Online Summarization via Submodular and Convex Optimization

Ehsan Elhamifar
College of Computer and Information Science
Northeastern University
eelhami@ccs.neu.edu

M. Clara De Paolis Kaluza
College of Computer and Information Science
Northeastern University
clara@ccs.neu.edu

Abstract

We consider the problem of subset selection in the online setting, where data arrive incrementally. Instead of storing and running subset selection on the entire dataset, we propose an incremental subset selection framework that, at each time instant, uses the previously selected set of representatives and the new batch of data in order to update the set of representatives. We cast the problem as an integer binary optimization minimizing the encoding cost of the data via representatives regularized by the number of selected items. As the proposed optimization is, in general, NP-hard and non-convex, we study a greedy approach based on unconstrained submodular optimization and also propose an efficient convex relaxation. We show that, under appropriate conditions, the solution of our proposed convex algorithm achieves the global optimal solution of the non-convex problem. Our results also address the conventional problem of subset selection in the offline setting, as a special case. By extensive experiments on the problem of video summarization, we demonstrate that our proposed online subset selection algorithms perform well on real data, capturing diverse representative events in videos, while they obtain objective function values close to the offline setting.

1. Introduction

Subset selection is the task of finding a subset of most informative items from a ground set. Besides helping to reduce the computational time and memory of algorithms, due to working on a much smaller representative set [1], it has found numerous applications, including, image and video summarization [2, 3, 4], clustering [5, 6, 7, 8, 9], feature and model selection [10, 11, 12], speech and document summarization [13, 14, 15], sensor placement [16, 17], social network marketing [18] and product recommendation [19]. Compared to dictionary learning methods such as Kmeans [20], KSVD [21] and HMMs [22], that *learn* centers/atoms in the input-space, subset selection methods *choose* centers/atoms from the given set of items.

The inputs to subset selection algorithms are in the form of either feature vector representations or pairwise similarities between items. Several subset selection criteria have been studied in the literature, including maximum cut objective [23, 24], maximum marginal relevance [25], capacitated and uncapacitated facility location objectives [26, 27], multi-linear coding [2, 28] and maximum volume subset [14, 29], which all try to characterize the informativeness/value of a subset of items in terms of ability to represent the entire distribution and/or having minimum information overlap among selected items. On the other hand, optimizing almost all subset selection criteria is, in general, NP-hard and non-convex [24, 26, 30], which has motivated the development and study of approximate methods for optimizing these criteria. This includes greedy approximate algorithms [26] for maximizing submodular functions, such as graph-cuts and facility location, which have worst-case approximation guarantees, as well as sampling methods from Determinantal Point Process (DPP) [14, 29, 31], a probability measure on the set of all subsets of a ground set, for approximately finding the maximum volume subset. Motivated by the maturity of convex optimization and advances in sparse and low-rank recovery, recent methods have focused on convex relaxation-based methods for subset selection [5, 6, 32, 33].

Online Subset Selection. Sequential data, including time-series, such as video and speech, and ordered data, such as text, form a significant part of modern datasets. Such datasets often grow incrementally, e.g., new text/document/speech arrive as an event or discussion develops, or video frames constantly get added to a database from surveillance cameras. Given the constant arrival of new data, waiting to gather the entire dataset, in order to perform subset selection and summarization, not only is impractical, but also requires large computational and memory resources. Indeed, we often need to perform real-time learning, inference, decision making and/or planning using representatives. Moreover, memory and computational limitations of devices capturing data do not allow to wait for and collect the entire dataset to perform subset selection.

Thus, there is a need for online subset selection techniques that can select representative items from a dataset as new items arrive and update the set of representatives accordingly, to minimize information overlap among previously selected items and new representatives.

Despite its importance, the problem of online subset selection has not been properly studied in the literature and the existing methods rely on the use of offline subset selection techniques on the new batch of data or greedy methods that select next samples with respect to the previously selected samples [16, 34, 35], hence, ignoring informativeness of new items with respect to the old ones and the possible need for updating the already selected set of items. Notice that standard submoadular methods with constraints on the number of representatives are not generally effective in the online setting, since the number of informative items that should be selected from each batch is different, in general. Moreover, they lead to selection of the maximum number of representatives, which may have information overlap.

Paper Contributions. In this paper, we propose an online subset selection framework that can effectively deal with incremental observations without requiring to store and process the entire data. We cast the problem as an integer binary optimization that incrementally updates the set of representatives using already obtained representatives and newly arrived data. As the proposed optimization is non-convex and, in general, NP-hard, we first investigate an efficient forward-backward greedy method based on submodular optimization, which has suboptimal performance guarantees. Moreover, we propose and study a convex relaxation of the original online subset selection optimization. We prove that, under appropriate conditions, our proposed convex algorithm is exact, i.e., it achieves the global optimal solution of the original non-convex formulation. Our theoretical results not only address online subset selection, but also explain the success of convex relaxations for the offline regime. By extensive experiments on real video data, we demonstrate the effectiveness of our framework for online summarization of videos.

Paper Organization. The organization of the paper is as follows. In Section 2, we review the problem of subset selection in the offline setting. In Section 3, we propose a framework for incremental online subset selection and study greedy and convex relaxations for the problem. In Section 4, we investigate theoretical guarantees of the convex formulation. In Section 5, we demonstrate the effectiveness of our methods for the problem of video summarization. Finally, Section 6 concludes the paper.

2. Subset Selection Review

In this section, we review the problem of subset selection using pairwise dissimilarities. We focus on the facility location objective function, which promotes diversity among representatives, induces clustering of the data and is amenable to convex relaxation. In the next section, we build on this objective function to address the problem of online subset selection.

Let $\mathcal{D}$ denote the set of N items from which we want to select a small representative set. Let d_{ij} denote the dissimilarity between item i and j in $\mathcal{D}$. In other words, d_{ij} indicates how well i represents j (the smaller, the better). We assume that $d_{ij} \geq 0$ and that for every item j, we have $d_{jj} < d_{ij}$ for all $i \neq j$, i.e., each point is the best representative for itself. Our goal is to find a small subset of $\mathcal{D}$, denoted by $\mathcal{S}$, that well represents the entire set, given pairwise dissimilarities $\{d_{ij}\}_{i,j=1,\dots,|\mathcal{D}|}$. The facility location objective achieves this goal by selecting a subset of items and assigning the rest of items to one and only one representative so that the total encoding cost of the set $\mathcal{D}$ via the representatives is minimized. More precisely, we solve

$$\min_{\mathcal{S}\subseteq\mathcal{D}} \lambda|\mathcal{S}| + \sum_{j\in\mathcal{D}} \min_{i\in\mathcal{S}} d_{ij}, \tag{1}$$

where, $\lambda > 0$ is a regularization parameter that puts a trade-off between the size of the representative set, $|\mathcal{S}|$, and the encoding cost of $\mathcal{D}$ via $\mathcal{S}$. Notice that without such a regularization, i.e., with $\lambda = 0$, we obtain the trivial solution of $\mathcal{S} = \mathcal{D}$, where each item is the representative of itself.

The recent work in [5] reformulates (1) as a simultaneous sparse recovery optimization problem, where it introduces binary optimization variables $z_{ij} \in \{0, 1\}$ corresponding to d_{ij}, with z_{ij} indicating if i will be a representative of j, and proposes to solve

$$\begin{aligned} &\min_{\{z_{ij}\}} \ \lambda \sum_{i\in\mathcal{D}} \mathrm{I}(\left\| \begin{bmatrix} z_{i1} & z_{i2} & \cdots \end{bmatrix} \right\|_p) + \sum_{j\in\mathcal{D}} \sum_{i\in\mathcal{D}} d_{ij} z_{ij} \\ &\text{s.t.} \ \ z_{ij} \in \{0, 1\}, \ \sum_{i=1}^{N} z_{ij} = 1, \ \ \forall\, i, j \in \mathcal{D}. \end{aligned} \tag{2}$$

Similar to (1), the first term in the above optimization counts the number of representatives ($\mathrm{I}(\cdot)$ denotes the indicator function) and the second term corresponds to the encoding of the set $\mathcal{D}$ via the representatives. We can obtain representatives from (2) as items i for which $z_{ij} = 1$ for some j in $\mathcal{D}$. We denote the set of representatives by $\mathcal{R}$. Moreover, (2) induces a clustering of the set $\mathcal{D}$, where each cluster is formed by each representative $i \in \mathcal{R}$ and items $j \in \mathcal{D}$ for which $z_{ij} = 1$.

Since (2) is, in general, NP-hard and non-convex, [5] has proposed a convex relaxation, by dropping the indicator function and relaxing the binary constraints on variables to

$z_{ij} \in [0,1]$. More precisely, one solves the optimization

$$\begin{aligned} \min_{\{z_{ij}\}} \ & \lambda \sum_{i\in\mathcal{D}} \left\| \begin{bmatrix} z_{i1} & z_{i2} & \cdots \end{bmatrix} \right\|_p + \sum_{j\in\mathcal{D}} \sum_{i\in\mathcal{D}} d_{ij} z_{ij} \\ \text{s.t.} \ & z_{ij} \geq 0, \ \sum_{i\in\mathcal{D}} z_{ij} = 1, \ \forall\, i,j \in \mathcal{D}, \end{aligned} \tag{3}$$

which is convex for $p \geq 1$. Indeed, under appropriate conditions on pairwise dissimilarities, (3) obtains the true clustering of the data, as shown in [5].

It is important to note that (3) performs subset selection in the offline regime, where the entire set $\mathcal{D}$ is available. Moreover, the theoretical guarantees under which (3) is equivalent to the original problem (2) is unknown. To address these problems, in the next section, we propose an incremental subset selection algorithm to deal with online observations, study greedy and convex relaxations for the problem, and propose theoretical results that guarantee the equivalence of the convex formulation with the original problem in both online and offline settings.

3. Online Subset Selection

Assume we have a sequential set of items that arrive in an incremental fashion. Our goal is to select a small subset of items in an online fashion that effectively represents the entire dataset. Let $\mathcal{D}_o^{(t)}$ denote the set of data points arrived prior to time t and $\mathcal{D}_n^{(t)}$ denote the set of newly arrived items at time t. To perform subset selection, we ideally would like to run the offline algorithm in (2) or its convex relaxation in (3) on the entire dataset available thus far, i.e., $\mathcal{D}_o^{(t)} \cup \mathcal{D}_n^{(t)}$. However, as t grows, the set $\mathcal{D}_o^{(t)}$ grows, which has the drawback of increasing the computational time and memory for running the offline subset selection.

In this section, we propose an algorithm that updates representatives at time t using previously selected representatives, denoted by $\mathcal{E}_o^{(t)}$, and the set $\mathcal{D}_n^{(t)}$, which significantly reduces the computational time and memory, especially in cases where $\mathcal{D}_n^{(t)}$ is of small or moderate size compared to $\mathcal{D}_o^{(t)}$. More specifically, starting from $\mathcal{D}_o^{(0)}$, we perform subset selection to obtain $\mathcal{E}_o^{(0)}$ and at time t, we propose a framework that uses old representatives $\mathcal{E}_o^{(t)}$ and new set $\mathcal{D}_n^{(t)}$ to update the set of representatives, forming $\mathcal{E}_o^{(t+1)}$ that will be used in the next time index. To do so, we need to make sure that representatives selected from $\mathcal{D}_n^{(t)}$ are not redundant with respect to previous representatives, $\mathcal{E}_o^{(t)}$.

Since we only need to focus on the formulation at time t, for simplicity of notation, we use $\mathcal{E}_o$ instead of $\mathcal{E}_o^{(t)}$, and similarly use $\mathcal{D}_o$, $\mathcal{D}_n$. With abuse of notation, we refer to both items and indices of items using $\mathcal{D}_o$, $\mathcal{D}_n$ and $\mathcal{E}_o$. Let $\{d_{ij}^{o,o}\}_{i,j\in\mathcal{E}_o}$ denote dissimilarities between old representatives, $\{d_{ij}^{o,n}\}_{i\in\mathcal{E}_o, j\in\mathcal{D}_n}$ denote dissimilarities between old representatives and new data, $\{d_{ij}^{n,o}\}_{i\in\mathcal{D}_n, j\in\mathcal{E}_o}$ is dissimilarities between new data and old representatives and $\{d_{ij}^{n,n}\}_{i,j\in\mathcal{D}_n}$ denote dissimilarities between new data. In the paper, we let dissimilarities be asymmetric and/or violate the triangle inequality.

To address the problem of incremental subset selection, we propose an optimization by defining variables $\mathcal{Z} \triangleq \{\{z_{ij}^{o,o}\}, \{z_{ij}^{o,n}\}, \{z_{ij}^{n,o}\}, \{z_{ij}^{n,n}\}\}$ associated with dissimilarities. We consider the following encoding cost function

$$\begin{aligned} J_{\text{enc}} \triangleq & \sum_{i\in\mathcal{E}_o} \sum_{j\in\mathcal{E}_o} d_{ij}^{o,o} z_{ij}^{o,o} + \sum_{i\in\mathcal{E}_o} \sum_{j\in\mathcal{D}_n} d_{ij}^{o,n} z_{ij}^{o,n} \\ & + \sum_{i\in\mathcal{D}_n} \sum_{j\in\mathcal{E}_o} d_{ij}^{n,o} z_{ij}^{n,o} + \sum_{i\in\mathcal{D}_n} \sum_{j\in\mathcal{D}_n} d_{ij}^{n,n} z_{ij}^{n,n}, \end{aligned} \tag{4}$$

which measures the total cost of encoding old representatives and new data via representatives selected from $\mathcal{E}_o \cup \mathcal{D}_n$. Similar to the offline regime, we need to restrict the size of the representative set from $\mathcal{E}_o \cup \mathcal{D}_n$. However, unlike before, items in $\mathcal{E}_o$ have already shown to have representative power, i.e., they are representatives of items in $\mathcal{D}_o \backslash \mathcal{E}_o$, hence, selecting items from $\mathcal{E}_o$ should not be penalized. On the other hand, we only need to select representatives from $\mathcal{D}_n$ as long as items in $\mathcal{E}_o$ do not have sufficient representation power. Thus, we only need to penalize the number of representatives selected from $\mathcal{D}_n$.

$$\begin{aligned} \min_{\mathcal{Z}} \ & J_{\text{enc}} + \lambda \sum_{i\in\mathcal{D}_n} \mathrm{I}(\left\| \begin{bmatrix} z_{i1}^{n,o} & z_{i2}^{n,o} & \cdots & z_{i1}^{n,n} & z_{i2}^{n,n} & \cdots \end{bmatrix} \right\|_p) \\ \text{s.t.} \ & z_{ij}^{o,o},\ z_{ij}^{n,o},\ z_{ij}^{o,n},\ z_{ij}^{n,n} \in \{0,1\},\ \forall i,\ j \\ & \sum_{i\in\mathcal{E}_o} z_{ij}^{o,o} + \sum_{i\in\mathcal{D}_n} z_{ij}^{n,o} = 1, \ \forall\, j \in \mathcal{E}_o, \\ & \sum_{i\in\mathcal{E}_o} z_{ij}^{o,n} + \sum_{i\in\mathcal{D}_n} z_{ij}^{n,n} = 1, \ \forall\, j \in \mathcal{D}_n, \end{aligned} \tag{5}$$

where the first term in the objective function measures the encoding cost of $\mathcal{E}_o \cup \mathcal{D}_n$ via representatives selected from $\mathcal{E}_o \cup \mathcal{D}_n$ and the second term penalizes the number of representatives selected from $\mathcal{D}_n$. The constraints ensure that selection variables are binary and that each point in $\mathcal{E}_o \cup \mathcal{D}_n$ must be represented by one representative. In other words, the effect of the proposed optimization is to use old representatives to represent new items, as long as the associated encoding cost is sufficiently small (we will quantify this later), and add new representatives from $\mathcal{D}_n$ when old representatives are insufficient, e.g., when new clusters appear in data. Notice that a sufficiently large regularization parameter λ promotes selecting only old representatives, $\mathcal{E}_o$, while a small λ promotes selecting a larger number of new representatives from $\mathcal{D}_n$.[1]

[1] We can allow old representatives to be removed and updated by putting a small regularization on the selection of old representatives.

We can show that the solution of the optimization in (5) always finds $z_{ii}^{o,o} = 1$, for all $i \in \mathcal{E}_o$, and $z_{ij}^{n,o} = 0$, for all $i \in \mathcal{D}_n$ and $j \in \mathcal{E}_o$. In other words, each old representative will always be selected as the representative of itself. Hence, we consider the simpler optimization

$$\begin{aligned} \min_{\mathcal{Z}'} \; & J'_{\text{enc}} + \lambda \sum_{i \in \mathcal{D}_n} \mathrm{I}(\left\| \begin{bmatrix} z_{i1}^{n,n} & z_{i2}^{n,n} & \cdots \end{bmatrix} \right\|_p) \\ \text{s.t.} \; & z_{ij}^{o,n},\, z_{ij}^{n,n} \in \{0,1\}, \;\; \forall i,\, j, \\ & \sum_{i \in \mathcal{E}_o} z_{ij}^{o,n} + \sum_{i \in \mathcal{D}_n} z_{ij}^{n,n} = 1, \;\; \forall\, j \in \mathcal{D}_n, \end{aligned} \tag{6}$$

over a smaller set of optimization variables, $\mathcal{Z}' \triangleq \{\{z_{ij}^{o,n}\}, \{z_{ij}^{n,n}\}\}$, with J'_{enc} being defined as

$$J'_{\text{enc}} \triangleq \sum_{i \in \mathcal{E}_o} \sum_{j \in \mathcal{D}_n} d_{ij}^{o,n} z_{ij}^{o,n} + \sum_{i \in \mathcal{D}_n} \sum_{j \in \mathcal{D}_n} d_{ij}^{n,n} z_{ij}^{n,n}, \tag{7}$$

measuring the encoding cost of new data, $\mathcal{D}_n$, via items in $\mathcal{E}_o \cup \mathcal{D}_n$. Indeed, we can show the following result.

Proposition 1 *The optimization programs* (5) *and* (6) *are equivalent, in that they obtain the same solutions for* $\{z_{ij}^{o,n}\}$ *and* $\{z_{ij}^{n,n}\}$.

Notice that the solution of the above optimization will determine the representatives and clustering of the data at the same time. More specifically, optimization variables $z_{ij}^{n,n}$ that are equal to 1 indicate that $i \in \mathcal{D}_n$ is a representative (we already know that all points in $\mathcal{E}_o$ will remain representatives). We denote the set of all representatives by $\mathcal{R}$. For $j \in \mathcal{D}_n$, we denote the representative of j by $M(j)$. In other words, we always have $z_{M(j)j} = 1$. We also obtain clustering of the data, where the ℓ-th group corresponds to items that are assigned to the ℓ-th representative.

3.1. Greedy Unconstrained Submodular Optimization Algorithm

In this section, we discuss an efficient algorithm based on unconstrained submodular optimization for solving the proposed online optimization in (6). To do so, note that we can write (6) in the equivalent form

$$\min_{\mathcal{S} \subseteq \mathcal{D}_n} f(\mathcal{S}), \tag{8}$$

where the function $f(\mathcal{S})$ is defined as

$$f(\mathcal{S}) \triangleq \sum_{j \in \mathcal{D}_n} \min \; \{\min_{i \in \mathcal{E}_o} d_{i,j}^{o,n}, \; \min_{i \in \mathcal{S}} d_{i,j}^{n,n}\} + \lambda |\mathcal{S}|. \tag{9}$$

It is important to note that the online optimization in (9), despite being non-convex and, in general, NP-hard, is submodular. In other words, $f(\mathcal{S})$ satisfies the diminishing return property, i.e., $J(\mathcal{S} \cup \{\ell\}) - J(\mathcal{S}) \geq J(\mathcal{T} \cup \{\ell\}) - J(\mathcal{T})$ for any $\mathcal{S} \subseteq \mathcal{T} \subseteq \mathcal{D}_n \backslash \{\ell\}$. Since (9) is a an unconstrained submodular optimization, we use the randomized linear-time algorithm, proposed in [36], described in Algorithm 1. Here, $\mathcal{D}_n^\ell$ denotes the ℓ-th item of $\mathcal{D}_n$. The above algorithm is a randomized greedy method that scans the entire dataset once and decides whether to include an item in the representative set. In fact, the greedy approach has a 0.5 approximation guarantee, i.e., its solution is always 0.5 or closer to the optimal cost of the original optimization in (9).

Algorithm 1 : Randomized Greedy Algorithm for Unconstrained Submodular Optimization

Input: Submodular function $f(\cdot)$ which is being maximized without constraints over the set $\mathcal{D}_n$.

Initialize: $\mathcal{X}_0 = \varnothing$ and $\mathcal{Y}_0 = \mathcal{D}_n$.
for $\ell = 1, \ldots, |\mathcal{D}_n|$ **do**
  $a_\ell \leftarrow \max\{f(\mathcal{X}_{\ell-1} \cup \mathcal{D}_n^\ell) - f(\mathcal{X}_{\ell-1}), 0\}$
  $b_\ell \leftarrow \max\{f(\mathcal{Y}_{\ell-1} \setminus \mathcal{D}_n^\ell) - f(\mathcal{Y}_{\ell-1}), 0\}$
  With probability $a_\ell/(a_\ell + b_\ell)$ do:
    $\mathcal{X}_\ell \leftarrow \mathcal{X}_{\ell-1} \cup \{\mathcal{D}_n^\ell\}, \; \mathcal{Y}_\ell \leftarrow \mathcal{Y}_{\ell-1}$
  Else (with probability $b_\ell/(a_\ell + b_\ell)$) do:
    $\mathcal{X}_\ell \leftarrow \mathcal{X}_{\ell-1}, \; \mathcal{Y}_\ell \leftarrow \mathcal{Y}_{\ell-1} \backslash \mathcal{D}_n^\ell$
end for

Output: Set of representatives from $\mathcal{D}_n$ indexed by $\mathcal{X}_{|\mathcal{D}_n|}$.

3.2. Convex Relaxation-based Algorithm

It is important to note that the proposed online optimizations in (5) and (6) are, in general, NP-hard and non-convex, due to counting operations on the number of nonzero optimization vectors as well as the binary constraints on optimization variables. To address the problem efficiently, we propose a convex relaxation, where we drop the indicator function and relax the optimization variables to be in $[0,1]$ instead of $\{0,1\}$. More specifically, we propose to solve

$$\begin{aligned} \min \; & J'_{\text{enc}} + \lambda \sum_{i \in \mathcal{D}_n} \left\| \begin{bmatrix} z_{i1}^{n,n} & z_{i2}^{n,n} & \cdots \end{bmatrix} \right\|_p \\ \text{s.t.} \; & z_{ij}^{o,n},\, z_{ij}^{n,n} \in [0,1], \;\; \forall i,\, j, \\ & \sum_{i \in \mathcal{E}_o} z_{ij}^{o,n} + \sum_{i \in \mathcal{D}_n} z_{ij}^{n,n} = 1, \;\; \forall\, j \in \mathcal{D}_n. \end{aligned} \tag{10}$$

We choose $p \in \{2, \infty\}$, for both of which the above optimization is convex. As we will show through theoretical analysis, the choice of the ℓ_p-norm affects the structure of representatives. While $p = \infty$ promotes selecting the medoid of each cluster, $p = 2$ allows to deviate from the medoid in favor of achieving unbalanced cluster sizes.

4. Theoretical Analysis

In this section, we investigate conditions under which the solution of our proposed convex optimization in (10) is

equivalent to the original non-convex optimization in (6). We present the results for $p = \infty$, however, the analysis for $p = 2$ is similar and involves an extra condition.

Before analyzing the convex algorithms, we study the properties of the solution of the non-convex optimization in (6), which is the solution we would like to achieve by solving the relaxation. We show that for the solution of (6), the following conditions must hold.

Theorem 1 *The solution of the optimization program in (6) satisfies the following conditions:*

1. *For every $i \in \mathcal{D}_n$, where $i \notin \mathcal{R}$ and $M(i) \in \mathcal{D}_n$, we have $\sum_{j:M(j)=M(i)} d^{n,n}_{M(i)j} \leq \sum_{j:M(j)=M(i)} d^{n,n}_{ij}$;*
2. *For every $i \in \mathcal{D}_n$, where $i \notin \mathcal{R}$ and $M(i) \in \mathcal{E}_o$, we have $\sum_{j:M(j)=M(i)} (d^{o,n}_{M(i)j} - d^{n,n}_{ij})_+ \leq \lambda$.*

Roughly speaking, the above conditions are definitions of medoids with respect to representatives from $\mathcal{D}_n$ and $\mathcal{E}_o$. The first condition in the above theorem is the conventional definition of the medoid of a group, i.e., the item in the group that achieves the minimum encoding cost. The second condition characterizes the medoid of a group in $\mathcal{D}_n$ represented by an item in $\mathcal{E}_o$. It states that each representative from $\mathcal{E}_o$ encodes items in $\mathcal{D}_n$ by a cost that is at most λ larger than the best encoding we can achieve by assigning a representative from $\mathcal{D}_n$ instead of $\mathcal{E}_o$. Otherwise, we can assign all items in the group to the item in $\mathcal{D}_n$ that obtains this minimum encoding cost, hence, decreasing the first term in the objective function by more than λ, while the additional representative increases the second term in the objective function by only λ, hence, a lower overall cost. Notice that the above result also applies to the offline setting, where $\mathcal{E}_o$ is empty. In that case, we only have the first condition of Theorem 1 being satisfied.

Next, we study conditions under which the convex relaxation in (10) with $p = \infty$ recovers the solution of the optimization program in (6).

Theorem 2 *The solution of the optimization program (10) is equivalent to (6) for a given λ and for $p = \infty$, if all the following conditions hold:*

1. *For every $i \in \mathcal{D}_n$ where $i \notin \mathcal{R}$ and for every j where $M(j) \neq M(i)$ and $M(j) \in \mathcal{D}_n$, we have $\frac{\lambda}{N_{M(j)}} + d^{n,n}_{M(j)j} < d^{n,n}_{ij}$;*
2. *For every $i \in \mathcal{D}_n$ where $i \notin \mathcal{R}$, $M(i) \in \mathcal{D}_n$ and for every j where $M(j) = M(i)$, we have $\frac{\lambda}{N_{M(j)}} + d^{n,n}_{M(j)j} \geq d^{n,n}_{ij}$;*
3. *For every $i \in \mathcal{D}_n$ where $i \notin \mathcal{R}$ and for every j where $M(j) \neq M(i)$ and $M(j) \in \mathcal{E}_o$, we have $d^{o,n}_{M(j)j} < d^{n,n}_{ij}$.*

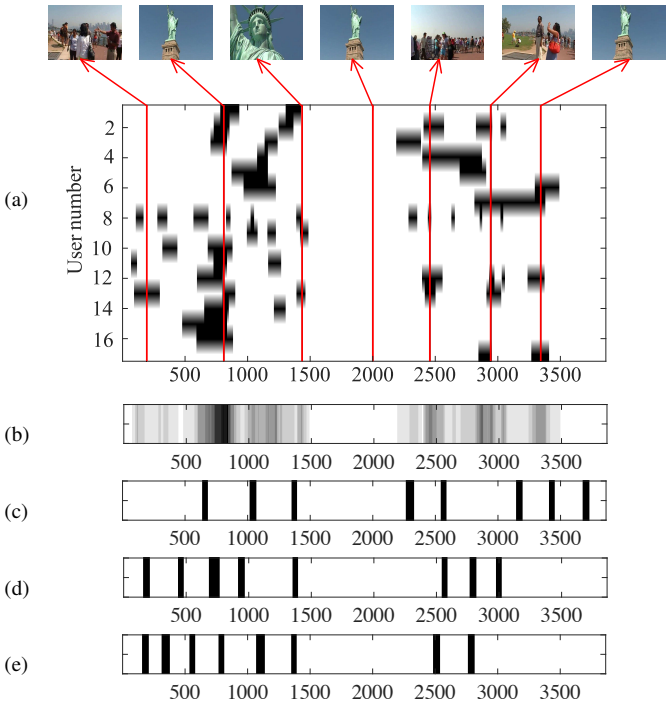

Figure 1: Ground truth summaries for the video "Statue of Liberty" from the SumMe dataset [37] (a) frames selected for summary by each user, with black indicating a chosen frame. A few illustrative frames shown above with a red line indicating their position in the timeline. Note that ground truths vary from user to user but there is overlap among segments across most users. (b) Composite ground truth, calculated by averaging all user summaries. For (c-g) regularization was chosen to match average user summary length. (c) Summary selected by the offline convex method with $p = \infty$ and $\alpha = 0.095$. (d) Online convex method summary with $p = \infty$, $\alpha = 0.38$, batch size of 15 superframes, about 20% of video length. (e) Online convex method summary with $p = \infty$, $\alpha = 0.77$, batch size of 8, about 10% of video length.

The first condition states that the closest item from other groups to a group represented by a new representative is sufficiently far from it. The second condition states that items in the same group are not far from each other, i.e., each item in group j is at most $\lambda/N_{M(j)}$ away from the representative of the group. Finally, the last condition states that items represented by new representatives must be sufficiently far from items represented by old representatives.

Remark 1 *The results presented in this section also apply to the offline setting, where $\mathcal{E}_o$ is empty. In that case, only the first condition holds in Theorem 1 and the first two conditions must hold in Theorem 2.*

5. Experiments

In this section, we evaluate and compare the performance of our proposed convex relaxation and submodular algo-

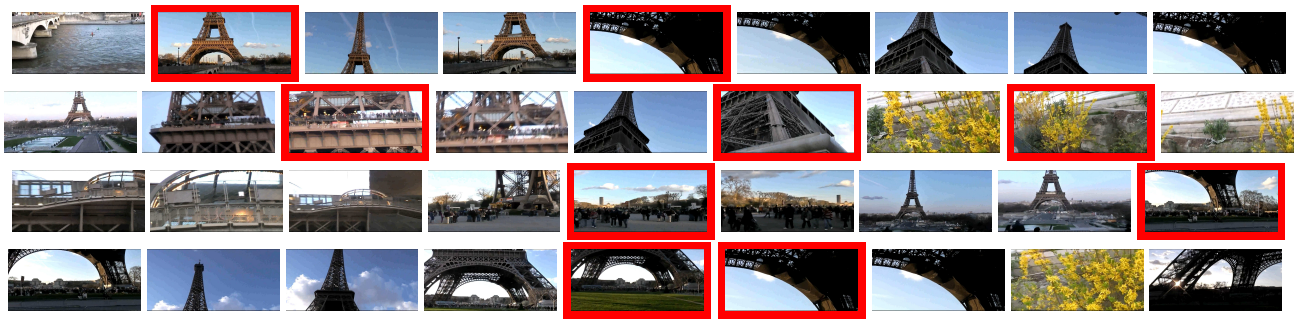

Figure 2: Summary for the "Eiffel Tower" video from the SumMe dataset for online convex method with $p = \infty$ with regularization parameter $\alpha = 0.445$. A representative frame is shown to represent a superframe. A red border designates a superframe chosen to be in the summary. Batch size was fixed to about 10% of the video, i.e., we process 10 superframes per batch for the video of 92 superframes.

Figure 3: Superframes chosen by the online submodular method for the same video as in Figure 2. Regularization parameter ($\alpha = 0.855$) has been set to choose the same number of superframes as the convex method with $p = \infty$. Batch size was fixed to about 10% of the video.

rithms for online summarization. We also compare these online methods to the corresponding offline summarization methods. Lastly, we investigate the effect of the choice of the ℓ_p-norm $p \in \{2, \infty\}$ and the effect of the regularization parameter λ on the online summary selection. To this end, we apply our methods to two real-world video datasets and report the results.

5.1. Datasets and preprocessing

We evaluate the performance of our methods on two real-world datasets, namely the SumMe [37] and the TVSum50 [38] datasets. The SumMe dataset contains 25 videos ranging in length from under 1 minute to just over 6 minutes and is comprised of videos with stationary, moving, or egocentric camera set-ups for varying subject categories. Each video has at least 15 accompanying human supplied summaries, where each participant selected between 5% and 15% of the frames in each video to include in their summary. These human supplied summaries serve as the ground truth summaries for the dataset. The TVSum dataset contains 50 videos from different categories and varying in length between 2 and 10 minutes. Here, at least 20 human evaluations are provided for each video. Rather than providing summaries of videos, participants ranked the relative importance of each two-second segment in every video.

Following the experimental set-up in [38], we use these segment rankings to select the top 15% of segments to serve as the human summaries for each video. To evaluate the performance of our methods against these user supplied summaries, we calculate a composite, or average, ground truth summary across all users. The composite summary consists of a score for each frame in the video corresponding to the percent of users that selected that frame for the summary. Figure 1a illustrates the ground truth summaries for all users for one video in the SumMe dataset and Figure 1b shows the corresponding composite ground truth summary.

For both datasets, we segment each video into superframes as described in [37], where each superframe is a sequence of frames chosen to produce natural cuts between shots. For each superframe, we extract Convolutional 3D (C3D) features as described in [39]. We use these features to produce the dissimilarity matrix $\mathcal{D}$ where $(\mathcal{D})_{ij}$ is the Euclidean distance between the feature vector of superframe i and that of superframe j. Thus, our methods generate summaries at a superframe level. That is, the automatically generated summaries are sets of superframes selected to summarize each video. We transform the superframe-level summary to a frame-level summary by taking all frames contained within a chosen superframe to be in the summary.

To select representatives via our proposed methods, we set $\lambda = \alpha\lambda_{\max}$, for $\alpha > 0$, where $\lambda_{\max}$ is the regularization parameter value for which we select one representative. We determine $\lambda_{\max}$ analytically similar to [5]. Figures 1c–1e demonstrate the selected frames by offline and online convex methods for the video "Statue of Liberty" in SumMe dataset. As the results show, the online method for different batch sizes selects frames of the video that coincide well with human ground-truth summries. Figures 2 and 3 demonstrate diverse automatic summaries obtained by the online convex and submodular methods, respectively, where 9 representative superframes are chosen. In each case, a representative frame is shown for each selected superframe.

5.2. Evaluating error with ground truth

To evaluate the performance of our proposed methods, we compare the automatically generated summaries with

		Convex, $p=\infty$			Convex, $p=2$			Greedy Submodular		
	Human	Offline	Baseline	Online	Offline	Baseline	Online	Offline	Baseline	Online
SumMe	0.03220 (0.2241)	0.00869 (0.1936)	-0.0211 (0.1493)	0.0230 (0.2200)	-0.0273 (0.1596)	-0.0282 (0.1610)	0.0291 (0.2720)	0.0263 (0.1984)	-0.0071 (0.1595)	0.0142 (0.1976)
TVSum	0.1696 (0.4763)	0.0159 (0.1830)	0.00055 (0.1525)	0.02303 (0.1970)	0.00524 (0.1834)	-0.00777 (0.1669)	-0.00357 (0.2469)	0.005016 (0.1695)	0.007184 (0.1745)	0.02446 (0.1885)

Table 1: Average MCC across all user summaries and all videos for SumMe and TVSum datasets. The mean MCC compared to all human summaries across all videos is listed first. We also compute for each video the MCC for the human summary with which the automatic summary most agrees, this is shown in parentheses. The first column shows the agreement among human summaries.

the ground truth summaries in each dataset. Because we do not explicitly restrict the size of the summaries generated by our methods, the size of the automatic summaries may differ greatly from the approximately fixed-size ground truth summaries. The disparity in the size of the summaries being compared renders the traditional F-measure used in e.g.[37] and [38] not well suited as a measure of agreement. In particular, the F-measure is a function of precision and recall and does not incorporate the *specificity* of a summary. The specificity, or true negative rate, is a measure of how well a method can identify negative examples (in this case, a frame *not* in the summary). If the size of the automatic summary is not restricted, the F-measure will favor larger summaries. Consider the extreme case where the automatic summary selects all frames to be in the summary. In this case recall is 1 since the method did identify all the "true" summary frames and the precision is equal to the size of the ground truth summary. The F-measure fails here because there is no penalty for missing "negative" examples.

As an alternative to the f-measure, we use the Matthews correlation coefficient (MCC) [40] defined as follows

$$\text{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \tag{11}$$

where TP is the number of true positives (the number of frames correctly identified as belonging to the summary), TN is the number of true negatives (the number of frames correctly identified as *not* belonging to the summary), FP is the number of false positives (the number of frames chosen by the algorithm to be in the summary that were not in the ground truth summary), and finally FN is the number of false negatives (the number of frames in the ground truth summary but not in the automatic summary). The MCC has a value between 1, corresponding to a perfect agreement with the ground truth, and -1, corresponding to a perfect disagreement with the ground truth.

5.3. Results

To evaluate the proposed online methods, we compare the MCC obtained for each video in the datasets for i) the corresponding offline summarization method; ii) a baseline online summarization where summaries are chosen for each batch without consideration of previously chosen representatives; iii) the proposed online method. For the online and baseline summarization methods, the batch size is fixed to be 10% of the superframes in the video. For each method, regularization was chosen to select a superframe summary as close as possible to 15% of the frames in the video to match the size of the ground truth summaries.

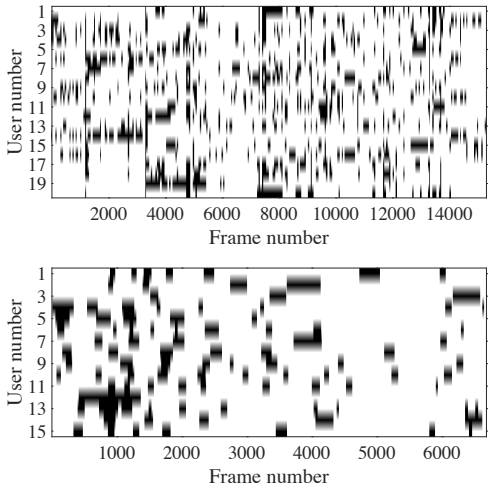

Figure 4: User summaries for a sample video from TVSum (top) and SumMe (bottom) datasets. User summaries vary widely. Some users prefer fewer, longer segments, some prefer more, shorter segments, and some use both techniques to summarize.

The average MCC results for the SumMe and TVSum datasets are listed in Table 1. We list the mean MCC across all users and all videos. To address the low agreement between users, we also show the average best MCC across videos. This measure shows the average agreement with the human summary that is most similar to the automated summary. The disparity between the mean agreement (MCC) across all videos and all users and the mean best agreement show the limitations of using users summaries as ground truth to evaluate summaries. In particular, a user may summarize each video in a different manner and each user may summarize a particular video differently, as shown in Figure 4. Table 1 also lists the mean MCC between human sum-

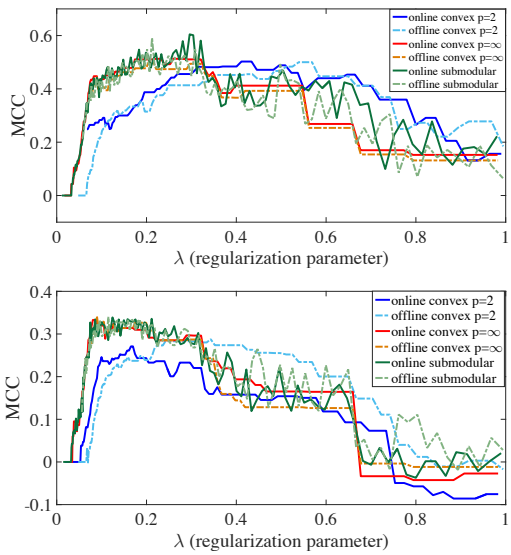

Figure 5: MCC for greedy submodular and convex with $p \in \{2, \infty\}$ methods in an online and offline setting. Batch size for online methods is 10% (top) and 20% (bottom) of total superframes.

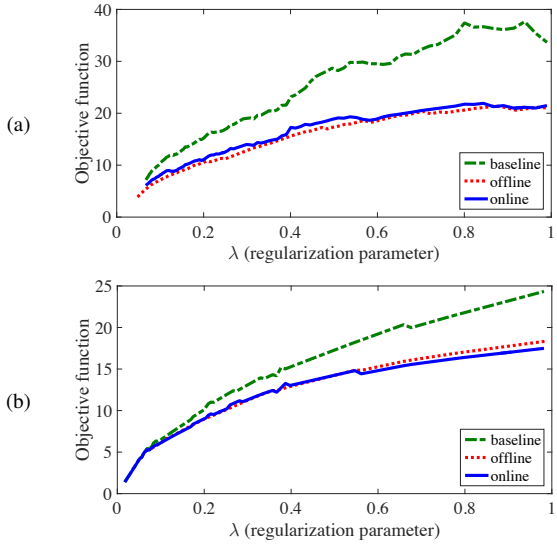

Figure 6: Objective function value of the convex method with (a) $p = 2$, (b) $p = \infty$ for summarizing the video "Car over camera" from the SumMe dataset. The online methods achieve values similar to the corresponding offline methods. The baseline obtains a higher cost, corresponding to long and/or repetitive summaries.

maries and the corresponding average best MCC between human summaries. The performance of the methods compared to the user summaries vary due to the variability of the user summaries themselves. Notice that the performance of the proposed online methods is close to the offline setting and better than baseline, in general. The offline method optimizes the objective function considering the entire video, yet achieves lower performance than the online method for $p = \infty$ for both datasets, for $p = 2$ for SumMe, and for submodular for TVSum. This result can be attributed to the fact that human summaries are generated "online" in the sense that users summarized the videos as they played, though they were allowed to return to previous frames to amend their summaries. However, the disparity among human summaries may be responsible for the untuitive results and thus we also analyze the objective value achieved by the methods, which will be discussed below. The objective function captures the encoding cost and the penalty for too-long summaries, therefore can be interpreted more directly than when relying on disparate ground truth summaries.

Figure 5 demonstrates the MCC values for the online and offline settings for two batch sizes. The results show that there is a wide range of λ for which the proposed methods perform well. Moreover, notice that the convex method with $p = \infty$ and the greedy submodular method show closer agreement than the convex method with $p = 2$.

Lastly, Figure 6 shows the value of the original non-convex objective function in (6) for the baseline, online and offline settings. As the results demonstrate, the online method achieves close objective values to the offline method, while it obtains significantly lower cost than the baseline. Small values of λ indicate a small penalty for selecting a long summary, thus all methods perform similarly by choosing most superframes for the summary. As the regularization increases, longer summaries are penalized more strongly. The close agreement between the online and offline settings serves to validate the effectiveness of our online methods, where representative frames for each online batch are selected among previously chosen representative frames and the frames from the current batch. These results also serve as an alternative method for evaluating automated summaries, which is especially useful when a human ground truth is not available or agreement between human summaries is low.

6. Conclusions

We studied the problem of subset selection in the online setting, where data arrive incrementally. We proposed an incremental subset selection framework that, at each time instant, uses the previously selected set of representatives and the new batch of data in order to update the set of representatives. We cast the problem as an integer binary optimization minimizing the encoding cost of the data via representatives regularized by the number of selected items. We studied a randomized greedy approach based on unconstrained submodular optimization and proposed a convex algorithm with theoretical performance guarantees. By experiments on real videos, we demonstrated the effectiveness of our methods for online video summarization.

References

[1] S. Garcia, J. Derrac, J. R. Cano, and F. Herrera, "Prototype selection for nearest neighbor classification: Taxonomy and empirical study," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 3, pp. 417–435, 2012. 1

[2] E. Elhamifar, G. Sapiro, and R. Vidal, "See all by looking at a few: Sparse modeling for finding representative objects," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012. 1

[3] B. Gong, W. Chao, K. Grauman, and F. Sha, "Diverse sequential subset selection for supervised video summarization," in *Neural Information Processing Systems*, 2014. 1

[4] I. Simon, N. Snavely, and S. M. Seitz, "Scene summarization for online image collections," in *IEEE International Conference on Computer Vision*, 2007. 1

[5] E. Elhamifar, G. Sapiro, and S. S. Sastry, "Dissimilarity-based sparse subset selection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016. 1, 2, 3, 6

[6] E. Elhamifar, G. Sapiro, and R. Vidal, "Finding exemplars from pairwise dissimilarities via simultaneous sparse recovery," *Neural Information Processing Systems*, 2012. 1

[7] G. Kim, E. Xing, L. Fei-Fei, and T. Kanade, "Distributed cosegmentation via submodular optimization on anisotropic diffusion," in *International Conference on Computer Vision*, 2011. 1

[8] A. Shah and Z. Ghahramani, "Determinantal clustering process – a nonparametric bayesian approach to kernel based semi-supervised clustering," in *Conference on Uncertainty in Artificial Intelligence*, 2013. 1

[9] R. Reichart and A. Korhonen, "Improved lexical acquisition through dpp-based verb clustering," in *Conference of the Association for Computational Linguistics*, 2013. 1

[10] E. Elhamifar, S. Burden, and S. S. Sastry, "Adaptive piecewise-affine inverse modeling of hybrid dynamical systems," in *World Congress of the International Federation of Automatic Control (IFAC)*, 2014. 1

[11] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research*, 2003. 1

[12] I. Misra, A. Shrivastava, and M. Hebert, "Data-driven exemplar model selection," in *Winter Conference on Applications of Computer Vision*, 2014. 1

[13] H. Lin and J. Bilmes, "Learning mixtures of submodular shells with application to document summarization," in *Conference on Uncertainty in Artificial Intelligence*, 2012. 1

[14] A. Kulesza and B. Taskar, "Determinantal point processes for machine learning," *Foundations and Trends in Machine Learning*, vol. 5, 2012. 1

[15] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, 2007. 1

[16] A. Krause, H. B. McMahan, C. Guestrin, and A. Gupta, "Robust submodular observation selection," *Journal of Machine Learning Research*, vol. 9, 2008. 1, 2

[17] S. Joshi and S. Boyd, "Sensor selection via convex optimization," *IEEE Transactions on Signal Processing*, vol. 57, 2009. 1

[18] J. Hartline, V. S. Mirrokni, and M. Sundararajan, "Optimal marketing strategies over social networks," in *World Wide Web Conference*, 2008. 1

[19] D. McSherry, "Diversity-conscious retrieval," in *Advances in Case-Based Reasoning*, 2002. 1

[20] R. Duda, P. Hart, and D. Stork, *Pattern Classification*. Wiley-Interscience, October 2004. 1

[21] M. Aharon, M. Elad, and A. M. Bruckstein, "K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006. 1

[22] L. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, 1989. 1

[23] F. Hadlock, "Finding a maximum cut of a planar graph in polynomial time," *SIAM Journal on Computing*, vol. 4, 1975. 1

[24] R. Motwani and P. Raghavan, "Randomized algorithms," *Cambridge University Press, New York*, 1995. 1

[25] J. Carbonell and J. Goldstein, "The use of mmr, diversity-based reranking for reordering documents and producing summaries," in *SIGIR*, 1998. 1

[26] G. Cornuejols, M. Fisher, and G. Nemhauser, "Exceptional paper-location of bank accounts to optimize float: An analytic study of exact and approximate algorithms," *Journal of Computer and System Sciences*, vol. 23, 1977. 1

[27] M. Charikar, S. Guha, E. Tardos, and D. B. Shmoys, "A constant-factor approximation algorithm for the k-median problem," *Journal of Computer and System Sciences*, vol. 65, 2002. 1

[28] E. Esser, M. Moller, S. Osher, G. Sapiro, and J. Xin, "A convex model for non-negative matrix factorization and dimensionality reduction on physical space," *IEEE Transactions on Image Processing*, vol. 21, no. 7, pp. 3239–3252, 2012. 1

[29] A. Borodin and G. Olshanski, "Distributions on partitions, point processes, and the hypergeometric kernel," *Communications in Mathematical Physics*, vol. 211, 2000. 1

[30] E. Amaldi and V. Kann, "On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems," *Theoretical Computer Science*, vol. 209, pp. 237–260, 1998. 1

[31] O. Macchi, "The coincidence approach to stochastic point processes," *Advances in Applied Probability*, vol. 7, 1995. 1

[32] P. Awasthi, A. S. Bandeira, M. Charikar, R. Krishnaswamy, S. Villar, and R. Ward, "Relax, no need to round: Integrality of clustering formulations," in *Conference on Innovations in Theoretical Computer Science (ITCS)*, 2015. 1

[33] A. Nellore and R. Ward, "Recovery guarantees for exemplar-based clustering," in *Information and Computation*, 2015. 1

[34] M. Gygli, Y. Song, and L. Cao, "Video2gif: Automatic generation of animated gifs from video," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 2

[35] H. Lin, J. Bilmes, and S. Xie, "Graph-based submodular selection for extractive summarization," *IEEE Automatic Speech Recognition and Understanding (ASRU)*, 2009. 2

[36] N. Buchbinder, M. Feldman, J. Naor, and R. Schwartz, "A tight linear time (1/2)-approximation for unconstrained submodular maximization," *Annual Symposium on Foundations of Computer Science*, 2012. 4

[37] M. Gygli, H. Grabner, H. Riemenschneider, and L. V. Gool, "Creating summaries from user videos," in *European Conference on Computer Vision*, 2014. 5, 6, 7

[38] Y. Song, J. Vallmitjana, A. Stent, and A. Jaimes, "Tvsum: Summarizing web videos using titles," *CVPR*, 2015. 6, 7

[39] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *International Conference on Computer Vision*, 2015. 6

[40] B. W. Matthews, "Comparison of the predicted and observed secondary structure of t4 phage lysozyme," *Biochimica et Biophysica Acta (BBA)-Protein Structure*, vol. 405, no. 2, pp. 442–451, 1975. 7