A Class of Submodular Functions for Document Summarization

Hui Lin, Jeff Bilmes

University of Washington, Seattle
Dept. of Electrical Engineering

June 20, 2011

Extractive Document Summarization

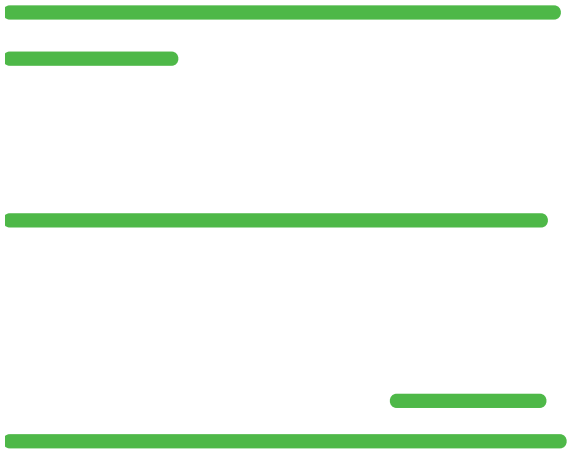
- The figure below represents the sentences of a document

Extractive Document Summarization

- We extract sentences (green) as a summary of the full document

Extractive Document Summarization

- We extract sentences (green) as a summary of the full document



Extractive Document Summarization

- We extract sentences (green) as a summary of the full document

- The summary on the left is a subset of the summary on the right.

Extractive Document Summarization

- We extract sentences (green) as a summary of the full document

- The summary on the left is a subset of the summary on the right.
- Consider adding a new (blue) sentence to each of the two summaries.

Extractive Document Summarization

- We extract sentences (green) as a summary of the full document

- The summary on the left is a subset of the summary on the right.
- Consider adding a new (blue) sentence to each of the two summaries.
- The marginal (incremental) benefit of adding the new (blue) sentence to the smaller (left) summary is no more than the marginal benefit of adding the new sentence to the larger (right) summary.

Extractive Document Summarization

- We extract sentences (green) as a summary of the full document

- The summary on the left is a subset of the summary on the right.
- Consider adding a new (blue) sentence to each of the two summaries.
- The marginal (incremental) benefit of adding the new (blue) sentence to the smaller (left) summary is no more than the marginal benefit of adding the new sentence to the larger (right) summary.
- **diminishing returns** $\leftrightarrow$ **submodularity**

Outline

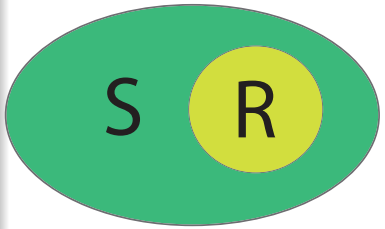
Submodular Set Functions

- There is a finite sized "ground set" of elements V
- We use set functions of the form $f : 2^V \rightarrow \mathbb{R}$
- A set function f is monotone nondecreasing if $\forall R \subseteq S$, $f(R) \leq f(S)$.

Definition of Submodular Functions

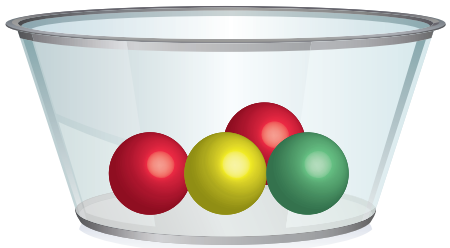
For any $R \subseteq S \subseteq V$ and $k \in V, k \notin S$, $f(\cdot)$ is **submodular** if

$$f(S + k) - f(S) \leq f(R + k) - f(R)$$

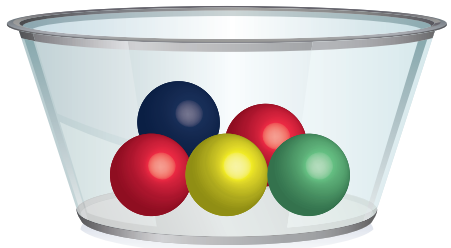


This is known as the principle of **diminishing returns**

Example: Number of Colors of Balls in Urns



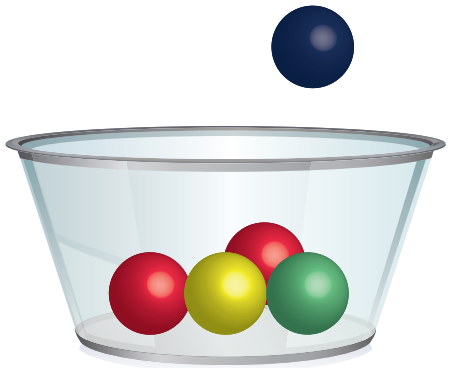
$f(R) = f($  $) = 3$



$f(S) = f($  $) = 4$

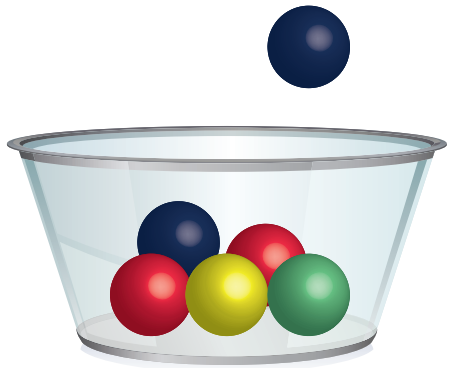
- Given a set A of colored balls
- $f(A)$: the number of distinct colors contained in the urn
- The incremental value of an object only **diminishes** in a **larger** context (diminishing returns).

Example: Number of Colors of Balls in Urns



$f(R) = f(\quad) = 3$

$f(R + k) = f(\quad + \quad) = 4$



$f(S) = f(\quad) = 4$

$f(S + k) = f(\quad + \quad) = 4$

- Given a set A of colored balls
- $f(A)$: the number of distinct colors contained in the urn
- The incremental value of an object only **diminishes** in a **larger** context (diminishing returns).

Why is submodularity attractive?

Why is submodularity attractive?

Why is convexity attractive?

How about submodularity:

Why is submodularity attractive?

Why is convexity attractive?

- convexity appears in many mathematical models in economy, engineering and other sciences.
- minimum can be found efficiently.
- convexity has many nice properties, e.g. convexity is preserved under many natural operations and transformations.

How about submodularity:

Why is submodularity attractive?

Why is convexity attractive?

- convexity appears in many mathematical models in economy, engineering and other sciences.
- minimum can be found efficiently.
- convexity has many nice properties, e.g. convexity is preserved under many natural operations and transformations.

How about submodularity:

- submodularity arises in many areas: combinatorics, economics, game theory, operation research, machine learning, and (now) natural language processing.
- minimum can be found in polynomial time
- submodularity has many nice properties, e.g. submodularity is preserved under many natural operations and transformations (e.g. scaling, addition, convolution, etc.)

Outline

Problem setup

- The ground set V corresponds to all the sentences in a document.
- Extractive document summarization: select a small subset $S \subseteq V$ that accurately represents the entirety (ground set V).

Problem setup

- The ground set V corresponds to all the sentences in a document.
- Extractive document summarization: select a small subset $S \subseteq V$ that accurately represents the entirety (ground set V).
- The summary is usually required to be length-limited.
  - c_i: cost (e.g., the number of words in sentence i),
  - b: the budget (e.g., the largest length allowed),
  - knapsack constraint: $\sum_{i \in S} c_i \leq b$.

Problem setup

- The ground set V corresponds to all the sentences in a document.
- Extractive document summarization: select a small subset $S \subseteq V$ that accurately represents the entirety (ground set V).
- The summary is usually required to be length-limited.
  - c_i: cost (e.g., the number of words in sentence i),
  - b: the budget (e.g., the largest length allowed),
  - knapsack constraint: $\sum_{i \in S} c_i \leq b$.
- A set function $f : 2^V \rightarrow \mathbb{R}$ measures the quality of the summary S,
- Thus, the summarization problem is formalized as:

Problem (Document Summarization Optimization Problem)

$$S^* \in \operatorname*{argmax}_{S \subseteq V} f(S) \text{ subject to: } \sum_{i \in S} c_i \leq b. \quad (1)$$

A Practical Algorithm for Large-Scale Summarization

When f is both **monotone** and **submodular**:

- A greedy algorithm with partial enumeration (Sviridenko, 2004), theoretical guarantee of near-optimal solution, but not practical for large data sets.

A Practical Algorithm for Large-Scale Summarization

When f is both **monotone** and **submodular**:

- A greedy algorithm with partial enumeration (Sviridenko, 2004), theoretical guarantee of near-optimal solution, but not practical for large data sets.
- A greedy algorithm (Lin and Bilmes, 2010): near-optimal with theoretical guarantee, and practical/scalable!
  - We choose next element with largest ratio of gain over **scaled** cost:

$$k \leftarrow \operatorname*{argmax}_{i \in U} \frac{f(G \cup \{i\}) - f(G)}{(c_i)^r}. \tag{2}$$

A Practical Algorithm for Large-Scale Summarization

When f is both **monotone** and **submodular**:

- A greedy algorithm with partial enumeration (Sviridenko, 2004), theoretical guarantee of near-optimal solution, but not practical for large data sets.
- A greedy algorithm (Lin and Bilmes, 2010): near-optimal with theoretical guarantee, and practical/scalable!
  - We choose next element with largest ratio of gain over **scaled** cost:

$$k \leftarrow \operatorname*{argmax}_{i \in U} \frac{f(G \cup \{i\}) - f(G)}{(c_i)^r}. \tag{2}$$

  - Scalability: the argmax above can be solved by $O(\log n)$ calls of f, thanks to submodularity
  - Integer linear programming (ILP) takes 17 hours vs. greedy which takes < 1 second!!

Objective Function Optimization: Performance in Practice

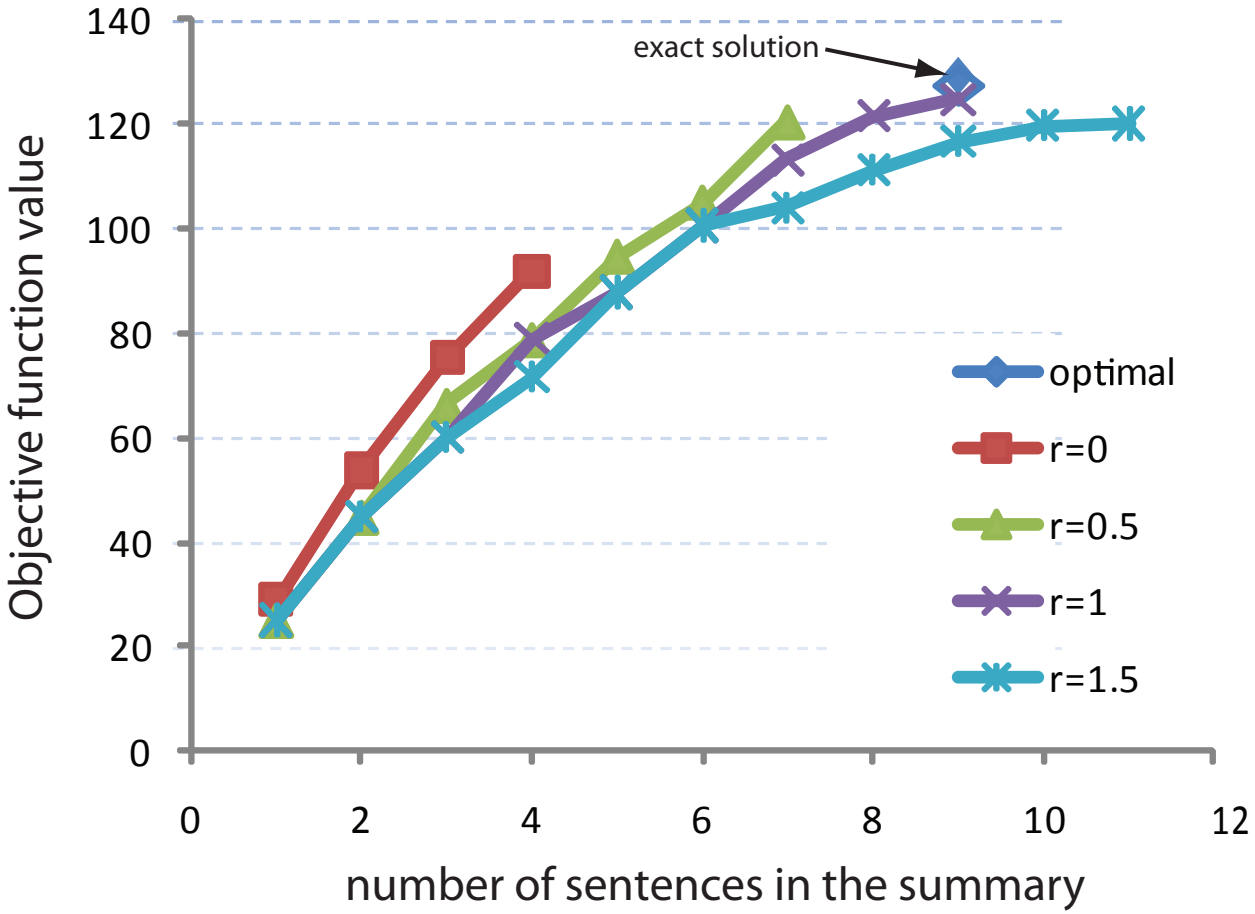


Figure: The plots show the achieved objective function value as the number of selected sentences grows. The plots stop when in each case adding more sentences violates the budget.

Outline

MMR is non-monotone submodular

Maximal Margin Relevance (MMR, Carbonell and Goldstein, 1998):

- MMR is very popular in document summarization.
- MMR corresponds to an objective function which is submodular but non-monotone (see paper for details).
- Therefore, the greedy algorithm's performance guarantee does not apply in this case (since MMR is not monotone).

MMR is non-monotone submodular

Maximal Margin Relevance (MMR, Carbonell and Goldstein, 1998):

- MMR is very popular in document summarization.
- MMR corresponds to an objective function which is submodular but non-monotone (see paper for details).
- Therefore, the greedy algorithm's performance guarantee does not apply in this case (since MMR is not monotone).
- Moreover, the greedy algorithm of MMR does **not** take cost into account, and therefore could lead to solutions that are significantly worse than the solutions found by the greedy algorithm with scaled cost.

MMR is non-monotone submodular

Maximal Margin Relevance (MMR, Carbonell and Goldstein, 1998):

- MMR is very popular in document summarization.
- MMR corresponds to an objective function which is submodular but non-monotone (see paper for details).
- Therefore, the greedy algorithm's performance guarantee does not apply in this case (since MMR is not monotone).
- Moreover, the greedy algorithm of MMR does **not** take cost into account, and therefore could lead to solutions that are significantly worse than the solutions found by the greedy algorithm with scaled cost.

MMR-like approaches:

- non-monotone because summary redundancy is penalized **negatively**.

Concept-based approach

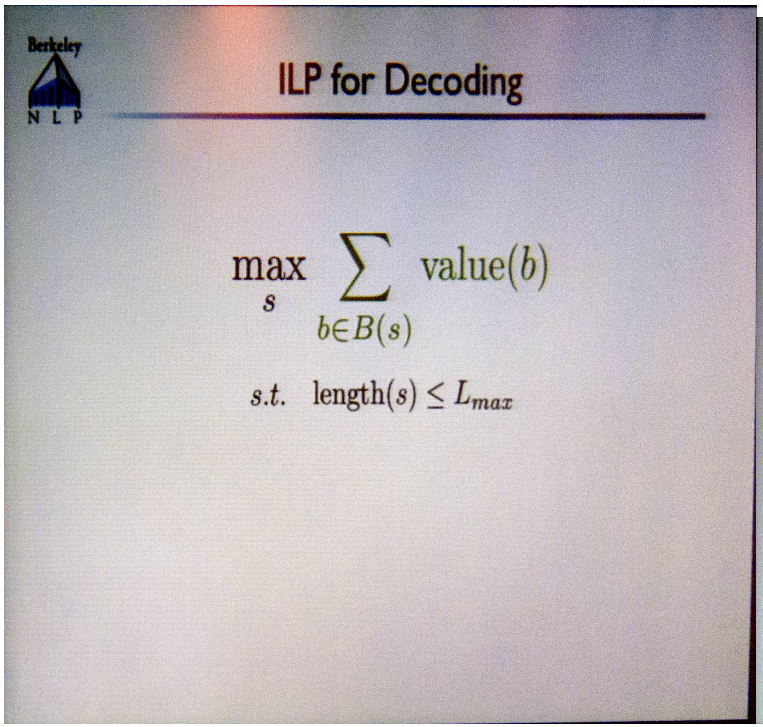
- Concepts: n-grams, keywords, etc.
- Maximizes the weighted credit of concepts covered the summary

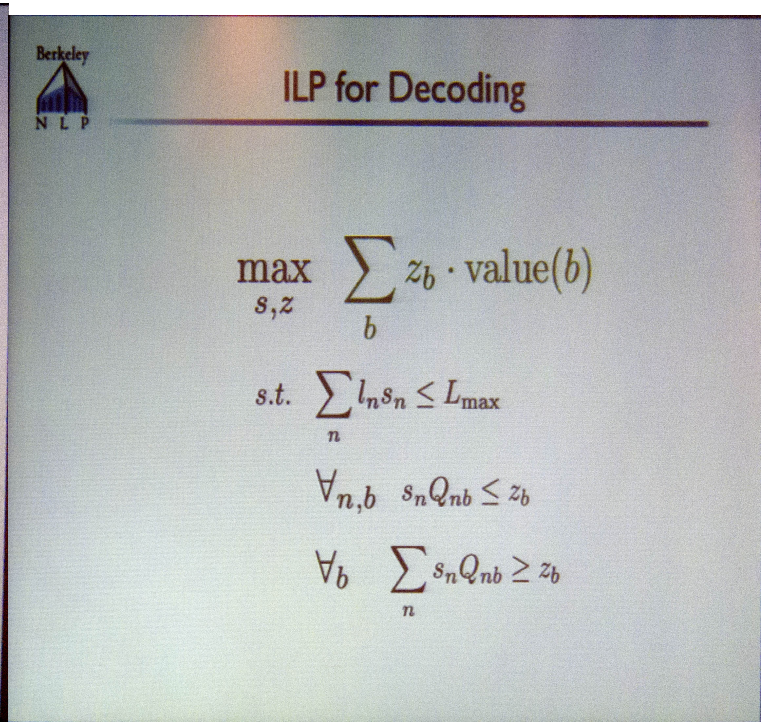
Concept-based approach

- Concepts: n-grams, keywords, etc.
- Maximizes the weighted credit of concepts covered the summary: submodular! (similar to the colored ball examples we saw)
- The objectives in the nice talk (Berg-Kirkpatrick et al., 2011) we saw at the beginning of this section are, actually, submodular ☺ when value(b) ≥ 0.

Concept-based approach

- Concepts: n-grams, keywords, etc.
- Maximizes the weighted credit of concepts covered the summary: submodular! (similar to the colored ball examples we saw)
- The objectives in the nice talk (Berg-Kirkpatrick et al., 2011) we saw at the beginning of this section are, actually, submodular ☺ when value(b) ≥ 0.





Even ROUGE-N itself is monotone submodular!!

ROUGE-N: high correlation to human evaluation (Lin 2004).

Even ROUGE-N itself is monotone submodular!!

ROUGE-N: high correlation to human evaluation (Lin 2004).

Theorem (Lin and Bilmes, 2011)

ROUGE-N is monotone submodular (proof in paper).

Even ROUGE-N itself is monotone submodular!!

ROUGE-N: high correlation to human evaluation (Lin 2004).

Theorem (Lin and Bilmes, 2011)

ROUGE-N is monotone submodular (proof in paper).

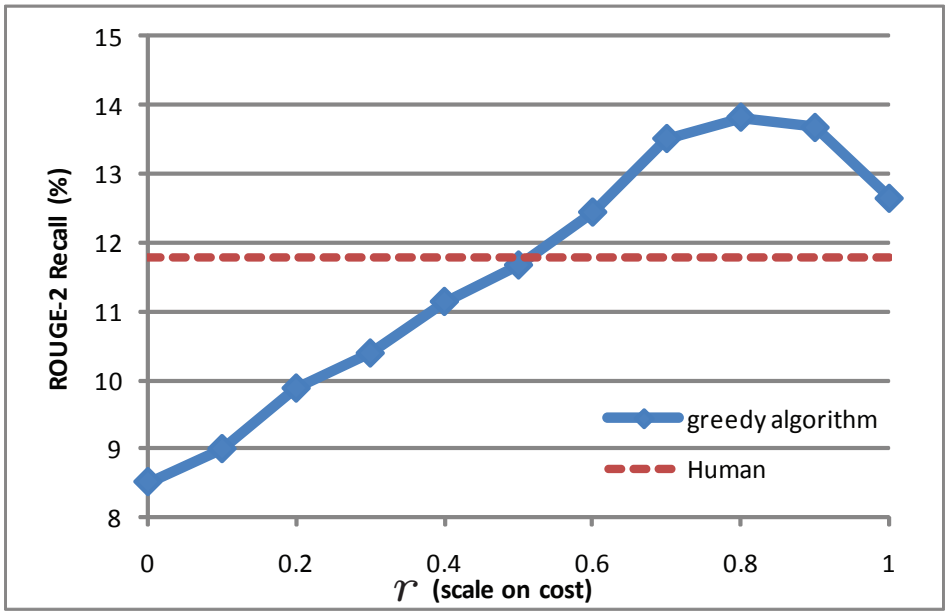


Figure: Oracle experiments on DUC-05. The red dash line indicates the best ROUGE-2 recall score of human summaries (summary with ID C).

Outline

The General Form of Our Submodular Functions

- Two properties of a good summary: relevance and non-redundancy.
- Common approaches (e.g., MMR): encourage relevance and (negatively) penalize redundancy.
- The redundancy penalty is usually what violates monotonicity.
- Our approach: we positively **reward diversity** instead of negatively penalizing redundancy:

The General Form of Our Submodular Functions

- Two properties of a good summary: relevance and non-redundancy.
- Common approaches (e.g., MMR): encourage relevance and (negatively) penalize redundancy.
- The redundancy penalty is usually what violates monotonicity.
- Our approach: we positively **reward diversity** instead of negatively penalizing redundancy:

Definition (The general form of our submodular functions)

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

- $\mathcal{L}(S)$ measures the coverage (or fidelity) of summary set S to the document.
- $\mathcal{R}(S)$ rewards diversity in S.
- $\lambda \geq 0$ is a trade-off coefficient.
- Analogous to the objectives widely used in machine learning: loss + regularization

Coverage function

Coverage Function

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

- $\mathcal{C}_i : 2^V \rightarrow \mathbb{R}$ is monotone submodular, and measures how well i is covered by S.
- $0 \leq \alpha \leq 1$ is a threshold coefficient — sufficient coverage fraction.

Coverage function

Coverage Function

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

- $\mathcal{C}_i : 2^V \rightarrow \mathbb{R}$ is monotone submodular, and measures how well i is covered by S.
- $0 \leq \alpha \leq 1$ is a threshold coefficient — sufficient coverage fraction.
- if $\min\{\mathcal{C}_i(S), \alpha\mathcal{C}_i(V)\} = \alpha\mathcal{C}_i(V)$, then sentence i is well covered by summary S (saturated).

Coverage function

Coverage Function

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

- $\mathcal{C}_i : 2^V \rightarrow \mathbb{R}$ is monotone submodular, and measures how well i is covered by S.
- $0 \leq \alpha \leq 1$ is a threshold coefficient — sufficient coverage fraction.
- if $\min\{\mathcal{C}_i(S), \alpha\mathcal{C}_i(V)\} = \alpha\mathcal{C}_i(V)$, then sentence i is well covered by summary S (saturated).
- After saturation, further increases in $\mathcal{C}_i(S)$ won't increase the objective function values (return diminishes).
- Therefore, new sentence added to S should focus on sentences that are not yet saturated, in order to increasing the objective function value.

Coverage function

Coverage Function

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

- $\mathcal{C}_i$ measures how well i is covered by S.
- One simple possible $\mathcal{C}_i$ (that we use in our experiments and works well) is:

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

  where $w_{i,j} \geq 0$ measures the similarity between i and j.
- With this C_i, $\mathcal{L}(S)$ is monotone submodular, as required.

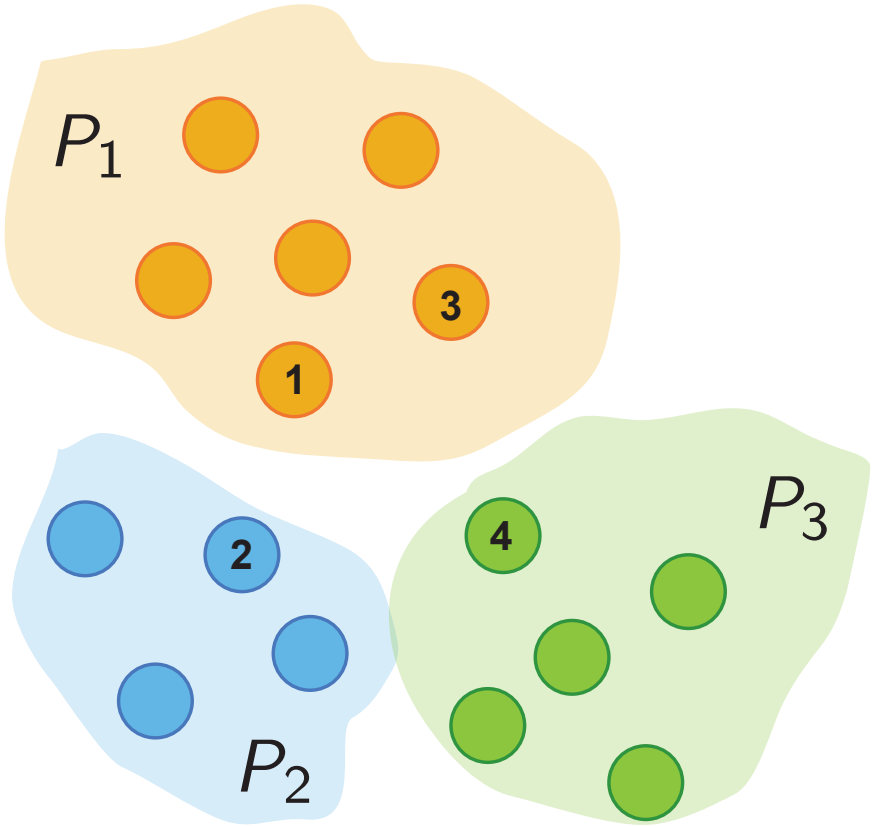
Diversity reward function

Diversity Reward Function

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

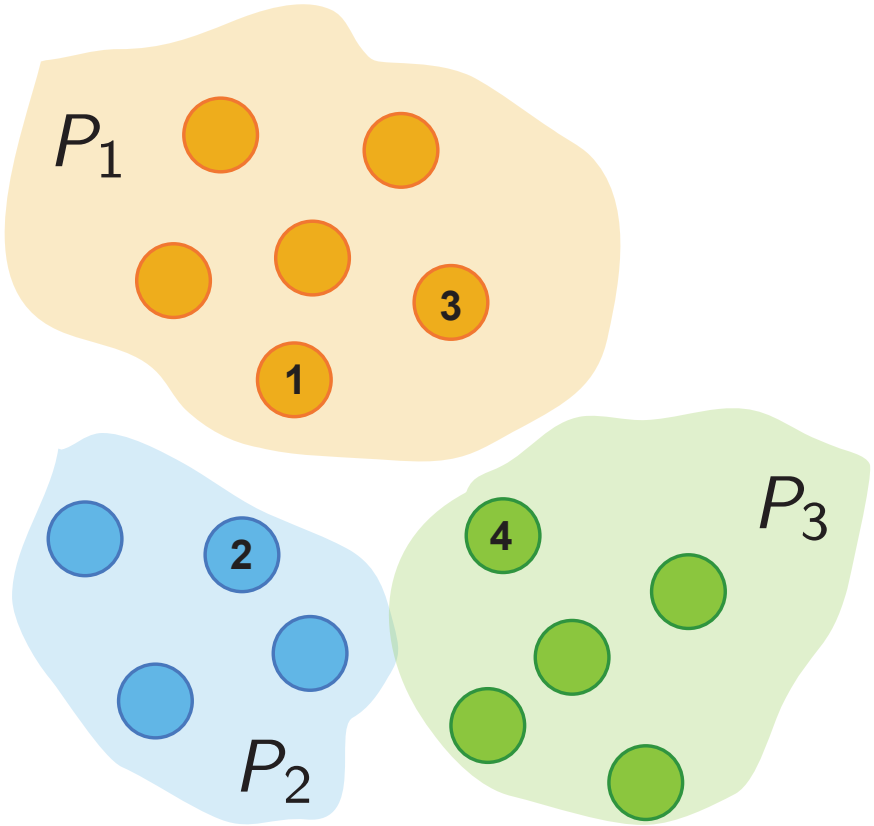
- $P_i, i = 1, \cdots K$ is a partition of the ground set V
- $r_j \geq 0$: **singleton reward** of j, which represents the importance of j to the summary.
- square root over the sum of rewards of sentences belong to the same partition (diminishing returns).
- $\mathcal{R}(S)$ is monotone submodular as well.

Diversity reward function - how does it reward diversity?



- 3 partitions: P_1, P_2, P_3.
- Singleton reward for sentence $1, 2, 3$ and 4: $r_1 = 5, r_2 = 5, r_3 = 4, r_4 = 3$.
- Current summary: $S = \{1, 2\}$
- consider adding a new sentence, 3 or 4.
- A diverse (non-redundant) summary: $\{1, 2, 4\}$.

Diversity reward function - how does it reward diversity?



- 3 partitions: P_1, P_2, P_3.
- Singleton reward for sentence $1, 2, 3$ and 4: $r_1 = 5, r_2 = 5, r_3 = 4, r_4 = 3$.
- Current summary: $S = \{1, 2\}$
- consider adding a new sentence, 3 or 4.
- A diverse (non-redundant) summary: $\{1, 2, 4\}$.

- Modular objective:
  $\mathcal{R}(\{1, 2, 3\}) = 5 + 5 + 4 = 14 > \mathcal{R}(\{1, 2, 4\}) = 5 + 5 + 3 = 13$
- Submodular objective:
  $\mathcal{R}(\{1, 2, 3\}) = 5 + \sqrt{5 + 4} = 8 < \mathcal{R}(\{1, 2, 4\}) = 5 + 5 + 3 = 13$

Diversity Reward Function

- **singleton reward** of j: the importance of being j (to the summary).
  - Query-independent (generic) case:

$$r_j = \frac{1}{N} \sum_{i \in V} w_{i,j}.$$

  - Query-dependent case, given a query Q,

$$r_j = \beta \frac{1}{N} \sum_{i \in V} w_{i,j} + (1 - \beta) r_{j,Q}$$

    where $r_{j,Q}$ measures the relevance between j and query Q.

Diversity Reward Function

- **singleton reward** of j: the importance of being j (to the summary).
  - Query-independent (generic) case:

$$r_j = \frac{1}{N} \sum_{i \in V} w_{i,j}.$$

  - Query-dependent case, given a query Q,

$$r_j = \beta \frac{1}{N} \sum_{i \in V} w_{i,j} + (1 - \beta) r_{j,Q}$$

    where $r_{j,Q}$ measures the relevance between j and query Q.

Multi-resolution Diversity Reward

$$\mathcal{R}(S) = \sum_{i=1}^{K_1} \sqrt{\sum_{j \in P_i^{(1)} \cap S} r_j} + \sum_{i=1}^{K_2} \sqrt{\sum_{j \in P_i^{(2)} \cap S} r_j} + \cdots$$

Outline

Generic Summarization

- DUC-04: generic summarization

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

DUC-04	R	F
$\mathcal{L}_1(S)$	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

Generic Summarization

- DUC-04: generic summarization

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

DUC-04	R	F
$\mathcal{L}_1(S)$	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

- Note: this is the best ROUGE-1 result ever reported on DUC-04.

Query-focused Summarization

- DUC-05,06,07: query-focused summarization
- For each document cluster, a title and a narrative (query) describing a user's information need are provided.
- Nelder-Mead (derivative-free) for parameter training.

DUC-05 results

Table: ROUGE-2 recall (R) and F-measure (F) results (%)

	R	F
$\mathcal{L}_1(S) + \lambda \mathcal{R}_Q(S)$	7.82	7.72
$\mathcal{L}_1(S) + \sum_{\kappa=1}^{3} \lambda_\kappa \mathcal{R}_{Q,\kappa}(S)$	**8.19**	**8.13**
Daumé III and Marcu (2006)	6.98	-
Wei et al. (2010)	8.02	-
Best system in DUC-05 (peer 15)	7.44	7.43

- DUC-06 was used as training set for the objective function with single diversity reward.
- DUC-06 and 07 were used as training sets for the objective function with multi-resolution diversity reward (new results since our camera-ready version of the paper)

DUC-05 results

Table: ROUGE-2 recall (R) and F-measure (F) results (%)

	R	F
$\mathcal{L}_1(S) + \lambda \mathcal{R}_Q(S)$	7.82	7.72
$\mathcal{L}_1(S) + \sum_{\kappa=1}^{3} \lambda_\kappa \mathcal{R}_{Q,\kappa}(S)$	**8.19**	**8.13**
Daumé III and Marcu (2006)	6.98	-
Wei et al. (2010)	8.02	-
Best system in DUC-05 (peer 15)	7.44	7.43

- DUC-06 was used as training set for the objective function with single diversity reward.
- DUC-06 and 07 were used as training sets for the objective function with multi-resolution diversity reward (new results since our camera-ready version of the paper)
- Note: this is the best ROUGE-2 result ever reported on DUC-05.

DUC-06 results

Table: ROUGE-2 recall (R) and F-measure (F) results (%)

	R	F
$\mathcal{L}_1(S) + \lambda \mathcal{R}_Q(S)$	9.75	9.77
$\mathcal{L}_1(S) + \sum_{\kappa=1}^{3} \lambda_\kappa \mathcal{R}_{Q,\kappa}(S)$	**9.81**	**9.82**
Celikyilmaz and Hakkani-tür (2010)	9.10	-
Shen and Li (2010)	9.30	-
Best system in DUC-06 (peer 24)	9.51	9.51

- DUC-05 was used as training set for the objective function with single diversity reward.
- DUC-05 and 07 were used as training sets for the objective function with multi-resolution diversity reward (new results since our camera-ready version of the paper)

DUC-06 results

Table: ROUGE-2 recall (R) and F-measure (F) results (%)

	R	F
$\mathcal{L}_1(S) + \lambda \mathcal{R}_Q(S)$	9.75	9.77
$\mathcal{L}_1(S) + \sum_{\kappa=1}^{3} \lambda_\kappa \mathcal{R}_{Q,\kappa}(S)$	**9.81**	**9.82**
Celikyilmaz and Hakkani-tür (2010)	9.10	-
Shen and Li (2010)	9.30	-
Best system in DUC-06 (peer 24)	9.51	9.51

- DUC-05 was used as training set for the objective function with single diversity reward.
- DUC-05 and 07 were used as training sets for the objective function with multi-resolution diversity reward (new results since our camera-ready version of the paper)
- Note: this is the best ROUGE-2 result ever reported on DUC-06.

DUC-07 results

Table: ROUGE-2 recall (R) and F-measure (F) results (%)

	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), using web search	**12.45**	12.29

- DUC-05 was used as training set for the objective function with single diversity reward.
- DUC-05 and 06 were used as training sets for the objective function with multi-resolution diversity reward.

DUC-07 results

Table: ROUGE-2 recall (R) and F-measure (F) results (%)

	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), using web search	**12.45**	12.29

- DUC-05 was used as training set for the objective function with single diversity reward.
- DUC-05 and 06 were used as training sets for the objective function with multi-resolution diversity reward.
- Note: this is the best ROUGE-2 F-measure result ever reported on DUC-07, and best ROUGE-2 R without web search expansion.

Outline

Summary

- Submodularity is natural fit for summarization problems (e.g., even ROUGE-N is submodular).
- A greedy algorithm using **scaled cost**: both scalable and near-optimal, thanks to submodularity.
- We have introduced a class of submodular functions: expressive and general (more advanced NLP techniques not used, but could be easily incorporated into our objective functions).
- We show the best results yet on DUC-04, 05, 06 and 07.