Combining Graph Degeneracy and Submodularity for Unsupervised Extractive Summarization

Antoine J.-P. Tixier, Polykarpos Meladianos, Michalis Vazirgiannis
Data Science and Mining Team (DaSciM)
École Polytechnique
Palaiseau, France

Abstract

We present a fully unsupervised, extractive text summarization system that leverages a submodularity framework introduced by past research. The framework allows summaries to be generated in a greedy way while preserving near-optimal performance guarantees. Our main contribution is the novel coverage reward term of the objective function optimized by the greedy algorithm. This component builds on the graph-of-words representation of text and the k-core decomposition algorithm to assign meaningful scores to words. We evaluate our approach on the AMI and ICSI meeting speech corpora, and on the DUC2001 news corpus. We reach state-of-the-art performance on all datasets. Results indicate that our method is particularly well-suited to the meeting domain.

1 Introduction

We present an extractive text summarization system and test it on automatic meeting speech transcriptions and news articles. Summarizing spontaneous multiparty meeting speech text is a difficult task fraught with many unique challenges (McKeown et al., 2005). Rather than the well-formed grammatical sentences found in traditional documents, the input data consist of *utterances*, or fragments of speech transcripts. Information is diluted across utterances due to speakers frequently hesitating and interrupting each other, and noise abounds in the form of disfluencies (often expressed with filler words such as “um”, “uh-huh”, etc.) and unrelated chit-chat. Since human transcriptions are very costly, the only transcriptions available in practice are often Automatic Speech Recognition (ASR) output. Recognition errors introduce much additional noise, making the task of summarization even more difficult. In this paper, we use ASR output as our sole input, and do not make use of additional data such as prosodic features (Murray et al., 2005).

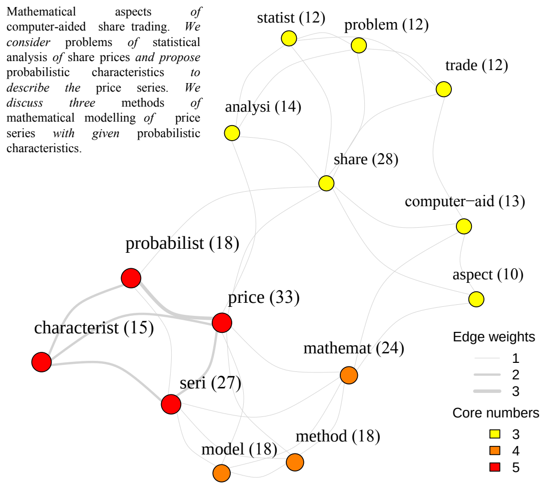
2 Background

2.1 Graph-of-words representation

A graph-of-words represents a piece of text as a network whose nodes are unique terms in the document, and whose edges encode some kind of term-term relationship information. Unlike the traditional vector space model that assumes term independence, a graph-of-words is an information-rich structure, and enables many powerful tools from graph theory to be applied to NLP tasks. The most famous example is probably the use of PageRank for unsupervised keyword extraction and document summarization (Mihalcea and Tarau, 2004).

More recent unsupervised NLP studies based on graphs reached state-of-the-art performance on a variety of tasks such as multi-sentence compression, information retrieval, real-time subevent detection from text streams, keyword extraction, and real-time topic detection (Filippova, 2010; Rousseau and Vazirgiannis, 2013; Meladianos et al., 2015; Tixier et al., 2016a; Meladianos et al., 2017).

While several variants of the graph-of-words representation exist, with different levels of sophistication and many graph building and graph mining parameters (Tixier et al., 2016b), we stick here to the traditional configuration of (Mihalcea and Tarau, 2004), which simply records co-occurrence statistics. In this setting, as illustrated in Figure 1, an undirected edge is drawn between two nodes if the unigrams they represent co-occur within a window of fixed size W that is slided over the full text from start to finish, overspanning sentences. In addition, edges are assigned integer weights matching co-occurrence counts. This approach follows the *Distributional Hypothesis* (Harris, 1954), in that it assumes the existence and strength of the dependence between textual units to be solely determined by the frequency with which they share local contexts of occurrence.



Mathematical aspects *of* computer-aided share trading. *We consider* problems *of* statistical analysis *of* share prices *and propose* probabilistic characteristics *to describe the* price series. *We discuss three* methods *of* mathematical modelling *of* price series *with given* probabilistic characteristics.

Figure 1: Undirected, weighted graph-of-words example. $W = 8$ and overspans sentences. Stemmed words, weighted k-core decomposition. Numbers inside parentheses are CoreRank scores. For clarity, non-(nouns and adjectives) in *italic* have been removed.

2.2 Graph degeneracy

Within the rest of this subsection, we will consider $G(V, E)$ to be an undirected, weighted graph with $n = |V|$ nodes and $m = |E|$ edges. The concept of graph degeneracy was introduced by (Seidman, 1983) and first applied to the study of cohesion in social networks. It is inherently related to the k-core decomposition technique.

k-core. A core of order k (or k-core) of G is a maximal connected subgraph of G in which every vertex v has at least degree k. The degree of v is the sum of the weights of its incident edges. Note that here, since edge weights are integers (co-occurrence counts), node degrees, and thus, the k's, are also integers.

The **k-core decomposition** of G is the set of all its cores from 0 or 1 (G itself, respectively in the disconnected/connected case) to k_{max} (its main core). As shown in Figure 2, it forms a hierarchy of nested subgraphs whose cohesiveness and size respectively increase and decrease with k.

The higher-level cores can be viewed as a *filtered version* of the graph that excludes noise (actually, the main core of a graph is a coarse approximation of its densest subgraph). This property of the core decomposition is highly valuable when dealing with graphs constructed from noisy text. The **core number** of a node is the highest order of a core that contains this node. As detailed in Algorithm 1, the k-core decomposition is obtained by implementing a pruning process that iteratively removes the lowest degree nodes from the graph.

Algorithm 1 k-core decomposition

```
Input: Undirected graph $G = (V, E)$
Output: Core numbers $c(v), \forall v \in V$
$i \leftarrow 0$
while $|V| > 0$ do
    while $\exists v : degree(v) \leq i$ do
        $c(v) \leftarrow i$
        $V \leftarrow V \setminus \{v\}$
        $E \leftarrow E \setminus \{(u, v) | u \in V\}$
    end while
    $i \leftarrow i + 1$
end while
```

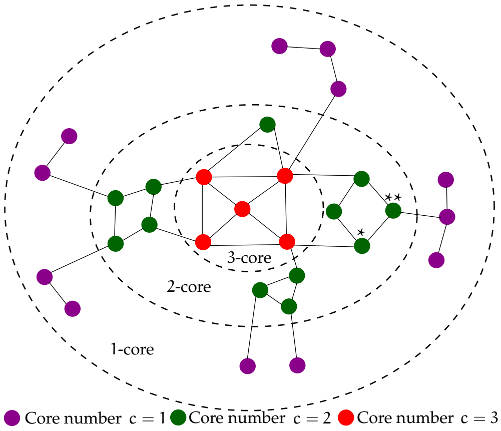


Figure 2: k-core decomposition of a graph and illustration of the value added by CoreRank. While nodes $\star$ and $\star\star$ have the same core number (=2), node $\star$ has a greater CoreRank score (3+2+2=7 vs 2+2+1=5), which better reflects its more central position in the graph.

Time complexity. While linear algorithms are available to compute the core decomposition of unweighted graphs (Batagelj and Zaversnik, 2003), it is slightly more expensive to obtain in the weighted case (our setting here), and requires $\mathcal{O}(m \log(n))$ (Batagelj and Zaveršnik, 2002). Finally, building a graph-of-words is linear: $\mathcal{O}(nW)$. Overall though, the whole pipeline remains very affordable, given that word co-occurrence networks constructed from single documents rarely feature more than hundreds of nodes. In fact, when dealing with single, short pieces of text, the k-core decomposition is fast enough to be used in real-time settings (Meladianos et al., 2017).

2.3 Submodularity and extractive summarization

Just like their convex counterparts in the continuous case, submodular functions share unique properties that make them conveniently optimizable. For this reason, they are are popular and have been applied to a variety of real-world problems, such as viral marketing (Kempe et al., 2003), sensor placement (Krause et al., 2008), and document summarization (Lin and Bilmes, 2011). In what follows, we briefly introduce the concept of submodularity and outline how it spontaneously comes into play when dealing with extractive summarization. For clarity and consistency, we provide explanations within the context of document summarization (without loss of generality).

Submodularity. A set function $F : 2^V \to \mathbb{R}$ where $V = \{v_1, ..., v_n\}$ is said to be *submodular* if it satisfies the property of *diminishing returns* (Krause and Golovin, 2012):

$$\forall A \subseteq B \subseteq V \setminus v, F(A \cup v) - F(A) \geq F(B \cup v) - F(B) \tag{1}$$

If F measures summary quality, *diminishing returns* means that the gain of adding a new sentence to a given summary should be greater than the gain of adding the same sentence to a larger summary containing the smaller one.

Monotonocity. Trivially, a set function is *monotone non-decreasing* if:

$$\forall A \subseteq B, F(A) \leq F(B) \tag{2}$$

Which means that the quality of a summary can only increase or stay the same as it grows in size, i.e., as we add sentences to it.

Budgeted maximization. The task of extractive summarization can be viewed as the selection, under a budget constraint, of the subset of sentences that best represents the entire set (i.e., the document). This problem translates to a combinatorial optimization task:

$$\arg\max_{S \subseteq V} F(S) \mid \sum_{v \in S} c_v \leq B \tag{3}$$

Where S is a subset of the full set of sentences V (i.e., a summary), $c_v \geq 0$ is the cost of sentence v, and B is the budget. Finally, F is a summary quality scoring set function, mapping 2^V (the finite ensemble of all subsets of V, i.e., of all possible summaries), to $\mathbb{R}$. In other words, F assigns a single numeric score to a given summary.

While finding an exact solution for Equation 3 is NP-hard, it was proven that under a cardinality constraint (unit costs), a greedy algorithm can approach it with factor $(e-1)/e \approx 0.63$ in the worst case (Nemhauser et al., 1978). However, for this guarantee to hold, F has to be submodular and monotone non-decreasing.

More recently, (Lin and Bilmes, 2010) proposed a modified greedy algorithm whose solution is guaranteed to be at least $1 - 1/\sqrt{e} \approx 0.39$ as good as the best one, under a general budget constraint (not necessarily unit costs). Empirically, the approximation factor was shown to be close to 90%. The constraints on F remain unchanged. More precisely, the algorithm of (Lin and Bilmes, 2010) iteratively selects the sentence that maximizes the ratio of objective function gain to scaled cost:

$$\frac{F(G \cup v) - F(G)}{c_v^r} \tag{4}$$

Where G is the current summary, c_v is the cost of sentence v (e.g., number of words, bytes...), and $r > 0$, the scaling factor, adjusts for the fact that the objective function F and the cost of a sentence might be expressed in different units and thus not be directly comparable.

Objective function. The choice of F is what matters here. Naturally, F should capture the desirable properties in a summary, which have traditionally been formalized in the literature as *relevance* and *non-redundancy*.

A well-known function capturing both aspects is Maximum Marginal Relevance (MMR) (Carbonell and Goldstein, 1998). Unfortunately, MMR *penalizes* for redundancy, which makes it non-monotone. Therefore, it cannot benefit from the near-optimality guarantees. To address this issue, (Lin and Bilmes, 2011) proposed to *positively reward* diversity, with objective function:

$$F(S) = C(S) + \lambda D(S) \tag{5}$$

Where C and D respectively reward coverage and diversity, and $\lambda \geq 0$ is a trade-off parameter. $\lambda D(S)$ can be viewed as a regularization term. We used an objective function of the form described by Equation 5 in our system. In the next subsection, we present and motivate our choices for C

and D.

3 Proposed system

Our system can be broken down into the four modules shown in Figure 3, which we detail in what follows.

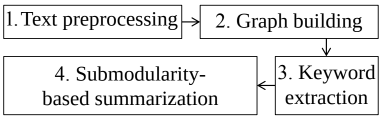


Figure 3: Overarching system process flow

3.1 Text preprocessing

The fully unsupervised nature of our system gives it the advantage of being applicable to different languages (and different types of textual input) with only minimal changes in the preprocessing steps. A necessary first step is thus to detect the language of the input text. So far, our model supports English and French, although our experiments were ran for the English language only.

• **Meeting speech**: utterances shorter than 0.85 second are then pruned out, words are lowercased and stemmed, and specific flags introduced by the ASR system (e.g., indicating inaudible sounds, such as "{vocalsound}" in English) are removed. Punctuation is also discarded. Custom stopwords and fillerwords for meeting speech, learned from the development sets of the AMI and ICSI corpora[1], are also discarded. French stopwords and fillerwords were learned from a database of French speech curated from various sources[2]. The surviving words are considered as node candidates for the next phase, without any part-of-speech-based filtering. Note that the absence of requirement for a POS tagger makes our system even more flexible.

• **Traditional documents**: standard stopwords are removed (e.g., SMART stopwords[3] for the English language), punctuation is removed, and words are lowercased and stemmed.

In parallel, a copy of the original untouched utterances/sentences is created. It is from this set that the algorithm will select from to generate the summary at step 4. In the meeting domain only, in order to improve readability, the last 3 words of each utterance are eliminated if they are filler words, and repeated consecutive unigrams (e.g. "remote remote"), and bigrams (e.g. "remote control remote control") are collapsed to single terms ("remote", "remote control"). Note that these extra cleaning steps were performed for our system as well as all the baselines.

3.2 Graph-building

A word co-occurrence network, as defined in Subsection 2.1, is built. The size of the sliding window was tuned on the development sets of each dataset, as will be explained in Subsection 4.4.

3.3 Keyword extraction and scoring

We used the *Density* and *CoreRank* heuristics introduced by (Tixier et al., 2016a). In brief, these techniques are based on the assumption, verified empirically, that *spreading influence* is a better "keywordedness" metric than random walk-based ones, such as PageRank. Influential spreaders are those nodes in the graph that can reach a large portion of the other nodes in the network at minimum time and cost. Research has shown (Kitsak et al., 2010) that the spreading influence of a node is better captured by its core number, because unlike the eigenvector centrality or PageRank measures, which only capture individual *prestige*, graph degeneracy also takes into account the extent to which a node is part of a dense, cohesive part of the graph. Such positional information is highly valuable in determining the ability of the node to propagate information throughout the network.

More precisely, the "Density" and "CoreRank" techniques were shown by (Tixier et al., 2016a) to reach state-of-the-art unsupervised keyword extraction performance on medium and large documents, respectively. Both methods decompose the word co-occurrence network of a given piece of text with the weighted k-core algorithm.

• "Density" then computes the density of each k-core subgraph and selects the optimal cut-off k_{best} in the hierarchy as the elbow in the *density* vs. k curve. It finally returns the members of the k_{best}-core of the graph as keywords. The assumption is that it is valuable to descend the hierarchy of cores as long as the desirable density properties are maintained, but once they are lost (as identified by the elbow), it is time to stop.

• The second method, "CoreRank", assigns to each node a score computed as the sum of the

[1] most frequent words followed by manual inspection

[2] available at https://github.com/Tixierae/EMNLP2017_NewSum

[3] http://jmlr.org/papers/volume5/lewis04a/a11-smart-stop-list/english.stop

core numbers of its neighbors (see Figure 1), and retains the top $p\%$ nodes as keywords (we used $p = 0.15$). As illustrated in Figure 2, by decreasing granularity from the subgraph to the node level, CoreRank generates a ranking of nodes that better captures their structural position in the graph. Also, stabilizing scores across node neighborhoods increases even more the inherent noise robustness property of graph degeneracy, which is particularly desirable when dealing with noisy text such as automatic speech transcriptions.

We encourage the reader to refer to the original paper for more information about the Density and CoreRank heuristics.

3.4 Extractive summarization

An objective function of the form presented in Equation 5 and the modified greedy algorithm of (Lin and Bilmes, 2010) are finally used to compose summaries by selecting from the original utterances with coverage and diversity functions as detailed next.

- *Coverage function.* We chose a concept-based coverage function. Such functions fulfill the monotonicity and submodularity requirements (Lin and Bilmes, 2011). More precisely, we compute the coverage of a candidate summary S as the weighted sum of the scores of the keywords it contains:

$$C(S) = \sum_{i \in S} n_i w_i \tag{6}$$

Where n_i is the number of times keyword i appears in S, and w_i is the score of keyword i. Non-keywords are not taken into account. Therefore, a summary not containing any keyword gets a null score. Remember that the keywords and their scores are given by the "Density" and "CoreRank" techniques, respectively for the AMI and ICSI corpora.

Note that (Riedhammer et al., 2008a) also used a concept-based relevance measure. However, the way we define, and the mechanism by which we extract and assign scores to concepts radically differ. Our degeneracy-based methods natively assign weights to all the words in the graph, and then extract keywords based on those weights, while (Riedhammer et al., 2008a) consider all n-grams and then use a basic frequency-based weighting scheme. Our work is also related to (Lin et al., 2009), but unlike us, the authors use a sentence semantic graph and a different objective function.

- *Diversity reward function.* We encourage diversity by taking into account the proportion of keywords covered by a candidate summary, irrespective of the scores of the keywords:

$$D(S) = N_{keywords \in S} / N_{keywords} \tag{7}$$

Where $N_{keywords \in S}$ is the number of (unique) keywords contained in the summary, and $N_{keywords}$ is the total number of keywords extracted for the meeting. Promoting non-redundancy is important as our coverage term does not inherently penalizes for redundancy, unlike for instance (Gillick et al., 2009).

4 Experimental setup

4.1 Datasets

We tested our approach on ASR output and regular text. The lists of meetings/documents IDs we used for development and testing are available on the project online repository[4].

4.1.1 Meeting speech transcriptions

We used two standard datasets very popular in the field of meeting speech summarization, the AMI and ICSI corpora.

- The **AMI corpus** (McCowan et al., 2005) comprises ASR transcripts for 137 meetings where 4 participants play a role within a fictive company. Average duration is 30 minutes (843 utterances, 6758 words, unprocessed). Each meeting is associated with a human-written abstractive summary of 300 words on average, and with a human-composed extractive summary (140 utterances on average). We used the same test set as in (Riedhammer et al., 2008b), featuring 20 meetings.
- The **ICSI corpus** (Janin et al., 2003) is a collection of 57 real life meetings involving between 2 and 6 participants. The average duration, 56 minutes, is much longer than for the AMI meetings, which reflects in the average size of the ASR transcriptions (1454 utterances, 15211 words, unprocessed). For consistency with previous work, we selected the standard test set of 6 meetings. For each meeting of this test set, 3 human abstractive and 3 human extractive summaries are available, of respective average sizes 390 words and 133 utterances.

[4] `https://github.com/Tixierae/EMNLP2017_NewSum` (`name_lists.txt`)

Note that for both the AMI and ICSI corpora, the ASR word error rate is quite high: it approaches 37%. For each corpus, we constructed a development set of 15 meetings randomly selected from the training set in order to perform parameter tuning.

4.1.2 Traditional documents

We also tested our approach on the **DUC2001** corpus[5]. This collection comprises 304 newswire/newspaper articles of average size 800 words. Each document is associated with a human-written abstractive summary of about 100 words. After removing the 13 articles that did not have an abstract and/or a body, whose bodies were shorter than 200 words, and whose abstracts contained less than 10 words, we generated a small development set of 15 randomly selected articles for parameter tuning. We then used the remaining documents as the test set, removing the ones whose size differed too much from the size of the articles in the development set (by at least 2 standard deviations, i.e. exceeded 46 sentences in size, see Fig 4). This left us with a test set of 207 documents.

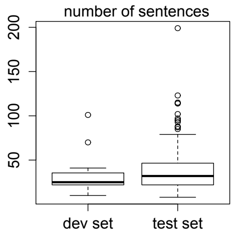


Figure 4: Size of the DUC2001 documents in development and test sets.

4.2 Evaluation

To align with previous efforts, the extractive summaries generated by our system and the baselines (that will be presented subsequently) were compared against the human *abstractive* summaries. We used the ROUGE-1 evaluation metric (Lin, 2004). ROUGE, based on n-gram overlap, is the standard way of evaluating performance in the field of textual summarization. In particular, ROUGE-1, which works at the unigram level, was shown to significantly correlate with human evaluations. While it has been suggested than correlation may be weaker in the meeting domain (Liu and Liu, 2008), we stuck to ROUGE because of the lack of a clear substitute, and for consistency with the literature, as a very large majority of studies previously published in the domain use ROUGE.

For each dataset, and for a given summarization method, ROUGE scores were computed for each meeting in the test set and then averaged to obtain an overall score for the method (macro-averaging). For the ICSI corpus, 3 human abstractive summaries are available for each meeting in the test set, so an average score was first computed.

4.3 Baseline systems

We benchmarked the performance of our system against six different baselines, presented below. The first two baselines were included based on the best practice recommendation of (Riedhammer et al., 2008b), in order to ease cross-comparison with other studies.

Random. This system randomly selects elements from the full list of utterances/sentences until the budget is violated. Since this approach is stochastic, ROUGE scores were averaged across 30 runs.

Longest greedy. Here, the longest utterance/sentence is selected at each step until the size constraint is satisfied.

TextRank (Mihalcea and Tarau, 2004). An undirected complete graph is built where nodes are utterances/sentences and edges are weighted according to the normalized content overlap of their endpoints. Finally, weighted PageRank is applied and the highest ranked nodes are selected for inclusion in the summary. We used a publicly available Python implementation[6].

ClusterRank (Garg et al., 2009). *AMI & ICSI only*. ClusterRank is an extension of TextRank tailored to meeting summarization. Utterances are first clustered based on their position in the transcript and their TF-IDF cosine similarity. Then, a complete graph is built from the clusters, with normalized cosine similarity edge weights. Finally, each utterance is assigned a score based on the weighted PageRank score of the node it belongs to and its cosine similarity with the node centroid. The utterances associated with the highest scores are then added to the summary, if they differ enough from it. Since the authors did not make their code publicly available, we wrote our own implementation in Python[7]. We set the win-

[5] http://www-nlpir.nist.gov/projects/duc/guidelines/2001.html

[6] https://github.com/summanlp/textrank

[7] available on the project repository.

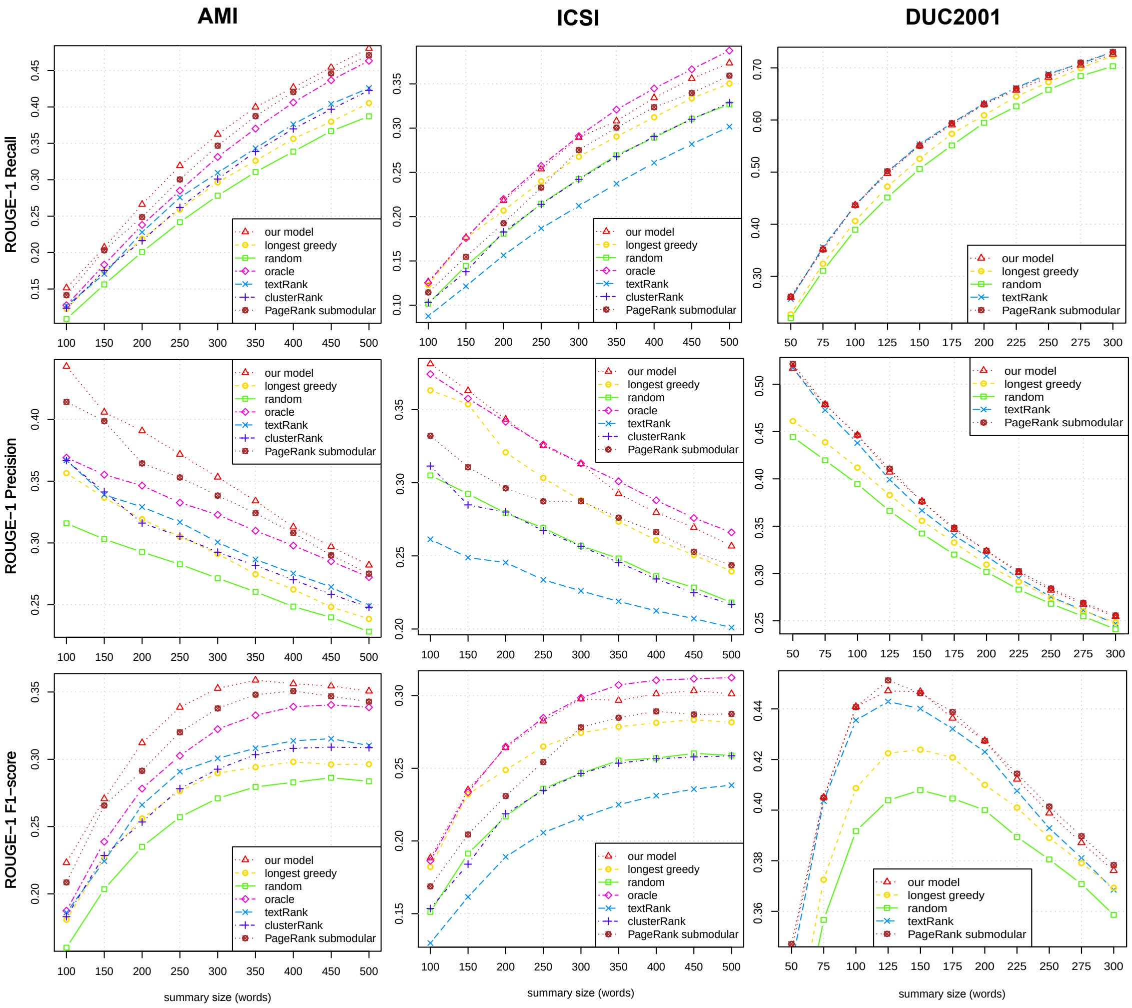


Figure 5: ROUGE-1 score comparisons for various budgets, on the 3 datasets used in this study.

dow threshold parameter to 3 like in the original paper, but increased the similarity threshold from 0.4 to 0.6 because 0.4 returned too many clusters.

PageRank submodular (PRsub). This baseline is exactly the same as our system, the only difference being that keyword scores are obtained through weighted PageRank rather than via a degeneracy-based technique (Density or CoreRank).

Oracle. *AMI & ICSI only*. This last baseline randomly selects utterances from the human extractive summaries until the budget has been reached. Again, we average ROUGE scores over 30 runs to account for the randomness of the procedure. Note that this approach assumes the human extractive summaries to be the best possible ones, which is arguable.

4.4 Parameter tuning

• λ and r. Recall that the main tuning parameters of our method and the PageRank submodular baseline (PRsub) are λ, which controls the trade-off between the coverage and the diversity terms C and D of our objective function, and r, the scaling factor, which makes the gain in objective function value and utterance cost comparable (see Equation 4). To tune these parameters, we conducted a grid search on the development set of each corpus, retaining the parameter combination maximizing the average ROUGE-1 F1-score, for summaries of fixed size equal to 300 and 100 words, respectively for the AMI & ICSI and the DUC2001 corpora. More precisely, our grid had axes $[0, 7]$ and $[0, 2]$ for λ and r respectively, with steps of 0.1 in each case. The best λ and r for each dataset are summarized in Table 1.

• W and heuristic. Still on the development sets of each collection, we also experimented with two window sizes for building the word co-occurrence network (6 and 12), and for our model, whether we should use the Density or CoreRank technique. The best window size was 12 on the AMI and ICSI corpora, and 6 on DUC2001. The Density method turned out to be best on the AMI corpus, while CoreRank yielded better results on the ICSI and DUC2001 corpora.

The reason why is not entirely clear. (Tixier et al., 2016a) initially found that with respect to keyword extraction, Density was better suited to medium-size documents ($\sim$ 400 words) while CoreRank was superior on longer documents ($\sim$ 1,300 words), because the latter is working at a finer granularity level (node level instead of subgraph level), and thus enjoys more flexibility. However, the AMI corpus comprises much bigger pieces of text (2,200 words on average, after preprocessing). Therefore, we could have expected the CoreRank heuristic to give better results on this dataset also. We hypothesize that the difference in task might explain why this is not the case. Indeed, in keyword extraction, we are interested in *selecting* keywords for direct comparison with the gold standard, whereas in summarization, we are only interested in *scoring* keywords, as an intermediary step towards sentence scoring and selection. Therefore, in summarization, working at the subgraph level and extracting larger numbers of keywords is not directly equivalent to sacrificing precision, since the less relevant keywords will have minimal impact on the sentence selection process due to their low scores.

System	AMI	ICSI	DUC2001
Our model	$(2, 0.9)$	$(5, 0.3)$	$(0.6, 0.1)$
PRsub	$(4.7, 0.5)$	$(4, 0.6)$	$(1.6, 0.2)$

Table 1: Optimal parameter values (λ,r) for our system and the submodular baseline.

As shown in Table 1, the λ values are all nonzero (and quite high), indicating that including a regularization term favoring diversity in our objective function is necessary. Moreover, the significantly greater values reached by λ on the AMI & ICSI datasets show that ensuring diversity is even more important when dealing with meeting transcripts, most probably because there is much more redundancy in spontaneous, noisy utterances than in sentences belonging to properly written news article, and also because more (sub)topics are discussed during meetings.

5 Results

5.1 Quantitative results

We consider the cost of an utterance/a sentence to be the number of words it contains, and the budget to be the maximum size allowed for a summary, measured in number of words. For each meeting/document in the test sets, we generated extractive summaries with budgets ranging from 100 to 500 words (AMI & ICSI corpora) and from 50 to 300 words (DUC2001 collection), with steps of 50 in each case.

Results for all datasets and all budgets are shown in Figure 5, while Tables 2, 3, and 4 provide detailed comparisons for the budget corresponding to the best performance achieved by a non-oracle system, respectively on the AMI, ICSI, and DUC2001 datasets. We tested for statistical significance in macro-averaged F1 scores using the non-parametric version of the t-test, the Mann-Whitney U test[8].

System	Recall	Precision	F-1 score
Our model	39.98	33.40	$35.88^\star$
PRsub	38.73	32.41	34.80
Oracle	37.02	30.99	33.27
TextRank	34.33	28.66	30.82
ClusterRank	33.87	28.18	30.35
Longest greedy	32.61	27.47	29.41
Random	31.06	26.05	27.95

Table 2: Macro-averaged ROUGE-1 scores on the AMI test set (20 meetings) for summaries of 350 words. *Statistically significant difference ($p < 0.03$) w.r.t. all baselines except PRsub.

System	Recall	Precision	F-1 score
Oracle	36.64	27.59	31.16
Our model	35.60	26.94	$30.34^\star$
PRsub	33.97	25.28	28.70
Longest greedy	33.37	25.06	28.33
Random	31.06	22.83	26.02
ClusterRank	31.00	22.48	25.78
TextRank	28.19	20.71	23.57

Table 3: Macro-averaged ROUGE scores on the ICSI test set (6 meetings) for summaries of 450 words. *Statistically significant difference ($p < 0.05$) w.r.t. all baselines except the oracle and PRsub.

System	Recall	Precision	F-1 score
PRsub	50.17	41.08	45.13
Our model	49.69	40.71	$44.71^\star$
TextRank	50.00	39.92	44.29
Longest greedy	47.22	38.29	42.25
Random	45.13	36.61	40.39

Table 4: Macro-averaged ROUGE scores on the DUC2001 test set (207 documents) for summaries of 125 words. *Statistically significant difference ($p < 0.03$) w.r.t. the Longest greedy and Random baselines.

- **Meeting domain**. Our approach significantly outperforms all baselines on the AMI corpus (including the oracle) and all systems on the ICSI corpus (except the oracle), both in terms of precision and recall. Also, our system proves con-

[8] https://stat.ethz.ch/R-manual/R-devel/library/stats/html/wilcox.test.html

sistently better throughout the different summary sizes. Until the peak is reached, the margin in F1 score between our model and the competitors even tend to widen as the budget increases.

Performance is weaker for all models on the ICSI corpus because in that case the system summaries have to jointly match 3 human summaries of different sizes (instead of a single summary), which is a much more difficult task.

Best performance is attained for a larger budget on the ICSI corpus (450 vs. 350 words), which can be explained by the fact that the ICSI human summaries tend to be larger than the AMI ones (390 vs 300 words, on average). Finally, remember that the extractive summaries generated by the systems were compared against the *abstractive* summaries freely written by human annotators, using their own words. This makes it impossible for extractive systems to reach perfect scores, because the gold standard contains words that were never used during the meeting, and thus that do not appear in the ASR transcriptions. Overall, our model is very competitive to the oracle, which is notable since the oracle has direct access to the human extractive summaries.

• **Regular documents**. The absolute ROUGE scores and the margins between systems are much greater (resp. smaller) than on the AMI and ICSI corpora, confirming without surprise that summarization is a much easier task when performed on well-written documents than on spontaneous meeting speech transcriptions. Although very close (0.42 difference in F1-score), our method does not reach absolute best performance, which is attained by the submodular baseline with PageRank-based coverage function, for summaries of 125 words (average size of the gold standard summaries is about 100 words). The absence of superiority on this dataset might be explained by the fact that graph degeneracy really adds value when dealing with noisy input, such as automatic speech transcriptions. However, on regular documents, the recognized superiority of degeneracy-based techniques over PageRank (Tixier et al., 2016a; Rousseau and Vazirgiannis, 2015) for keyword extraction does not seem to translate into a significantly better measure of coverage for sentence scoring.

5.2 Qualitative results

Instead of providing a single sample summary at the end of this paper, we deployed our system as an interactive web application[9]. With the interface, the user can generate summaries with our system for all the meetings/documents in the AMI, ICSI, and DUC2001 test sets. Custom files are accepted as well, and links to examples of such files in French and English are provided.

What can be observed in the meeting domain is that while the keywords extracted tend to be very relevant and their scores meaningful, and while the utterances selected by our system tend to have good coverage and relatively low redundancy, the summaries suffer in readability, which can be explained by the fully extractive nature of our approach, and the low quality of the input (37% word error rate). This qualitative aspect of performance is not captured by ROUGE-1 which simply computes unigram overlap statistics.

6 Conclusion

We presented a fully unsupervised system that uses a powerful submodularity framework introduced by past research to generate extractive summaries of textual documents in a greedy way with near-optimal performance guarantees. Our principal contribution is in the coverage term of the objective function that is optimized by the greedy algorithm. This term leverages graph degeneracy applied on word co-occurrence networks to rank words according to their structural position in the graph. Evaluation shows that our system reaches state-of-the-art extractive performance, and is especially well-suited to be used on noisy text, such as ASR output from meetings. Future work should focus on improving the readability of the final summaries. To this purpose, unsupervised graph-based sentence compression and/or natural language generation techniques, like in (Filippova, 2010; Mehdad et al., 2013) seem very promising.

7 Acknowledgments

We are thankful to the three anonymous reviewers for their helpful comments and suggestions, and to Prof. Benoît Favre for his kind help in getting access to the meeting datasets. This research was supported by the OpenPaaS::NG project.

[9] `http://bit.ly/2r5jeL0` (works better in Chrome).

References

Vladimir Batagelj and Matjaž Zaveršnik. 2002. Generalized cores. *arXiv preprint cs/0202039* .

Vladimir Batagelj and Matjaz Zaversnik. 2003. An o (m) algorithm for cores decomposition of networks. *arXiv preprint cs/0310049* .

Jaime Carbonell and Jade Goldstein. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, pages 335–336.

Katja Filippova. 2010. Multi-sentence compression: Finding shortest paths in word graphs. In *Proceedings of the 23rd International Conference on Computational Linguistics*. Association for Computational Linguistics, pages 322–330.

Nikhil Garg, Benoit Favre, Korbinian Reidhammer, and Dilek Hakkani Tür. 2009. Clusterrank: a graph based method for meeting summarization. Technical report, Idiap.

Daniel Gillick, Benoit Favre, Dilek Hakkani-Tür, Bernd Bohnet, Yang Liu, and Shasha Xie. 2009. The icsi/utd summarization system at tac 2009. In *TAC*.

Zellig S Harris. 1954. Distributional structure. *Word* 10(2-3):146–162.

Adam Janin, Don Baron, Jane Edwards, Dan Ellis, David Gelbart, Nelson Morgan, Barbara Peskin, Thilo Pfau, Elizabeth Shriberg, Andreas Stolcke, et al. 2003. The icsi meeting corpus. In *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*. IEEE, volume 1, pages I–364.

David Kempe, Jon Kleinberg, and Éva Tardos. 2003. Maximizing the Spread of Influence through a Social Network. In *Proceedings of the 9th International Conference on Knowledge Discovery and Data Mining*. pages 137–146.

Maksim Kitsak, Lazaros K Gallos, Shlomo Havlin, Fredrik Liljeros, Lev Muchnik, H Eugene Stanley, and Hernán A Makse. 2010. Identification of influential spreaders in complex networks. *Nature Physics* 6(11):888–893.

Andreas Krause and Daniel Golovin. 2012. Submodular function maximization. *Tractability: Practical Approaches to Hard Problems* 3(19):8.

Andreas Krause, Jure Leskovec, Carlos Guestrin, Jeanne VanBriesen, and Christos Faloutsos. 2008. Efficient sensor placement optimization for securing large water distribution networks. *Journal of Water Resources Planning and Management* 134(6):516–526.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out: Proceedings of the ACL-04 workshop*. volume 8.

Hui Lin and Jeff Bilmes. 2010. Multi-document summarization via budgeted maximization of submodular functions. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, pages 912–920.

Hui Lin and Jeff Bilmes. 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.

Hui Lin, Jeff Bilmes, and Shasha Xie. 2009. Graph-based submodular selection for extractive summarization. In *Automatic Speech Recognition & Understanding, 2009. ASRU 2009. IEEE Workshop on*. IEEE, pages 381–386.

Feifan Liu and Yang Liu. 2008. Correlation between rouge and human evaluation of extractive meeting summaries. In *Proceedings of the 46th annual meeting of the association for computational linguistics on human language technologies: Short papers*. Association for Computational Linguistics, pages 201–204.

Iain McCowan, Jean Carletta, W Kraaij, S Ashby, S Bourban, M Flynn, M Guillemot, T Hain, J Kadlec, V Karaiskos, et al. 2005. The ami meeting corpus. In *Proceedings of the 5th International Conference on Methods and Techniques in Behavioral Research*. volume 88.

Kathleen McKeown, Julia Hirschberg, Michel Galley, and Sameer Maskey. 2005. From text to speech summarization. In *Acoustics, Speech, and Signal Processing, 2005. Proceedings.(ICASSP'05). IEEE International Conference on*. IEEE, volume 5, pages v–997.

Yashar Mehdad, Giuseppe Carenini, Frank W Tompa, and Raymond T Ng. 2013. Abstractive meeting summarization with entailment and fusion. In *Proc. of the 14th European Workshop on Natural Language Generation*. pages 136–146.

Polykarpos Meladianos, Giannis Nikolentzos, François Rousseau, Yannis Stavrakas, and Michalis Vazirgiannis. 2015. Degeneracy-based real-time subevent detection in twitter stream. In *Ninth International AAAI Conference on Web and Social Media (ICWSM)*.

Polykarpos Meladianos, Antoine J-P Tixier, Giannis Nikolentzos, and Michalis Vazirgiannis. 2017. Real-time keyword extraction from conversations. *EACL 2017* page 462.

Rada Mihalcea and Paul Tarau. 2004. TextRank: bringing order into texts. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.

Gabriel Murray, Steve Renals, and Jean Carletta. 2005. Extractive summarization of meeting recordings. .

George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. 1978. An analysis of approximations for maximizing submodular set functionsi. *Mathematical Programming* 14(1):265–294.

Korbinian Riedhammer, Benoit Favre, and Dilek Hakkani-Tur. 2008a. A keyphrase based approach to interactive meeting summarization. In *Spoken Language Technology Workshop, 2008. SLT 2008. IEEE*. IEEE, pages 153–156.

Korbinian Riedhammer, Dan Gillick, Benoit Favre, and Dilek Hakkani-Tür. 2008b. Packing the meeting summarization knapsack. In *Ninth Annual Conference of the International Speech Communication Association*.

François Rousseau and Michalis Vazirgiannis. 2013. Graph-of-word and tw-idf: new approach to ad hoc ir. In *Proceedings of the 22nd ACM international conference on Conference on Information & Knowledge Management (CIKM)*. ACM, pages 59–68.

François Rousseau and Michalis Vazirgiannis. 2015. Main core retention on graph-of-words for single-document keyword extraction. In *European Conference on Information Retrieval*. Springer, pages 382–393.

Stephen B Seidman. 1983. Network structure and minimum degree. *Social networks* 5(3):269–287.

Antoine J-P Tixier, Fragkiskos D Malliaros, and Michalis Vazirgiannis. 2016a. A graph degeneracy-based approach to keyword extraction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Antoine J-P Tixier, Konstantinos Skianis, and Michalis Vazirgiannis. 2016b. Gowvis: a web application for graph-of-words-based text visualization and summarization. *ACL 2016* page 151.