A Class of Submodular Functions for Document Summarization

Hui Lin
Dept. of Electrical Engineering
University of Washington
Seattle, WA 98195, USA
hlin@ee.washington.edu

Jeff Bilmes
Dept. of Electrical Engineering
University of Washington
Seattle, WA 98195, USA
bilmes@ee.washington.edu

Abstract

We design a class of submodular functions meant for document summarization tasks. These functions each combine two terms, one which encourages the summary to be representative of the corpus, and the other which positively rewards diversity. Critically, our functions are monotone nondecreasing and submodular, which means that an efficient scalable greedy optimization scheme has a constant factor guarantee of optimality. When evaluated on DUC 2004-2007 corpora, we obtain better than existing state-of-art results in both generic and query-focused document summarization. Lastly, we show that several well-established methods for document summarization correspond, in fact, to submodular function optimization, adding further evidence that submodular functions are a natural fit for document summarization.

1 Introduction

In this paper, we address the problem of generic and query-based extractive summarization from collections of related documents, a task commonly known as *multi-document summarization*. We treat this task as monotone submodular function maximization (to be defined in Section 2). This has a number of critical benefits. On the one hand, there exists a simple greedy algorithm for monotone submodular function maximization where the summary solution obtained (say $\hat{S}$) is guaranteed to be almost as good as the best possible solution (say S_{opt}) according to an objective $\mathcal{F}$. More precisely, the greedy algorithm is a constant factor approximation to the cardinality constrained version of the problem, so that $\mathcal{F}(\hat{S}) \geq (1 - 1/e)\mathcal{F}(S_{\text{opt}}) \approx 0.632\mathcal{F}(S_{\text{opt}})$. This is particularly attractive since the quality of the solution does not depend on the size of the problem, so even very large size problems do well. It is also important to note that this is a worst case bound, and in most cases the quality of the solution obtained will be much better than this bound suggests.

Of course, none of this is useful if the objective function $\mathcal{F}$ is inappropriate for the summarization task. In this paper, we argue that monotone nondecreasing submodular functions $\mathcal{F}$ are an ideal class of functions to investigate for document summarization. We show, in fact, that many well-established methods for summarization (Carbonell and Goldstein, 1998; Filatova and Hatzivassiloglou, 2004; Takamura and Okumura, 2009; Riedhammer et al., 2010; Shen and Li, 2010) correspond to submodular function optimization, a property not explicitly mentioned in these publications. We take this fact, however, as testament to the value of submodular functions for summarization: if summarization algorithms are repeatedly developed that, by chance, happen to be an instance of a submodular function optimization, this suggests that submodular functions are a natural fit. On the other hand, other authors have started realizing explicitly the value of submodular functions for summarization (Lin and Bilmes, 2010; Qazvinian et al., 2010).

Submodular functions share many properties in common with convex functions, one of which is that they are closed under a number of common combination operations (summation, certain compositions, restrictions, and so on). These operations give us the tools necessary to design a powerful submodular objective for submodular document summarization that extends beyond any previous work. We demonstrate this by carefully crafting a class of submodular func-

tions we feel are ideal for extractive summarization tasks, both generic and query-focused. In doing so, we demonstrate better than existing state-of-the-art performance on a number of standard summarization evaluation tasks, namely DUC-04 through to DUC-07. We believe our work, moreover, might act as a springboard for researchers in summarization to consider the problem of "how to design a submodular function" for the summarization task.

In Section 2, we provide a brief background on submodular functions and their optimization. Section 3 describes how the task of extractive summarization can be viewed as a problem of submodular function maximization. We also in this section show that many standard methods for summarization are, in fact, already performing submodular function optimization. In Section 4, we present our own submodular functions. Section 5 presents results on both generic and query-focused summarization tasks, showing as far as we know the best known ROUGE results for DUC-04 through DUC-06, and the best known precision results for DUC-07, and the best recall DUC-07 results among those that do not use a web search engine. Section 6 discusses implications for future work.

2 Background on Submodularity

We are given a set of objects $V = \{v_1, \ldots, v_n\}$ and a function $\mathcal{F} : 2^V \rightarrow \mathbb{R}$ that returns a real value for any subset $S \subseteq V$. We are interested in finding the subset of bounded size $|S| \leq k$ that maximizes the function, e.g., $\operatorname{argmax}_{S \subseteq V} \mathcal{F}(S)$. In general, this operation is hopelessly intractable, an unfortunate fact since the optimization coincides with many important applications. For example, $\mathcal{F}$ might correspond to the value or coverage of a set of sensor locations in an environment, and the goal is to find the best locations for a fixed number of sensors (Krause et al., 2008).

If the function $\mathcal{F}$ is monotone submodular then the maximization is still NP complete, but it was shown in (Nemhauser et al., 1978) that a greedy algorithm finds an approximate solution guaranteed to be within $\frac{e-1}{e} \sim 0.63$ of the optimal solution, as mentioned in Section 1. A version of this algorithm (Minoux, 1978), moreover, scales to very large data sets. Submodular functions are those that satisfy the property of *diminishing returns*: for any $A \subseteq B \subseteq V \backslash v$, a *submodular* function $\mathcal{F}$ must satisfy $\mathcal{F}(A+v) - \mathcal{F}(A) \geq \mathcal{F}(B+v) - \mathcal{F}(B)$. That is, the incremental "value" of v decreases as the context in which v is considered grows from A to B. An equivalent definition, useful mathematically, is that for any $A, B \subseteq V$, we must have that $\mathcal{F}(A) + \mathcal{F}(B) \geq \mathcal{F}(A \cup B) + \mathcal{F}(A \cap B)$. If this is satisfied everywhere with equality, then the function $\mathcal{F}$ is called *modular*, and in such case $\mathcal{F}(A) = c + \sum_{a \in A} \vec{f}_a$ for a sized $|V|$ vector $\vec{f}$ of real values and constant c. A set function $\mathcal{F}$ is *monotone nondecreasing* if $\forall A \subseteq B$, $\mathcal{F}(A) \leq \mathcal{F}(B)$. As shorthand, in this paper, monotone nondecreasing submodular functions will simply be referred to as *monotone submodular*.

Historically, submodular functions have their roots in economics, game theory, combinatorial optimization, and operations research. More recently, submodular functions have started receiving attention in the machine learning and computer vision community (Kempe et al., 2003; Narasimhan and Bilmes, 2005; Krause and Guestrin, 2005; Narasimhan and Bilmes, 2007; Krause et al., 2008; Kolmogorov and Zabin, 2004) and have recently been introduced to natural language processing for the tasks of document summarization (Lin and Bilmes, 2010) and word alignment (Lin and Bilmes, 2011).

Submodular functions share a number of properties in common with convex and concave functions (Lovász, 1983), including their wide applicability, their generality, their multiple options for their representation, and their closure under a number of common operators (including mixtures, truncation, complementation, and certain convolutions). For example, if a collection of functions $\{\mathcal{F}_i\}_i$ is submodular, then so is their weighted sum $\mathcal{F} = \sum_i \alpha_i \mathcal{F}_i$ where α_i are nonnegative weights. It is not hard to show that submodular functions also have the following composition property with concave functions:

Theorem 1. *Given functions $\mathcal{F} : 2^V \rightarrow \mathbb{R}$ and $f : \mathbb{R} \rightarrow \mathbb{R}$, the composition $\mathcal{F}' = f \circ \mathcal{F} : 2^V \rightarrow \mathbb{R}$ (i.e., $\mathcal{F}'(S) = f(\mathcal{F}(S))$) is nondecreasing submodular, if f is non-decreasing concave and $\mathcal{F}$ is nondecreasing submodular.*

This property will be quite useful when defining submodular functions for document summarization.

3 Submodularity in Summarization

3.1 Summarization with knapsack constraint

Let the *ground set* V represents all the sentences (or other linguistic units) in a document (or document collection, in the multi-document summarization case). The task of extractive document summarization is to select a subset $S \subseteq V$ to represent the entirety (ground set V). There are typically constraints on S, however. Obviously, we should have $|S| < |V| = N$ as it is a summary and should be small. In standard summarization tasks (e.g., DUC evaluations), the summary is usually required to be length-limited. Therefore, constraints on S can naturally be modeled as *knapsack constraints*: $\sum_{i \in S} c_i \leq b$, where c_i is the non-negative cost of selecting unit i (e.g., the number of words in the sentence) and b is our *budget*. If we use a set function $\mathcal{F} : 2^V \rightarrow \mathbb{R}$ to measure the quality of the summary set S, the summarization problem can then be formalized as the following combinatorial optimization problem:

Problem 1. *Find*

$$S^* \in \operatorname*{argmax}_{S \subseteq V} \mathcal{F}(S) \textit{ subject to: } \sum_{i \in S} c_i \leq b.$$

Since this is a generalization of the cardinality constraint (where $c_i = 1, \forall i$), this also constitutes a (well-known) NP-hard problem. In this case as well, however, a modified greedy algorithm with partial enumeration can solve Problem 1 near-optimally with $(1-1/e)$-approximation factor if $\mathcal{F}$ is monotone submodular (Sviridenko, 2004). The partial enumeration, however, is too computationally expensive for real world applications. In (Lin and Bilmes, 2010), we generalize the work by Khuller et al. (1999) on the budgeted maximum cover problem to the general submodular framework, and show a practical greedy algorithm with a $(1 - 1/\sqrt{e})$-approximation factor, where each greedy step adds the unit with the largest ratio of objective function gain to scaled cost, while not violating the budget constraint (see (Lin and Bilmes, 2010) for details). Note that in all cases, submodularity and monotonicity are two necessary ingredients to guarantee that the greedy algorithm gives near-optimal solutions.

In fact, greedy-like algorithms have been widely used in summarization. One of the more popular approaches is *maximum marginal relevance* (MMR) (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 ones already selected. Interestingly, the gain function defined in the original MMR paper (Carbonell and Goldstein, 1998) satisfies diminishing returns, a fact apparently unnoticed until now. In particular, Carbonell and Goldstein (1998) define an objective function gain of adding element k to set S ($k \notin S$) as:

$$\lambda Sim_1(s_k, q) - (1 - \lambda) \max_{i \in S} Sim_2(s_i, s_k), \quad (1)$$

where $Sim_1(s_k, q)$ measures the similarity between unit s_k to a query q, $Sim_2(s_i, s_k)$ measures the similarity between unit s_i and unit s_k, and $0 \leq \lambda \leq 1$ is a trade-off coefficient. We have:

Theorem 2. *Given an expression for $\mathcal{F}_{MMR}$ such that $\mathcal{F}_{MMR}(S \cup \{k\}) - \mathcal{F}_{MMR}(S)$ is equal to Eq. 1, $\mathcal{F}_{MMR}$ is non-monotone submodular.*

Obviously, diminishing-returns hold since

$$\max_{i \in S} Sim_2(s_i, s_k) \leq \max_{i \in R} Sim_2(s_i, s_k)$$

for all $S \subseteq R$, and therefore $\mathcal{F}_{\text{MMR}}$ is submodular. On the other hand, $\mathcal{F}_{\text{MMR}}$, would not be monotone, so the greedy algorithm's constant-factor approximation guarantee does not apply in this case.

When scoring a summary at the sub-sentence level, submodularity naturally arises. Concept-based summarization (Filatova and Hatzivassiloglou, 2004; Takamura and Okumura, 2009; Riedhammer et al., 2010; Qazvinian et al., 2010) usually maximizes the weighted credit of concepts covered by the summary. Although the authors may not have noticed, their objective functions are also submodular, adding more evidence suggesting that submodularity is natural for summarization tasks. Indeed, let S be a subset of sentences in the document and denote $\Gamma(S)$ as the set of concepts contained in S. The total credit of the concepts covered by S is then

$$\mathcal{F}_{\text{concept}}(S) \triangleq \sum_{i \in \Gamma(S)} c_i,$$

where c_i is the credit of concept i. This function is known to be submodular (Narayanan, 1997).

Similar to the MMR approach, in (Lin and Bilmes, 2010), a submodular graph based objective function is proposed where a graph cut function, measuring the similarity of the summary to the rest of document, is combined with a subtracted redundancy penalty function. The objective function is submodular but again, non-monotone. We theoretically justify that the performance guarantee of the greedy algorithm holds for this objective function with high probability (Lin and Bilmes, 2010). Our justification, however, is shown to be applicable only to certain particular non-monotone submodular functions, under certain reasonable assumptions about the probability distribution over weights of the graph.

3.2 Summarization with covering constraint

Another perspective is to treat the summarization problem as finding a low-cost subset of the document under the constraint that a summary should cover all (or a sufficient amount of) the information in the document. Formally, this can be expressed as

Problem 2. *Find*

$$S^* \in \operatorname*{argmin}_{S \subseteq V} \sum_{i \in S} c_i \textit{ subject to: } \mathcal{F}(S) \geq \alpha,$$

where c_i are the element costs, and set function $\mathcal{F}(S)$ measure the information covered by S. When $\mathcal{F}$ is submodular, the constraint $\mathcal{F}(S) \geq \alpha$ is called a *submodular cover* constraint. When $\mathcal{F}$ is monotone submodular, a greedy algorithm that iteratively selects k with minimum $c_k/(\mathcal{F}(S \cup \{k\}) - \mathcal{F}(S))$ has approximation guarantees (Wolsey, 1982). Recent work (Shen and Li, 2010) proposes to model document summarization as finding a minimum dominating set and a greedy algorithm is used to solve the problem. The dominating set constraint is also a submodular cover constraint. Define $\delta(S)$ be the set of elements that is either in S or is adjacent to some element in S. Then S is a dominating set if $|\delta(S)| = |V|$. Note that

$$\mathcal{F}_{\text{dom}}(S) \triangleq |\delta(S)|$$

is monotone submodular. The dominating set constraint is then also a submodular cover constraint, and therefore the approaches in (Shen and Li, 2010) are special cases of Problem 2. The solutions found in this framework, however, do not necessarily satisfy a summary's budget constraint. Consequently, a subset of the solution found by solving Problem 2 has to be constructed as the final summary, and the near-optimality is no longer guaranteed. Therefore, solving Problem 1 for document summarization appears to be a better framework regarding global optimality. In the present paper, our framework is that of Problem 1.

3.3 Automatic summarization evaluation

Automatic evaluation of summary quality is important for the research of document summarization as it avoids the labor-intensive and potentially inconsistent human evaluation. ROUGE (Lin, 2004) is widely used for summarization evaluation and it has been shown that ROUGE-N scores are highly correlated with human evaluation (Lin, 2004). Interestingly, ROUGE-N is monotone submodular, adding further evidence that monotone submodular functions are natural for document summarization.

Theorem 3. *ROUGE-N is monotone submodular.*

Proof. By definition (Lin, 2004), ROUGE-N is the n-gram recall between a candidate summary and a set of reference summaries. Precisely, let S be the candidate summary (a set of sentences extracted from the ground set V), $c_e : 2^V \rightarrow \mathbb{Z}_+$ be the number of times n-gram e occurs in summary S, and R_i be the set of n-grams contained in the reference summary i (suppose we have K reference summaries, i.e., $i = 1, \cdots, K$). Then ROUGE-N can be written as the following set function:

$$\mathcal{F}_{\text{ROUGE-N}}(S) \triangleq \frac{\sum_{i=1}^{K} \sum_{e \in R_i} \min(c_e(S), r_{e,i})}{\sum_{i=1}^{K} \sum_{e \in R_i} r_{e,i}},$$

where $r_{e,i}$ is the number of times n-gram e occurs in reference summary i. Since $c_e(S)$ is monotone modular and $\min(x, a)$ is a concave non-decreasing function of x, $\min(c_e(S), r_{e,i})$ is monotone submodular by Theorem 1. Since summation preserves submodularity, and the denominator is constant, we see that $\mathcal{F}_{\text{ROUGE-N}}$ is monotone submodular. □

Since the reference summaries are unknown, it is of course impossible to optimize $\mathcal{F}_{\text{ROUGE-N}}$ directly. Therefore, some approaches (Filatova and Hatzivassiloglou, 2004; Takamura and Okumura, 2009; Riedhammer et al., 2010) instead define "concepts". Alter-

natively, we herein propose a class of monotone submodular functions that naturally models the quality of a summary while not depending on an explicit notion of concepts, as we will see in the following section.

4 Monotone Submodular Objectives

Two properties of a good summary are *relevance* and *non-redundancy*. Objective functions for extractive summarization usually measure these two separately and then mix them together trading off encouraging relevance and penalizing redundancy. The redundancy penalty usually violates the monotonicity of the objective functions (Carbonell and Goldstein, 1998; Lin and Bilmes, 2010). We therefore propose to positively *reward diversity* instead of negatively penalizing redundancy. In particular, we model the summary quality as

$$\mathcal{F}(S) = \mathcal{L}(S) + \lambda \mathcal{R}(S), \tag{2}$$

where $\mathcal{L}(S)$ measures the coverage, or "fidelity", of summary set S to the document, $\mathcal{R}(S)$ rewards diversity in S, and $\lambda \geq 0$ is a trade-off coefficient. Note that the above is analogous to the objectives widely used in machine learning, where a loss function that measures the training set error (we measure the coverage of summary to a document), is combined with a regularization term encouraging certain desirable (e.g., sparsity) properties (in our case, we "regularize" the solution to be more diverse). In the following, we discuss how both $\mathcal{L}(S)$ and $\mathcal{R}(S)$ are naturally monotone submodular.

4.1 Coverage function

$\mathcal{L}(S)$ can be interpreted either as a set function that measures the similarity of summary set S to the document to be summarized, or as a function representing some form of "coverage" of V by S. Most naturally, $\mathcal{L}(S)$ should be monotone, as coverage improves with a larger summary. $\mathcal{L}(S)$ should also be submodular: consider adding a new sentence into two summary sets, one a subset of the other. Intuitively, the increment when adding a new sentence to the small summary set should be larger than the increment when adding it to the larger set, as the information carried by the new sentence might have already been covered by those sentences that are in the larger summary but not in the smaller summary. This is exactly the property of diminishing returns. Indeed, Shannon entropy, as the measurement of information, is another well-known monotone submodular function.

There are several ways to define $\mathcal{L}(S)$ in our context. For instance, we could use $\mathcal{L}(S) = \sum_{i \in V, j \in S} w_{i,j}$ where $w_{i,j}$ represents the similarity between i and j. $\mathcal{L}(S)$ could also be facility location objective, i.e., $\mathcal{L}(S) = \sum_{i \in V} \max_{j \in S} w_{i,j}$, as used in (Lin et al., 2009). We could also use $\mathcal{L}(S) = \sum_{i \in \Gamma(S)} c_i$ as used in concept-based summarization, where the definition of "concept" and the mechanism to extract these concepts become important. All of these are monotone submodular.

Alternatively, in this paper we propose the following objective that does not reply on concepts. Let

$$\mathcal{L}(S) = \sum_{i \in V} \min \left\{ \mathcal{C}_i(S), \alpha\, \mathcal{C}_i(V) \right\}, \tag{3}$$

where $\mathcal{C}_i : 2^V \rightarrow \mathbb{R}$ is a monotone submodular function and $0 \leq \alpha \leq 1$ is a threshold co-efficient. Firstly, $\mathcal{L}(S)$ as defined in Eqn. 3 is a monotone submodular function. The monotonicity is immediate. To see that $\mathcal{L}(S)$ is submodular, consider the fact that $f(x) = \min(x, a)$ where $a \geq 0$ is a concave non-decreasing function, and by Theorem 1, each summand in Eqn. 3 is a submodular function, and as summation preserves submodularity, $\mathcal{L}(S)$ is submodular.

Next, we explain the intuition behind Eqn. 3. Basically, $\mathcal{C}_i(S)$ measures how similar S is to element i, or how much of i is "covered" by S. Then $\mathcal{C}_i(V)$ is just the largest value that $\mathcal{C}_i(S)$ can achieve. We call i "saturated" by S when $\min\{\mathcal{C}_i(S), \alpha\mathcal{C}_i(V)\} = \alpha\mathcal{C}_i(V)$. When i is already saturated in this way, any new sentence j can not further improve the coverage of i even if it is very similar to i (i.e., $\mathcal{C}_i(S \cup \{j\}) - \mathcal{C}_i(S)$ is large). This will give other sentences that are not yet saturated a higher chance of being better covered, and therefore the resulting summary tends to better cover the entire document.

One simple way to define $\mathcal{C}_i(S)$ is just to use

$$\mathcal{C}_i(S) = \sum_{j \in S} w_{i,j} \tag{4}$$

where $w_{i,j} \geq 0$ measures the similarity between i and j. In this case, when $\alpha = 1$, Eqn. 3 reduces to the case where $\mathcal{L}(S) = \sum_{i \in V, j \in S} w_{i,j}$. As we will see in Section 5, having an α that is less than

1 significantly improves the performance compared to the case when $\alpha = 1$, which coincides with our intuition that using a truncation threshold improves the final summary's coverage.

4.2 Diversity reward function

Instead of penalizing redundancy by subtracting from the objective, we propose to reward diversity by adding the following to the objective:

$$\mathcal{R}(S) = \sum_{i=1}^{K} \sqrt{\sum_{j \in P_i \cap S} r_j}. \quad (5)$$

where $P_i, i = 1, \cdots K$ is a partition of the ground set V (i.e., $\bigcup_i P_i = V$ and the P_is are disjoint) into separate clusters, and $r_i \geq 0$ indicates the *singleton reward* of i (i.e., the reward of adding i into the empty set). The value r_i estimates the importance of i to the summary. The function $\mathcal{R}(S)$ rewards diversity in that there is usually more benefit to selecting a sentence from a cluster not yet having one of its elements already chosen. As soon as an element is selected from a cluster, other elements from the same cluster start having diminishing gain, thanks to the square root function. For instance, consider the case where $k_1, k_2 \in P_1$, $k_3 \in P_2$, and $r_{k_1} = 4$, $r_{k_2} = 9$, and $r_{k_3} = 4$. Assume k_1 is already in the summary set S. Greedily selecting the next element will choose k_3 rather than k_2 since $\sqrt{13} < 2 + 2$. In other words, adding k_3 achieves a greater reward as it increases the diversity of the summary (by choosing from a different cluster). Note, $\mathcal{R}(S)$ is distinct from $\mathcal{L}(S)$ in that $\mathcal{R}(S)$ might wish to include certain outlier material that $\mathcal{L}(S)$ could ignore.

It is easy to show that $\mathcal{R}(S)$ is submodular by using the composition rule from Theorem 1. The square root is non-decreasing concave function. Inside each square root lies a modular function with non-negative weights (and thus is monotone). Applying the square root to such a monotone submodular function yields a submodular function, and summing them all together retains submodularity, as mentioned in Section 2. The monotonicity of $\mathcal{R}(S)$ is straightforward. Note, the form of Eqn. 5 is similar to structured group norms (e.g., (Zhao et al., 2009)), recently shown to be related to submodularity (Bach, 2010; Jegelka and Bilmes, 2011).

Several extensions to Eqn. 5 are discussed next: First, instead of using a ground set partition, intersecting clusters can be used. Second, the square root function in Eqn. 5 can be replaced with any other non-decreasing concave functions (e.g., $f(x) = log(1 + x)$) while preserving the desired property of $\mathcal{R}(S)$, and the curvature of the concave function then determines the rate that the reward diminishes. Last, multi-resolution clustering (or partitions) with different sizes (K) can be used, i.e., we can use a mixture of components, each of which has the structure of Eqn. 5. A mixture can better represent the core structure of the ground set (e.g., the hierarchical structure in the documents (Celikyilmaz and Hakkani-tür, 2010)). All such extensions preserve both monotonicity and submodularity.

5 Experiments

The document understanding conference (DUC) (`http://duc.nist.org`) was the main forum providing benchmarks for researchers working on document summarization. The tasks in DUC evolved from single-document summarization to multi-document summarization, and from generic summarization (2001–2004) to query-focused summarization (2005–2007). As ROUGE (Lin, 2004) has been officially adopted for DUC evaluations since 2004, we also take it as our main evaluation criterion. We evaluated our approaches on DUC data 2003-2007, and demonstrate results on both generic and query-focused summarization. In all experiments, the modified greedy algorithm (Lin and Bilmes, 2010) was used for summary generation.

5.1 Generic summarization

Summarization tasks in DUC-03 and DUC-04 are multi-document summarization on English news articles. In each task, 50 document clusters are given, each of which consists of 10 documents. For each document cluster, the system generated summary may not be longer than 665 bytes including spaces and punctuation. We used DUC-03 as our development set, and tested on DUC-04 data. We show ROUGE-1 scores[1] as it was the main evaluation criterion for DUC-03, 04 evaluations.

[1]ROUGE version 1.5.5 with options: -a -c 95 -b 665 -m -n 4 -w 1.2

Documents were pre-processed by segmenting sentences and stemming words using the Porter Stemmer. Each sentence was represented using a bag-of-terms vector, where we used context terms up to bi-grams. Similarity between sentence i and sentence j, i.e., $w_{i,j}$, was computed using cosine similarity:

$$w_{i,j} = \frac{\sum_{w \in s_i} \text{tf}_{w,i} \times \text{tf}_{w,j} \times \text{idf}_w^2}{\sqrt{\sum_{w \in s_i} \text{tf}_{w,s_i}^2 \text{idf}_w^2} \sqrt{\sum_{w \in s_j} \text{tf}_{w,j}^2 \text{idf}_w^2}},$$

where $\text{tf}_{w,i}$ and $\text{tf}_{w,j}$ are the numbers of times that w appears in s_i and sentence s_j respectively, and idf_w is the inverse document frequency (IDF) of term w (up to bigram), which was calculated as the logarithm of the ratio of the number of articles that w appears over the total number of all articles in the document cluster.

Table 1: ROUGE-1 recall (R) and F-measure (F) results (%) on DUC-04. DUC-03 was used as development set.

DUC-04	R	F
$\sum_{i \in V} \sum_{j \in S} w_{i,j}$	33.59	32.44
$\mathcal{L}_1(S)$	39.03	38.65
$\mathcal{R}_1(S)$	38.23	37.81
$\mathcal{L}_1(S) + \lambda \mathcal{R}_1(S)$	**39.35**	**38.90**
Takamura and Okumura (2009)	38.50	-
Wang et al. (2009)	39.07	-
Lin and Bilmes (2010)	-	38.39
Best system in DUC-04 (peer 65)	38.28	37.94

We first tested our coverage and diversity reward objectives separately. For coverage, we use a modular $\mathcal{C}_i(S) = \sum_{j \in S} w_{i,j}$ for each sentence i, i.e.,

$$\mathcal{L}_1(S) = \sum_{i \in V} \min \left\{ \sum_{j \in S} w_{i,j}, \alpha \sum_{k \in V} w_{i,k} \right\}. \quad (6)$$

When $\alpha = 1$, $\mathcal{L}_1(S)$ reduces to $\sum_{i \in V, j \in S} w_{i,j}$, which measures the overall similarity of summary set S to ground set V. As mentioned in Section 4.1, using such similarity measurement could possibly over-concentrate on a small portion of the document and result in a poor coverage of the whole document. As shown in Table 1, optimizing this objective function gives a ROUGE-1 F-measure score 32.44%. On the other hand, when using $\mathcal{L}_1(S)$ with an $\alpha < 1$ (the value of α was determined on DUC-03 using a grid search), a ROUGE-1 F-measure score 38.65% is achieved, which is already better than the best performing system in DUC-04.

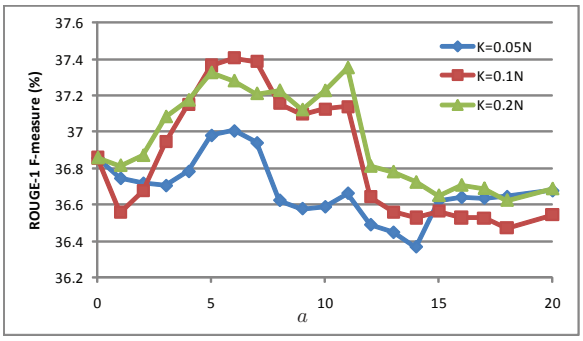

Figure 1: ROUGE-1 F-measure scores on DUC-03 when α and K vary in objective function $\mathcal{L}_1(S) + \lambda \mathcal{R}_1(S)$, where $\lambda = 6$ and $\alpha = \frac{a}{N}$.

As for the diversity reward objective, we define the singleton reward as $r_i = \frac{1}{N} \sum_j w_{i,j}$, which is the average similarity of sentence i to the rest of the document. It basically states that the more similar to the whole document a sentence is, the more reward there will be by adding this sentence to an empty summary set. By using this singleton reward, we have the following diversity reward function:

$$\mathcal{R}_1(S) = \sum_{k=1}^{K} \sqrt{\sum_{j \in S \cap P_k} \frac{1}{N} \sum_{i \in V} w_{i,j}}. \quad (7)$$

In order to generate $P_k, k = 1, \cdots K$, we used CLUTO[2] to cluster the sentences, where the IDF-weighted term vector was used as feature vector, and a direct K-mean clustering algorithm was used. In this experiment, we set $K = 0.2N$. In other words, there are 5 sentences in each cluster on average. And as we can see in Table 1, optimizing the diversity reward function alone achieves comparable performance to the DUC-04 best system.

Combining $\mathcal{L}_1(S)$ and $\mathcal{R}_1(S)$, our system outperforms the best system in DUC-04 significantly, and it also outperforms several recent systems, including a concept-based summarization approach (Takamura and Okumura, 2009), a sentence topic model based system (Wang et al., 2009), and our MMR-styled submodular system (Lin and Bilmes, 2010). Figure 1 illustrates how ROUGE-1 scores change when α and K vary on the development set (DUC-03).

[2]http://glaros.dtc.umn.edu/gkhome/cluto/cluto/overview

Table 2: ROUGE-2 recall (R) and F-measure (F) results (%) on DUC-05, where DUC-05 was used as training set.

DUC-05	R	F
$\mathcal{L}_1(S) + \lambda\mathcal{R}_Q(S)$	**8.38**	**8.31**
Daumé III and Marcu (2006)	7.62	-
Extr, Daumé et al. (2009)	7.67	-
Vine, Daumé et al. (2009)	8.24	-

Table 3: ROUGE-2 recall (R) and F-measure (F) results on DUC-05 (%). We used DUC-06 as training set.

DUC-05	R	F
$\mathcal{L}_1(S) + \lambda\mathcal{R}_Q(S)$	**7.82**	**7.72**
Daumé III and Marcu (2006)	6.98	-
Best system in DUC-05 (peer 15)	7.44	7.43

5.2 Query-focused summarization

We evaluated our approach on the task of query-focused summarization using DUC 05-07 data. In DUC-05 and DUC-06, participants were given 50 document clusters, where each cluster contains 25 news articles related to the same topic. Participants were asked to generate summaries of at most 250 words for each cluster. For each cluster, a title and a narrative describing a user's information need are provided. The narrative is usually composed of a set of questions or a multi-sentence task description. The main task in DUC-07 is the same as in DUC-06.

In DUC 05-07, ROUGE-2 was the primary criterion for evaluation, and thus we also report ROUGE-2[3] (both recall R, and precision F). Documents were processed as in Section 5.1. We used both the title and the narrative as query, where stop words, including some function words (e.g., "describe") that appear frequently in the query, were removed. All queries were then stemmed using the Porter Stemmer.

Note that there are several ways to incorporate query-focused information into both the coverage and diversity reward objectives. For instance, $\mathcal{C}_i(S)$ could be query-dependent in how it measures how much query-dependent information in i is covered by S. Also, the coefficient α could be query and sentence dependent, where it takes larger value when a sentence is more relevant to query (i.e., a larger value of α means later truncation, and therefore more possible coverage). Similarly, sentence clustering and singleton rewards in the diversity function can also

[3]ROUGE version 1.5.5 was used with option -n 2 -x -m -2 4 -u -c 95 -r 1000 -f A -p 0.5 -t 0 -d -l 250

Table 4: ROUGE-2 recall (R) and F-measure (F) results (%) on DUC-06, where DUC-05 was used as training set.

DUC-06	R	F
$\mathcal{L}_1(S) + \lambda\mathcal{R}_Q(S)$	**9.75**	**9.77**
Celikyilmaz and Hakkani-tür (2010)	9.10	-
Shen and Li (2010)	9.30	-
Best system in DUC-06 (peer 24)	9.51	9.51

Table 5: ROUGE-2 recall (R) and F-measure (F) results (%) on DUC-07. DUC-05 was used as training set for objective $\mathcal{L}_1(S) + \lambda\mathcal{R}_Q(S)$. DUC-05 and DUC-06 were used as training sets for objective $\mathcal{L}_1(S) + \sum_\kappa \lambda_\kappa \mathcal{R}_{Q,\kappa}(S)$.

DUC-07	R	F
$\mathcal{L}_1(S) + \lambda\mathcal{R}_Q(S)$	12.18	12.13
$\mathcal{L}_1(S) + \sum_{\kappa=1}^{3} \lambda_\kappa \mathcal{R}_{Q,\kappa}(S)$	12.38	**12.33**
Toutanova et al. (2007)	11.89	11.89
Haghighi and Vanderwende (2009)	11.80	-
Celikyilmaz and Hakkani-tür (2010)	11.40	-
Best system in DUC-07 (peer 15)	12.45	12.29

be query-dependent. In this experiment, we explore an objective with a query-independent coverage function ($\mathcal{R}_1(S)$), indicating prior importance, combined with a query-dependent diversity reward function, where the latter is defined as:

$$\mathcal{R}_Q(S) = \sum_{k=1}^{K} \sqrt{\sum_{j \in S \cap P_k} \left(\frac{\beta}{N} \sum_{i \in V} w_{i,j} + (1-\beta) r_{j,Q} \right)},$$

where $0 \leq \beta \leq 1$, and $r_{j,Q}$ represents the relevance between sentence j to query Q. This query-dependent reward function is derived by using a singleton reward that is expressed as a convex combination of the query-independent score ($\frac{1}{N}\sum_{i \in V} w_{i,j}$) and the query-dependent score ($r_{j,Q}$) of a sentence. We simply used the number of terms (up to a bi-gram) that sentence j overlaps the query Q as $r_{j,Q}$, where the IDF weighting is not used (i.e., every term in the query, after stop word removal, was treated as equally important). Both query-independent and query-dependent scores were then normalized by their largest value respectively such that they had roughly the same dynamic range.

To better estimate of the relevance between query and sentences, we further expanded sentences with synonyms and hypernyms of its constituent words. In particular, part-of-speech tags were obtained for each sentence using the maximum entropy part-of-speech tagger (Ratnaparkhi, 1996), and all nouns were then

expanded with their synonyms and hypernyms using WordNet (Fellbaum, 1998). Note that these expanded documents were only used in the estimation $r_{j,Q}$, and we plan to further explore whether there is benefit to use the expanded documents either in sentence similarity estimation or in sentence clustering in our future work. We also tried to expand the query with synonyms and observed a performance decrease, presumably due to noisy information in a query expression.

While it is possible to use an approach that is similar to (Toutanova et al., 2007) to learn the coefficients in our objective function, we trained all coefficients to maximize ROUGE-2 F-measure score using the Nelder-Mead (derivative-free) method. Using $\mathcal{L}_1(S)+\lambda\mathcal{R}_Q(S)$ as the objective and with the same sentence clustering algorithm as in the generic summarization experiment ($K = 0.2N$), our system, when both trained and tested on DUC-05 (results in Table 2), outperforms the Bayesian query-focused summarization approach and the search-based structured prediction approach, which were also trained and tested on DUC-05 (Daumé et al., 2009). Note that the system in (Daumé et al., 2009) that achieves its best performance (8.24% in ROUGE-2 recall) is a so called "vine-growth" system, which can be seen as an abstractive approach, whereas our system is *purely* an extractive system. Comparing to the extractive system in (Daumé et al., 2009), our system performs much better (8.38% v.s. 7.67%). More importantly, when trained only on DUC-06 and tested on DUC-05 (results in Table 3), our approach outperforms the best system in DUC-05 significantly.

We further tested the system trained on DUC-05 on both DUC-06 and DUC-07. The results on DUC-06 are shown in Table. 4. Our system outperforms the best system in DUC-06, as well as two recent approaches (Shen and Li, 2010; Celikyilmaz and Hakkani-tür, 2010). On DUC-07, in terms of ROUGE-2 score, our system outperforms PYTHY (Toutanova et al., 2007), a state-of-the-art supervised summarization system, as well as two recent systems including a generative summarization system based on topic models (Haghighi and Vanderwende, 2009), and a hybrid hierarchical summarization system (Celikyilmaz and Hakkani-tür, 2010). It also achieves comparable performance to the best DUC-07 system. Note that in the best DUC-07 system (Pingali et al., 2007; Jagarlamudi et al., 2006), an external web search engine (Yahoo!) was used to estimate a language model for query relevance. In our system, no such web search expansion was used.

To further improve the performance of our system, we used both DUC-05 and DUC-06 as a training set, and introduced three diversity reward terms into the objective where three different sentence clusterings with different resolutions were produced (with sizes $0.3N, 0.15N$ and $0.05N$). Denoting a diversity reward corresponding to clustering κ as $\mathcal{R}_{Q,\kappa}(S)$, we model the summary quality as $\mathcal{L}_1(S) + \sum_{\kappa=1}^{3} \lambda_\kappa \mathcal{R}_{Q,\kappa}(S)$. As shown in Table 5, using this objective function with multi-resolution diversity rewards improves our results further, and outperforms the best system in DUC-07 in terms of ROUGE-2 F-measure score.

6 Conclusion and discussion

In this paper, we show that submodularity naturally arises in document summarization. Not only do many existing automatic summarization methods correspond to submodular function optimization, but also the widely used ROUGE evaluation is closely related to submodular functions. As the corresponding submodular optimization problem can be solved efficiently and effectively, the remaining question is then how to design a submodular objective that best models the task. To address this problem, we introduce a powerful class of monotone submodular functions that are well suited to document summarization by modeling two important properties of a summary, fidelity and diversity. While more advanced NLP techniques could be easily incorporated into our functions (e.g., language models could define a better $\mathcal{C}_i(S)$, more advanced relevance estimations for the singleton rewards r_i, and better and/or overlapping clustering algorithms for our diversity reward), we already show top results on standard benchmark evaluations using fairly basic NLP methods (e.g., term weighting and WordNet expansion), all, we believe, thanks to the power and generality of submodular functions. As information retrieval and web search are closely related to query-focused summarization, our approach might be beneficial in those areas as well.

References

F. Bach. 2010. Structured sparsity-inducing norms through submodular functions. *Advances in Neural Information Processing Systems*.

J. Carbonell and J. Goldstein. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proc. of SIGIR*.

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

H. Daumé, J. Langford, and D. Marcu. 2009. Search-based structured prediction. *Machine learning*, 75(3):297–325.

H. Daumé III and D. Marcu. 2006. Bayesian query-focused summarization. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, page 312.

C. Fellbaum. 1998. *WordNet: An electronic lexical database*. The MIT press.

E. Filatova and V. Hatzivassiloglou. 2004. Event-based extractive summarization. In *Proceedings of ACL Workshop on Summarization*, volume 111.

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

J. Jagarlamudi, P. Pingali, and V. Varma. 2006. Query independent sentence scoring approach to DUC 2006. In *DUC 2006*.

S. Jegelka and J. A. Bilmes. 2011. Submodularity beyond submodular energies: coupling edges in graph cuts. In *Computer Vision and Pattern Recognition (CVPR)*, Colorado Springs, CO, June.

D. Kempe, J. Kleinberg, and E. Tardos. 2003. Maximizing the spread of influence through a social network. In *Proceedings of the 9th Conference on SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*.

S. Khuller, A. Moss, and J. Naor. 1999. The budgeted maximum coverage problem. *Information Processing Letters*, 70(1):39–45.

V. Kolmogorov and R. Zabin. 2004. What energy functions can be minimized via graph cuts? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(2):147–159.

A. Krause and C. Guestrin. 2005. Near-optimal nonmyopic value of information in graphical models. In *Proc. of Uncertainty in AI*.

A. Krause, H.B. McMahan, C. Guestrin, and A. Gupta. 2008. Robust submodular observation selection. *Journal of Machine Learning Research*, 9:2761–2801.

H. Lin and J. Bilmes. 2010. Multi-document summarization via budgeted maximization of submodular functions. In *North American chapter of the Association for Computational Linguistics/Human Language Technology Conference (NAACL/HLT-2010)*, Los Angeles, CA, June.

H. Lin and J. Bilmes. 2011. Word alignment via submodular maximization over matroids. In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT)*, Portland, OR, June.

H. Lin, J. Bilmes, and S. Xie. 2009. Graph-based submodular selection for extractive summarization. In *Proc. IEEE Automatic Speech Recognition and Understanding (ASRU)*, Merano, Italy, December.

C.-Y. Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*.

L. Lovász. 1983. Submodular functions and convexity. *Mathematical programming-The state of the art,(eds. A. Bachem, M. Grotschel and B. Korte) Springer*, pages 235–257.

M. Minoux. 1978. Accelerated greedy algorithms for maximizing submodular set functions. *Optimization Techniques*, pages 234–243.

M. Narasimhan and J. Bilmes. 2005. A submodular-supermodular procedure with applications to discriminative structure learning. In *Proc. Conf. Uncertainty in Artifical Intelligence*, Edinburgh, Scotland, July. Morgan Kaufmann Publishers.

M. Narasimhan and J. Bilmes. 2007. Local search for balanced submodular clusterings. In *Twentieth International Joint Conference on Artificial Intelligence (IJCAI07)*, Hyderabad, India, January.

H. Narayanan. 1997. *Submodular functions and electrical networks*. North-Holland.

G.L. Nemhauser, L.A. Wolsey, and M.L. Fisher. 1978. An analysis of approximations for maximizing submodular set functions I. *Mathematical Programming*, 14(1):265–294.

P. Pingali, K. Rahul, and V. Varma. 2007. IIIT Hyderabad at DUC 2007. *Proceedings of DUC 2007*.

V. Qazvinian, D.R. Radev, and A. Ozgür. 2010. Citation Summarization Through Keyphrase Extraction. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 895–903.

A. Ratnaparkhi. 1996. A maximum entropy model for part-of-speech tagging. In *EMNLP*, volume 1, pages 133–142.

K. Riedhammer, B. Favre, and D. Hakkani-Tür. 2010. Long story short-Global unsupervised models for keyphrase based meeting summarization. *Speech Communication*.

C. Shen and T. Li. 2010. Multi-document summarization via the minimum dominating set. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 984–992, Beijing, China, August. Coling 2010 Organizing Committee.

M. Sviridenko. 2004. A note on maximizing a submodular set function subject to a knapsack constraint. *Operations Research Letters*, 32(1):41–43.

H. Takamura and M. Okumura. 2009. Text summarization model based on maximum coverage problem and its variant. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 781–789. Association for Computational Linguistics.

K. Toutanova, C. Brockett, M. Gamon, J. Jagarlamudi, H. Suzuki, and L. Vanderwende. 2007. The PYTHY summarization system: Microsoft research at DUC 2007. In *the proceedings of Document Understanding Conference*.

D. Wang, S. Zhu, T. Li, and Y. Gong. 2009. Multi-document summarization using sentence-based topic models. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 297–300, Suntec, Singapore, August. Association for Computational Linguistics.

L.A. Wolsey. 1982. An analysis of the greedy algorithm for the submodular set covering problem. *Combinatorica*, 2(4):385–393.

P. Zhao, G. Rocha, and B. Yu. 2009. Grouped and hierarchical model selection through composite absolute penalties. *Annals of Statistics*, 37(6A):3468–3497.