Summarizing Documents Using Submodular Functions

Pritish Chakraborty - 21Q050011,
Harsh Diwakar - 213070018

Introduction

- Paper [Lin and Bilmes, 2011] explores methods for document summarization.
  - What are the criteria for a good summary? Diversity, coverage...
  - How is the summary evaluated? ROUGE
- Authors demonstrate how summarization can be treated as a submodular maximization problem.
  - Controllable hyperparameters for a more robust summary.
  - Extractive method for query-based document summarization.
- We run experiments on the WikiHow dataset using the authors' submodular function for different hyperparameters.
- We attempt to learn a deep submodular function [Dolhansky and Bilmes, 2016] for a variant of given document summarization task as well.

Formalization

Idea

The idea is to select a subset of documents S from a ground set V such that a monotone submodular information measure f is maximized. Formally, we have that -:

$$\max_{S \in 2^V} f(S)$$

What Are We Looking For?

- **Relevance**: Extracted summary must be relevant to the document it was extracted from.
- **Non-redundancy**: The summary should be as *diverse* as possible, and not cover concepts already covered before.
- Earlier formulations compute both objectives separately, and mix the two by encouraging relevance and penalizing redundancy. Paper rewards diversity instead.

Submodularity

- Submodular set functions $f : 2^V \mapsto \mathbb{R}$ follow the property of **diminishing gains**. That is, $\forall A \subseteq B \subseteq V$ and e $\in V$, $e \notin B$, f(A $\cup \{e\}) - f(A) \geq f(B \cup \{e\}) - f(B)$.
- The maximization of such functions has been found to be NP-hard. As such, we need to look for an approximate solution.
- **Nemhauser's result**: If the objective function is *monotone* and *submodular*, we are guaranteed atleast 63% of the optimal solution if we use a greedy algorithm. Many variants of this greedy algorithm exist, such as stochastic [Badanidiyuru et al., 2014] for non-monotone case and distorted [Harshaw et al., 2019] for γ-weakly submodular case.
- Examples of submodular functions include the graph cut and facility location.

Evaluating Summaries

- Authors use the well-known ROUGE metric for document summarization.
- ROUGE takes the form below, where $\mathcal{S}$ is the set of human-generated reference summaries and $\mathcal{W}$ is the set of features. If the bag of words model is followed, $c_w(A)$ is the frequency of w in summary A.

$$r_{\mathcal{S}}(A) = \frac{\sum_{w \in \mathcal{W}} \sum_{s \in \mathcal{S}} \min(c_w(A), c_w(s))}{\sum_{w \in \mathcal{W}} \sum_{s \in \mathcal{S}} c_w(s)}$$

  - Regarding word models, others such as TF-IDF and CBOW are also used in practice.
  - For ML, in our experience, pre-trained word embeddings were a better solution.
- ROUGE has been shown to be submodular.
- In their experiments, authors use TF-IDF with similarity function based on cosine similarity of TF-IDF vectors.

Document Summarization with submodular Function

- A two part objective function for document summarization:

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

  λ is a trade-off coefficient

- $\mathcal{L}(S)$ is coverage function and can have a form:

$$\mathcal{L}(S) = \sum_{i \in V} \min\{C_i(S), \alpha C_i(V)\}$$

  $C_i : 2^V \to \mathcal{R}$ is a monotone function and $0 \leq \alpha \leq 1$ is a threshold coefficient

- $\mathcal{R}(S)$ is a diversity function and have a form:

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

  $P_i, i = 1, \ldots K$ is partition of the ground set V and $r_i \geq 0$ is a reward of adding i into the empty set.

Coverage Function

- A set function that measures the similarity of summary set S with the document set

$$\mathcal{L}(S) = \sum_{i \in V} \min\{C_i(S), \alpha C_i(V)\}$$

- C_i measures how similar S is to element i or, how much of i is covered by S
- A simple way to define $C_i(S)$ is

$$C_i(S) = \sum_j w_{ij}$$

  $w_{ij} \geq 0$ measures the similarity between sentences i and j
- Hence, combined Coverage Function becomes:

$$\mathcal{L}(S) = \sum_{i \in V} \min \left\{ \sum_{j \in S} w_{ij}, \alpha \sum_{k \in V} w_{ik} \right\}$$

Diversity Function

- A set function that rewards diversity in set S

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

- $P_i, i = 1, \dots K$ is partition of the ground set V into separate clusters
  - $\cap_i P_i = V$
  - P_is are disjoint
  - Found using k-means Clustering [Likas et al., 2003]
- r_i is average similarity of sentence i to the rest of the document
  - $r_i = \frac{1}{N} \sum_j w_{ij}$
  - N is the total number of sentences.
- Hence, overall Diversity Function becomes:

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

Deep Submodular Functions i

- Family of submodular functions that are a strict generalization of many known submodular functions, such as SCMMs and feature-based functions.
- Outperforms feature-based functions due to configurable depth and interaction between layers.
- Structure very similar to deep neural networks (DNN); thus based on a known paradigm.
- Modern DNNs can be made to behave like DSFs if weights are constrained to be non-negative.
  - DSFs also allow for skip connections and other DNN mechanics.
  - Popular activation funcs like sigmoid and tanh are concave in non-negative domain - exploit concave over submodular.

Deep Submodular Functions ii

- Example DSF -:

$$f(A) = \hat{\sigma}(\sum_{u \in \mathcal{U}} w_u \sqrt{m_u(A)})$$

- Trained using loss augmented inference.

$$\min_{w \geq 0} \sum_{S \in \mathcal{S}} \left(\max_{A \in 2^V} [f(A) + l_S(A)] - f(S) \right)_+ + \frac{\lambda}{2} ||w||_2^2$$

  - Aim is to maximize loss margin between human summary and best scoring candidate summary.
  - In our experiments we choose multiple candidate summaries as slight variations of human summary.

Deep Submodular Functions iii

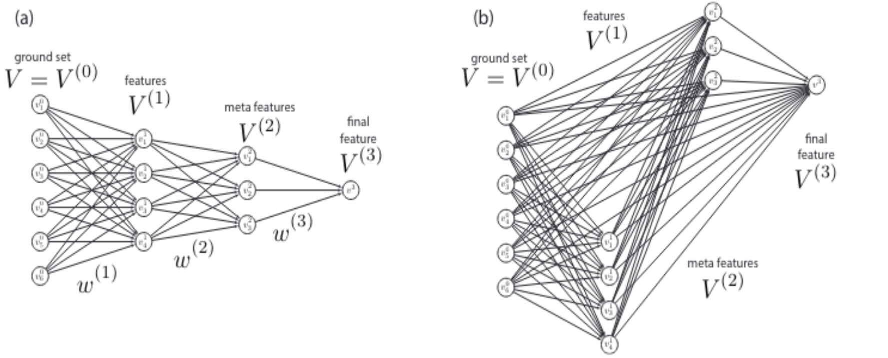


Figure 1: (a) 3-layered DSF; (b) DSF with skip connections [Dolhansky and Bilmes, 2016]

Experiments and Results i

- WikiHow Dataset
  - Consists of several documents demarcated by human-provided summaries
  - We used 1000 of the articles for evaluating the performance
- Pre-processing
  - Segmenting the Sentences
  - Stemming by Porter Stemmer
  - Calculate term frequency- inverse document frequency (TF-IDF) for every sentence
  - Use TF-IDF to find the cosine similarity for w_{ij}
- Implementation of trade-off between Coverage and diversity functions:

$$\mathcal{F}(S) = \lambda\mathcal{L}(S) + (1-\lambda)\mathcal{R}(S)$$

Experiments and Results ii

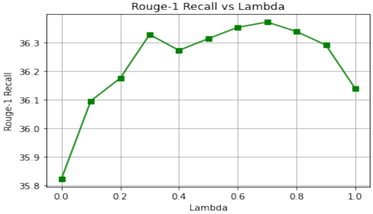
- Parameters
  - Total Number of Clusters $K = 0.2 * N$, N is total number of sentences
  - α in Eq. 7 is $\alpha = a/N$, $a = 6$
- Results with different applied function for summarization

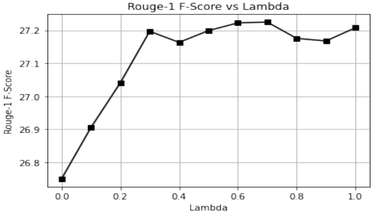
Experiment	ROUGE-1 Recall	ROUGE-1 F1- Score	ROUGE-L Recall	ROUGE-L F-1 Score
$\mathcal{L}(S)$	36.14	27.20	21.83	16.24
$\mathcal{R}(S)$	35.82	26.75	21.72	16.05
$0.15\mathcal{L}(S) + 0.85\mathcal{R}(S)$	34.67	27.67	20.81	16.41

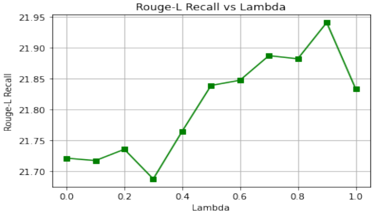
- Only coverage function is giving higher recall however, the F1-Score is increased by using the combination of both the functions.

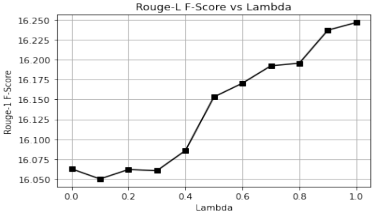
Experiments and Results iii

- Experiment with different values of Lambda









DSF Experiments

- Used a variant of example DSF which does not have modular function.
- Single-layered DSF with a normalized sigmoid at the end to impose concave wrapper.
- Train and test split on the WikiHow dataset.
  - Attempt to use TF-IDF vectorization but matrix too large to fit in memory.
  - Shifted to pre-trained, fixed-size word embeddings from the GloVe embedding repository.
- Loss-augmented inference with regularizing parameter implemented.
  - Choice of loss: Leaky ReLU.
  - Ran into issue of constant loss.
- Candidate summaries generated as variations of desirable human summaries.

References i

Badanidiyuru, A., Mirzasoleiman, B., Karbasi, A., and Krause, A. (2014).
Streaming submodular maximization: Massive data summarization on the fly.
In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 671–680.

Dolhansky, B. W. and Bilmes, J. A. (2016).
Deep submodular functions: Definitions and learning.
Advances in Neural Information Processing Systems, 29.

References ii

Harshaw, C., Feldman, M., Ward, J., and Karbasi, A. (2019).
Submodular maximization beyond non-negativity: Guarantees, fast algorithms, and applications.
In *International Conference on Machine Learning*, pages 2634–2643. PMLR.

Likas, A., Vlassis, N., and Verbeek, J. J. (2003).
The global k-means clustering algorithm.
Pattern recognition, 36(2):451–461.

Lin, H. and Bilmes, J. (2011).
A class of submodular functions for document summarization.
In *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies*, pages 510–520.