-SU4	ROUGE-2	ROUGE-SU4
Average-Human	0.10236	0.16221	0.11249	0.1706
DUC Average	0.06024	0.11488	0.07543	0.13206
Random	0.04143	0.09066	0.04892	0.10083
LSA	0.04079	0.09352	0.05022	0.10226
SNMF	0.06043	0.12298	0.08549	0.13981
Qs-MRF	0.0779	0.1366	0.08917	0.14329
Wiki	0.07074	0.13002	0.08091	0.14022
MSSF	0.0731	0.12718	0.09193	0.14611

Table 6 Results on update summarization

	TAC08 A		TAC08 B	
	ROUGE-2	ROUGE-SU4	ROUGE-2	ROUGE-SU4
TAC Best	0.1114	0.14298	0.10108	0.13669
TAC Median	0.08123	0.11975	0.06927	0.11046
MSSF	0.08327	0.12109	0.09451	0.13180

achieves the best result. This is due to the novel adoption of the submodular function. Note that our method is simpler than the other systems because of the greedy heuristic.

5.3 Update summarization

Note that the data set for the update summarization, (i.e. the main task of TAC08 summarization track), is composed of 48 topics and 20 news wire articles for each topic. The 20 articles are grouped into two groups. The update summarization task requires to produce two summaries, involving the initial summary (TAC08 A), which is the standard query-focused summarization, and the update summary (TAC08 B) under the assumption that the reader has already read the first 10 documents.

Table 6 shows the comparative experimental results on the update summarization. In Table 6, "TAC Best" and "TAC Median" represent the best and median results from the participants of the TAC08 summarization track in the two tasks respectively according to the TAC08 report [3]. As seen from the results, the ROUGE scores of our methods are higher than the median results. The good results of the best team typically come from the fact that they utilize advanced natural language processing (NLP) techniques to resolve pronouns and other anaphoric expressions. Although we can spend more efforts on the preprocessing or the language processing step, our goal here is to demonstrate the effectiveness of formalizing the update summarization problem using the submodular function and hence we do not utilize advanced NLP techniques for preprocessing. Experimental results demonstrate that our simple update summarization method based on the submodular function can lead to the competitive performance for the update summarization.

Table 7 TDT2 corpora topic description

Topic	Description
1	Iraq Issues
2	Asia's economic crisis
3	Lewinsky scandal
4	Nagano Olympic Games
5	Nuclear Issues in Indian and Pakistan
6	Jakarta Riot

5.4 Comparative summarization

For the comparative summarization, we use the top six largest clusters of documents from the TDT2 corpora to compare the summary generated by different comparative summarization methods. The topics of the six document clusters are described as in Table 7.

From each of the topics, 30 documents are extracted randomly to produce a one-sentence summary. For the comparison purpose, we select the sentence that is the most similar to other sentences in the document group as the baseline, denoted as "MS". We also implement the methods proposed by [28]. Table 8 shows the summaries generated by MS, the discriminative sentence selection (DSS) [28] and our method MSSF. As we can see, DSS can extract discriminative sentences for all the topics except topic 4 and topic 6. Note that the sentence extracted by DSS for topic 4 may be discriminative from other topics, but it is deviated from

Table 8 A case study on comparative document summarization. Some unimportant words are skipped due to the space limit. The bold font is used to annotate the phrases that are highly related with the topics, and italic font is used to highlight the sentences that are not proper to be used in the summary

Topic	MS	DSS	MSSF
1	... U.S. Secretary of State Madeleine Albright arrives to consult on the stand-off between the **United Nations and Iraq**	**the U.S. envoy to the United Nations**, Bill Richardson, ... play down China's refusal to support threats of **military force against Iraq**	The arrival of U.S. Secretary of State Madeleine Albright could be an early test of **the accord Iraq signed ten days ago with U.N. secretary-general Kofi Annan**
2	**Thailand's currency**, the baht, **dropped through a key psychological level** of ... amid a regional sell-off sparked by escalating **social unrest in Indonesia**	Earlier, driven largely by **the declining yen**, **South Korea's stock market fell** by ..., while the **Nikkei 225 benchmark index dipped** below 15,000 in the morning ...	Prueher addressed the army seminar in Manila and told delegates that **Asia's financial troubles** have affected **the United States' joint military activities with its Asian allies**
3	... attorneys representing **President Clinton and Monica Lewinsky**	The following night **Isikoff** ..., where he directly followed the recitation of the top-10 list: "Top 10 **White House Jobs That Sound Dirty**"	In Washington, Ken Starr's grand jury continued its investigation of the **Monica Lewinsky matter**
4	Eight women and six men were named Saturday night as the first **U.S. Olympic Snowboard Team** as their sport gets set to make its debut in **Nagano, Japan**	*this tunnel is Finland's cross country version of Tokyo's alpine ski dome, and Olympic skiers flock from Russia, ..., France and Austria this past summer to work out the kinks ...*	Seizinger, **the German all-round skier** who has been **Street's fiercest rival** in recent years, did win **the downhill title**
5	**U.S. officials** have announced **sanctions** Washington will impose on **India and Pakistan** for conducting **nuclear tests**	The **sanctions** would stop all foreign aid except for humanitarian purposes, **ban military sales to India** ...	**Weapons experts** say **Pakistan** has long thought to have had all the components necessary to **build a nuclear device**
6	... remain in force around **Jakarta**, and at the Parliament building where **thousands of students staged a sit-in** Tuesday ...	President Suharto has given much to his country over the past 30 years, raising Indonesia's standing in the world ...	... to **press their demand** for what they feel should be true **political reform**, that is, **election of a totally new government** ...

the topic Nagano Olympic Games. The MS method can extract general sentences related to the corresponding topics to some extent. However, some sentences extracted by MS only contains the keywords of the related topics, but not the essence of the topic (i.e., the summary of topic 3). Comparatively, our MSSF method can extract discriminative sentences for all topics with the essential idea. For example, the summary of topic 6 clearly explains the reason why the Jakarta Riot happened.

5.5 Improved algorithm

To evaluate the efficiency of the improved greedy algorithm used in our summarization framework, we compare the general greedy algorithm with this new algorithm for generic, query-focused and update summarization tasks on DUC04, DUC05 and TAC08 accordingly. Notice that the summaries generated by the general greedy algorithm and the improved one are the same. For each summarization task, we perform each of the two algorithms for ten times, and compute the average running time for each of them. The comparison results are shown in Fig. 2. We observed that the improved algorithm is shown to be more efficient on all the tasks. The running time comparison demonstrates the efficiency of our proposed summarization framework.

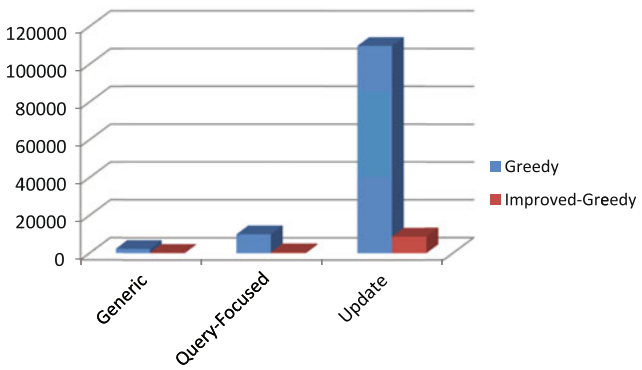

Fig. 2 Average running time (in milliseconds) of two algorithms on three summarization tasks

6 Conclusion

In this paper, we present a new principled and versatile summarization framework—MSSF for MULTI-DOCUMENT SUMMARIZATION using the SUBMODULAR FUNCTION.

This framework can deal with different summarization tasks, including generic, query-focused, updated, comparative summarization. The empirical results show that this framework outperforms the other rivals in the generic summarization and is competitive in other summarization tasks. The ability to address these summarization problems benefits from various submodular functions for corresponding summarization tasks. Our proposed framework is shown to be more efficient because of the proposed improved summarization algorithm.

Acknowledgements The work is partially supported by National Science Foundation (NSF) under grants IIS-0546280 and CCF-0939179, and by Department of Homeland Security (DHS) under grant 2010-ST-062-000039.

References

1. Chen CM, Liu CY (2009) Personalized e-news monitoring agent system for tracking user-interested Chinese news events. Appl Intell 30(2):121–141
2. Dang HT (2007) Overview of DUC 2007. In: Document understanding conference, pp 1–10
3. Dang HT, Owczarzak K (2008) Overview of the TAC 2008 update summarization task. In: Proceedings of text analysis conference
4. Daumé H, Marcu D (2006) Bayesian query-focused summarization. In: Annual meeting—Association for Computational Linguistics, vol 44, p 305
5. Dimililer N, Varoğlu E, Altınçay H (2009) Classifier subset selection for biomedical named entity recognition. Appl Intell 31(3):267–282
6. Erkan G, Radev DR (2004) Lexpagerank: Prestige in multi-document text summarization. In: Proceedings of EMNLP, vol 4
7. Gérard C et al (1984) Submodular set functions, matroids and the greedy algorithm: tight worst-case bounds and some generalizations of the Rado-Edmonds theorem. Discrete Appl Math 7(3):251–274
8. Goldstein J, Mittal V, Carbonell J, Kantrowitz M (2000) Multi-document summarization by sentence extraction. In: NAACL-ANLP 2000 workshop on automatic summarization. Association for Computational Linguistics, Stroudsburg, pp 40–48
9. Haghighi A, Vanderwende L (2009) Exploring content models for multi-document summarization. In: Proceedings of human language technologies: The 2009 annual conference of the North American Chapter of the Association for Computational Linguistics on ZZZ. Association for Computational Linguistics, Stroudsburg, pp 362–370
10. Jurafsky D, Martin JH, Kehler A, Vander Linden K, Ward N (2000) Speech and language processing. Prentice Hall, New York
11. Khuller S, Moss A, Naor JS (1999) The budgeted maximum coverage problem. Inf Process Lett 70(1):39–45
12. Leskovec J, Krause A, Guestrin C, Faloutsos C, VanBriesen J, Glance N (2007) Cost-effective outbreak detection in networks. In: Proceedings of the 13th ACM SIGKDD international conference on knowledge discovery and data mining. ACM, New York, p 429
13. Li J, Li L, Li T (2011) MSSF: A multi-document summarization framework based on submodularity. In: Proceedings of SIGIR'11
14. Lin CY (2004) Rouge: A package for automatic evaluation of summaries. In: Proceedings of the workshop on text summarization branches out (WAS 2004), pp 25–26
15. Lin H, Bilmes J (2010) Multi-document summarization via budgeted maximization of submodular functions. In: NAACL/HLT
16. Mani I (2001) Automatic summarization. Comput Linguist 28(2)
17. Minoux M (1978) Accelerated greedy algorithms for maximizing submodular set functions. Optim Tech 234–243
18. Nastase V (2008) Topic-driven multi-document summarization with encyclopedic knowledge and spreading activation. In: Proceedings of the conference on empirical methods in natural language processing. Association for Computational Linguistics, Stroudsburg, pp 763–772
19. Nemhauser GL, Wolsey LA (1981) Maximizing submodular set functions: formulations and analysis of algorithms. Stud Graphs Discrete Program 11:279–301
20. Nemhauser GL, Wolsey LA, Fisher ML (1978) An analysis of approximations for maximizing submodular set functions. Math Program 14(1):265–294
21. Radev DR, Jing H, Sty M, Tam D (2004) Centroid-based summarization of multiple documents. Inf Process Manag 40(6):919–938
22. Saggion H, Bontcheva K, Cunningham H (2003) Robust generic and query-based summarisation. In: Proceedings of the European chapter of computational linguistics (EACL). Research notes and demos
23. Steinberger J, Jezek K (2004) Using latent semantic analysis in text summarization and summary evaluation. In: Proc. ISIM04, pp 93–100
24. Tang J, Yao L, Chen D (2009) Multi-topic based query-oriented summarization. In: Proceedings of SDM
25. Wan X, Yang J, Xiao J (2007) Manifold-ranking based topic-focused multi-document summarization. In: Proceedings of IJCAI, pp 2903–2908
26. Wan X, Yang J, Xiao J (2007) Towards an iterative reinforcement approach for simultaneous document summarization and keyword extraction. In: Annual meeting—Association for Computational Linguistics, vol 45, p 552
27. 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. ACM, New York, pp 307–314
28. Wang D, Zhu S, Li T, Gong Y (2009) Comparative document summarization via discriminative sentence selection. In: Proceeding of the 18th ACM conference on information and knowledge management. ACM, New York, pp 1963–1966
29. Wei F, Li W, Lu Q, He Y (2008) Query-sensitive mutual reinforcement chain and its application in query-oriented multi-document summarization. In: Proceedings of the 31st annual international ACM SIGIR conference on research and development in information retrieval. ACM, New York, pp 283–290