where $K \in \mathbb{Z}^+$ is the number of clusters (partitions), $\mathcal{A}$ is a clustering algorithm, and $r_i \in [0, 1]$ is the singleton reward of element $i \in V$. This function has a similar form to the well known submodular facility location function, but defined on a partition of the ground set. We thus call it clustered facility location. If a summary contains multiple elements from a same cluster, the element with largest singleton reward will be regarded as the “representative" of this cluster, and only the reward of this representative will be counted into the final score. This again diminishes returns of choosing elements from the same cluster and therefore $f^{\text{C-facility}}$ is submodular.

Fidelity shell components

Given a ground set $V$, we define fidelity components as

$$f^{\text{fidelity}}_{\alpha}(V, \mathcal{C}_i) = \frac{1}{|V|} \sum_{i \in V} \min \left\{ \frac{C_i(S)}{C_i(V)}, \alpha \right\},$$

where $0 < \alpha \leq 1$ is a saturation threshold and $C_i : 2^V \rightarrow \mathbb{R}$ is a monotone submodular function modeling how $S$ covers the information contained in $i$. This function is the normalized version of the coverage function defined in [28]. Basically, the saturation threshold controls how much of a given element $i \in V$ should be covered; once $C_i(S)$ is large enough such that the ratio of it over its largest possible value ($C_i(V)$) is above threshold, covering more of $i$ does not further increase the function value. Therefore, a larger value of $f^{\text{fidelity}}$ tends to have more $i \in V$ well covered. When a document is given, we can instantiate different submodular shells using a variety of $C_i$.

5.3 A Submodular Loss Function

The most widely used evaluation criteria for summarization is the ROUGE score, which is basically a submodular function that counts n-gram recall rate over human summaries. 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 [26] can be written as the following set function:

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

where $r_{e,i}$ is the number of times n-gram $e$ occurs in reference summary $i$. $f_{\text{ROUGE-N}}(S)$ is submodular, as shown in [28], but cannot be used as a loss function since it basically measures “accuracy” rather than loss. An alternative is to use $1 - f_{\text{ROUGE-N}}(S)$ as a loss function, but this is supermodular. Note that in order to have the risk of the approximated learned model bounded, performance guarantees are required for the approximation algorithms used in loss augmented inference. When using $1 - f_{\text{ROUGE-N}}$, which is supermodular as a loss function, in the objective function for loss augmented inference (Eqn. (3)) along with a submodular shell mixture as the score function, the resulting objective function for LAI is then a submodular function plus a supermodular function. While an algorithm (e.g., submodular-supermodular procedure [33, 17]) is available to approximately optimize the sum of a submodular function and a supermodular function, performance guarantees usually do not exist for these algorithms (although this strategy might work well in practice and should ultimately be tested). Therefore, when using one-minus-ROUGE as the loss function, the greedy algorithm no longer provides a near-optimal solution when applied to the non-submodular objective, and the risk bound shown in Theorem 1 no longer holds.

To address this issue, we propose a ROUGE-like loss function that measures the “complement recall”:

$$\ell_{\text{ROUGE}}(S) \triangleq \frac{\sum_{e \in \bar{R}} \omega_e c_e(S)}{\sum_{e \in \bar{R}} \omega_e r_{e}},$$

where $\bar{R} = N \setminus \bigcup_i R_i$, and $N$ is the set of all the n-grams occur in the set of documents, and $r_e = c_e(V)$ is the number of times n-gram $e$ occurs in all the documents, $\omega_e$ is a non-negative weight for $e$, and $\bar{R}$ is the set of n-grams that are not covered by any human reference summary. Instead of counting with respect to a reference summary, $\ell_{\text{ROUGE}}$ counts the n-grams of a candidate summary $S$ w.r.t. the complement of reference summaries.

Intuitively, we want a summary $S$ to cover as many reference n-grams as possible so that it will get a high ROUGE-score; this is similar to having $S$ be large and overlapping as little as possible with the n-grams that are not in human references. In this sense, $\ell_{\text{ROUGE}}$ measures the portion of how many n-grams in the complement of the reference n-grams set are covered, and when comparing summaries with the same size, the smaller $\ell_{\text{ROUGE}}$ is, the better. The best case, i.e., the human reference itself, will have $\ell_{\text{ROUGE}}$ equal to 0.

Obviously, a poor summary that would also have $\ell_{\text{ROUGE}}$ equal to 0 is an empty summary. It is worth noting that $\ell_{\text{ROUGE}}$ only makes sense when comparing summaries that are close to the same budget. Fortunately, most summarization algorithms try to consume every bit of the budget in order to consume as much information as possible under the budget constraint. For summaries produced in this way, $\ell_{\text{ROUGE}}$ offers a fair indicator of their quality: the smaller the loss value, the larger the number of reference n-gram overlaps there are, and therefore the better the summary. We