Citation Summarization Through Keyphrase Extraction

Vahed Qazvinian
Department of EECS
University of Michigan
vahed@umich.edu

Dragomir R. Radev
School of Information and
Department of EECS
University of Michigan
radev@umich.edu

Arzucan Özgür
Department of EECS
University of Michigan
ozgur@umich.edu

Abstract

This paper presents an approach to summarize single scientific papers, by extracting its contributions from the set of citation sentences written in other papers. Our methodology is based on extracting significant keyphrases from the set of citation sentences and using these keyphrases to build the summary. Comparisons show how this methodology excels at the task of single paper summarization, and how it out-performs other multi-document summarization methods.

1 Introduction

In recent years statistical physicists and computer scientists have shown great interest in analyzing complex adaptive systems. The study of such systems can provide valuable insight on the behavioral aspects of the involved agents with potential applications in economics and science. One such aspect is to understand what motivates people to provide the $n+1^{st}$ review of an artifact given that they are unlikely to add something significant that has not already been said or emphasized. Citations are part of such complex systems where articles use citations as a way to mention different contributions of other papers, resulting in a collective system.

The focus of this work is on the corpora created based on citation sentences. A citation sentence is a sentence in an article containing a citation and can contain zero or more *nuggets* (i.e., non-overlapping contributions) about the cited article. For example the following sentences are a few citation sentences that appeared in the NLP literature in past that talk about Resnik's work.

The STRAND system **(Resnik, 1999)***, for example, uses structural markup information from the pages, without looking at their content, to attempt to align them.*

Resnik (1999) *addressed the issue of language identification for finding Web pages in the languages of interest.*

Mining the Web for bilingual text **(Resnik, 1999)** *is not likely to provide sufficient quantities of high quality data..*

The set of citations is important to analyze because human summarizers have put their effort collectively but independently to read the target article and cite its important contributions. This has been shown in other work too (Elkiss et al., 2008; Nanba et al., 2004; Qazvinian and Radev, 2008; Mei and Zhai, 2008; Mohammad et al., 2009). In this work, we introduce a technique to summarize the set of citation sentences and cover the major contributions of the target paper. Our methodology first finds the set of keyphrases that represent important information units (i.e., nuggets), and then finds the best set of k sentences to cover more, and more important nuggets.

Our results confirm the effectiveness of the method and show that it outperforms other state of the art summarization techniques. Moreover, as shown in the paper, this method does not need to calculate the full cosine similarity matrix for a document cluster, which is the most time consuming part of the mentioned baseline methods.

1.1 Related Work

Previous work has used citations to produce summaries of scientific work (Qazvinian and Radev,

Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010), pages 895–903,
Beijing, August 2010

2008; Mei and Zhai, 2008; Elkiss et al., 2008). Other work (Bradshaw, 2003; Bradshaw, 2002) benefits from citations to determine the content of articles and introduce "Reference Directed Indexing" to improve the results of a search engine.

In other work, (Nanba and Okumura, 1999) analyze citation sentences and automatically categorize citations into three groups using 160 predefined phrase-based rules to support a system for writing a survey. Previous research has shown the importance of the citation summaries in understanding what a paper contributes. In particular, (Elkiss et al., 2008) performed a large-scale study on citation summaries and their importance. Results from this experiment confirmed that the "Self Cohesion" (Elkiss et al., 2008) of a citation summary of an article is consistently higher than the that of its abstract and that citations contain additional information that does not appear in abstracts.

Kan et al. (2002) use annotated bibliographies to cover certain aspects of summarization and suggest using metadata and critical document features as well as the prominent content-based features to summarize documents. Kupiec et al. (1995) use a statistical method and show how extracts can be used to create summaries but use no annotated metadata in summarization.

Siddharthan and Teufel describe a new task to decide the scientific attribution of an article (Siddharthan and Teufel, 2007) and show high human agreement as well as an improvement in the performance of Argumentative Zoning (Teufel, 2005). Argumentative Zoning is a rhetorical classification task, in which sentences are labeled as one of Own, Other, Background, Textual, Aim, Basis, Contrast according to their role in the author's argument. These all show the importance of citation summaries and the vast area for new work to analyze them to produce a summary for a given topic.

The Maximal Marginal Relevance (MMR) summarization method, which is based on a greedy algorithm, is described in (Carbonell and Goldstein, 1998). MMR uses the full similarity matrix to choose the sentences that are the least similar to the sentences already selected for the summary. We selected this method as one of our baseline methods, which we have explained in more details in Section 4.

Fact	Occurrences
f_1: " Supervised Learning"	5
f_2: " instance/concept relations"	3
f_3: "Part-of-Speech tagging"	3
f_4: "filtering QA results"	2
f_5: "lexico-semantic information"	2
f_6: "hyponym relations"	2

Table 2: Nuggets of P03-1001 extracted by annotators.

2 Data

In order to evaluate our method, we use the ACL Anthology Network (AAN), which is a collection of papers from the Computational Linguistics journal and proceedings from ACL conferences and workshops and includes more than $13,000$ papers (Radev et al., 2009). We use 25 manually annotated papers from (Qazvinian and Radev, 2008), which are highly cited articles in AAN. Table 1 shows the ACL ID, title, and the number of citation sentences for these papers.

The annotation guidelines asked a number of annotators to read the citation summary of each paper and extract a list of the main contributions of that paper. Each item on the list is a non-overlapping contribution (nugget) perceived by reading the citation summary. The annotation strictly instructed the annotators to focus on the citing sentences to do the task and not their own background on the topic. Then, extracted nuggets are reviewed and those nuggets that have only been mentioned by 1 annotator are removed. Finally, the union of the rest is used as a set of nuggets representing each paper.

Table 2 lists the nuggets extracted by annotators for P03-1001.

3 Methodology

Our methodology assumes that each citation sentence covers 0 or more nuggets about the cited papers, and tries to pick sentences that maximize nugget coverage with respect to summary length.

These nuggets are essentially represented using keyphrases. Therefore, we try to extract significant keyphrases in order to represent nuggets each sentence contains. Here, the keyphrases are ex-

ACL-ID	Title	# citations
N03-1017	Statistical Phrase-Based Translation	180
P02-1006	Learning Surface Text Patterns For A Question Answering System	74
P05-1012	On-line Large-Margin Training Of Dependency Parsers	71
C96-1058	Three New Probabilistic Models For Dependency Parsing: An Exploration	66
P05-1033	A Hierarchical Phrase-Based Model For Statistical Machine Translation	65
P97-1003	Three Generative, Lexicalized Models For Statistical Parsing	55
P99-1065	A Statistical Parser For Czech	54
J04-4002	The Alignment Template Approach To Statistical Machine Translation	50
D03-1017	Towards Answering Opinion Questions: Separating Facts From Opinions ...	42
P05-1013	Pseudo-Projective Dependency Parsing	40
W00-0403	Centroid-Based Summarization Of Multiple Documents: Sentence Extraction, ...	31
P03-1001	Offline Strategies For Online Question Answering: Answering Questions Before They Are Asked	27
N04-1033	Improvements In Phrase-Based Statistical Machine Translation	24
A00-2024	Cut And Paste Based Text Summarization	20
W00-0603	A Rule-Based Question Answering System For Reading Comprehension Tests	19
A00-1043	Sentence Reduction For Automatic Text Summarization	19
C00-1072	The Automated Acquisition Of Topic Signatures For Text Summarization	19
W05-1203	Measuring The Semantic Similarity Of Texts	17
W03-0510	The Potential And Limitations Of Automatic Sentence Extraction For Summarization	15
W03-0301	An Evaluation Exercise For Word Alignment	14
A00-1023	A Question Answering System Supported By Information Extraction	13
D04-9907	Scaling Web-Based Acquisition Of Entailment Relations	12
P05-1014	The Distributional Inclusion Hypotheses And Lexical Entailment	10
H05-1047	A Semantic Approach To Recognizing Textual Entailment	8
H05-1079	Recognising Textual Entailment With Logical Inference	9

Table 1: List of papers chosen from AAN for evaluation together with the number of sentences citing each.

	unique	all	max freq
unigrams	229,631	7,746,792	437,308
bigrams	2,256,385	7,746,791	73,957
3-grams	5,125,249	7,746,790	3,600
4-grams	6,713,568	7,746,789	2,408

Table 3: Statistics on the abstract corpus in AAN used as the background data

pressed using N-grams, and thus these building units are the key elements to our summarization. For each citation sentence d_i, our method first extracts a set of important keyphrases, D_i, and then tries to find sentences that have a larger number of important and non-redundant keyphrases. In order to take the first step, we extract statistically significantly frequent N-grams (up to $N = 4$) from each citing sentence and use them as the set of representative keyphrases for that citing sentence.

3.1 Automatic Keyphrase Extraction

A list of keyphrases for each citation sentence can be generated by extracting N-grams that occur significantly frequently in that sentence compared to a large corpus of such N-grams. Our method for such an extraction is inspired by the previous work by Tomokiyo and Hurst (Tomokiyo and Hurst, 2003).

A language model, $\mathcal{M}$, is a statistical model that assigns probabilities to a sequence of N-grams. Every language model is a probability distribution over all N-grams and thus the probabilities of all N-grams of the same length sum up to 1. In order to extract keyphrases from a text using statistical significance we need two language models. The first model is referred to as the *Background Model ($\mathcal{BM}$)* and is built using a large text corpus. Here we build the BM using the text of all the paper abstracts provided in AAN[1]. The second language model is called the *Foreground Model ($\mathcal{FM}$)* and is the model built on the text from which keyphrases are being extracted. In this work, the set of all citation sentences that cite a particular target paper are used to build a foreground language model.

Let g^i be an N-gram of size i and $C_{\mathcal{M}}(g^i)$ denote the count of g^i in the model $\mathcal{M}$. First, we extract the counts of each N-grams in both the background ($\mathcal{BM}$) and the foreground corpora ($\mathcal{FM}$).

[1]http://chernobog.si.umich.edu/clair/anthology/index.cgi

$$
\begin{aligned}
M_{\mathcal{BM}} &= \sum_{g^i \in \{\mathcal{BM} \cup \mathcal{FM}\}} 1 \\
N_{\mathcal{BM}} &= \sum_{g^i \in \{\mathcal{BM} \cup \mathcal{FM}\}} C_{\mathcal{BM}}(g^i) \\
N_{\mathcal{FM}} &= \sum_{g^i \in \mathcal{FM}} C_{\mathcal{FM}}(g^i) \\
\hat{p}_{\mathcal{FM}}(g^i) &= C_{\mathcal{FM}}(g^i)/N_{\mathcal{FM}} \\
\hat{p}_{\mathcal{BM}}(g^i) &= (C_{\mathcal{BM}}(g^i)+1)/(M_{\mathcal{BM}}+N_{\mathcal{BM}})
\end{aligned}
$$

The last equation is also known as Laplace smoothing (Manning and Schutze, 2002) and handles the N-grams in the foreground corpus that have a 0 occurrence frequency in the background corpus. Next, we extract N-grams from the foreground corpus that have significant frequencies compared to the frequency of the same N-grams in the background model and its individual terms in the foreground model.

To measure how randomly a set of consecutive terms are forming an N-gram, Tomokiyo and Hurst (Tomokiyo and Hurst, 2003) use pointwise divergence. In particular, for an N-gram of size i, $g^i = (w_1 w_2 \cdots w_i)$,

$$
\delta_{g^i}(\mathcal{FM}^i \| \mathcal{FM}^1) = \hat{p}_{\mathcal{FM}}(g^i) \log\left(\frac{\hat{p}_{\mathcal{FM}}(g^i)}{\prod_{j=1}^{i} \hat{p}_{\mathcal{FM}}(w_j)}\right)
$$

This equation shows the extent to which the terms forming g^i have occurred together randomly. In other words, it indicates the extent of information that we lose by assuming independence of each word by applying the unigram model, instead of the N-gram model.

In addition, to measure how randomly a sequence of words appear in the foreground model with respect to the background model, we use pointwise divergence as well. Here, pointwise divergence defines how much information we lose by assuming that g^i is drawn from the background model instead of the foreground model:

$$
\delta_{g^i}(\mathcal{FM}^i \| \mathcal{BM}^i) = \hat{p}_{\mathcal{FM}}(g^i) \log\left(\frac{\hat{p}_{\mathcal{FM}}(g^i)}{\hat{p}_{\mathcal{BM}}(g^i)}\right)
$$

(Corley and Mihalcea, 2005) applied or utilized lexical based word overlap measures.
{overlap measures, word overlap, lexical based, utilized lexical}

Table 4: Example: citation sentence for W05-1203 written by D06-1621, and its extracted bigrams.

We set the criteria of choosing a sequence of words as significant to be whether it has positive pointwise divergence with respect to both the background model, and individual terms of the foreground model. In other words we extract all g^i from $\mathcal{FM}$ for which the both properties are positive:

$$
\begin{aligned}
\delta_{g^i}(\mathcal{FM}^i \| \mathcal{BM}^i) &> 0 \\
\delta_{g^i}(\mathcal{FM}^i \| \mathcal{FM}^1) &\geq 0
\end{aligned}
$$

The equality condition in the second equation is specifically set to handle unigrams, in which $\hat{p}_{\mathcal{FM}}(g^i) = \prod_{j=1}^{i} \hat{p}_{\mathcal{FM}}(w_j)$.

In order to handle the text corpora and building the language models, we have used the CMU-Cambridge Language Model toolkit (Clarkson and Rosenfeld, 1997). We use the set of citation sentences for each paper to build foreground language models. Furthermore, we employ this tool and make the background model using nearly 11,000 abstracts from AAN. Table 3 summarizes some of the statistics about the background data.

Once keyphrases (significant N-grams) of each sentence are extracted, we remove all N-grams in which more than half of the terms are stopwords. For instance, we remove all stopword unigrams, if any, and all bigrams with at least one stopword in them. For 3-grams and 4-grams we use a threshold of 2 and 3 stopwords respectively. After that, the set of remaining N-grams is used to represent each sentence and to build summaries. Table 4 shows an example of a citation sentence from D06-1621 citing W05-1203 (Corley and Mihalcea, 2005), and its extracted bigrams.

3.2 Sentence Selection

After extracting the set of keyphrases for each sentence, d_i, the sentence is represented using its set

of N-grams, denoted by D_i. Then, the goal is to pick sentences (sets) for each paper that cover more important and non-redundant keyphrases. Essentially, keyphrases that have been repeated in more sentences are more important and could represent more important nuggets. Therefore, sentences that contain more frequent keyphrases are more important. Based on this intuition we define the reward of building a summary comprising a set of keyphrases S as

$$f(S) = |S \cap A|$$

where A is the set of all keyphrases from sentences not in the summary.

The set function f has three main properties. First, it is non-negative. Second, it is monotone (i.e., For every set v we have $f(S + v) \geq f(S)$). Third, f is sub-modular. The submodularity means that for a set v and two sets $S \subseteq T$ we have

$$f(S + v) - f(S) \geq f(T + v) - f(T)$$

Intuitively, this property implies that adding a set v to S will increase the reward at least as much as it would to a larger set T. In the summarization setting, this means that adding a sentence to a smaller summary will increase the reward of the summary at least as much as adding it to a larger summary that subsumes it. The following theorem formalizes this and is followed by a proof.

Theorem 1 *The reward function f is submodular.*

Proof
We start by defining a gain function $\mathcal{G}$ of adding sentence (set) D_i to $\mathcal{S}_{k-1}$ where $\mathcal{S}_{k-1}$ is the set of keyphrases in a summary built using $k-1$ sentences, and D_i is a candidate sentence to be added:

$$\mathcal{G}(D_i, \mathcal{S}_{k-1}) = f(\mathcal{S}_{k-1} \cup D_i) - f(\mathcal{S}_{k-1})$$

Simple investigation through a Venn diagram proof shows that $\mathcal{G}$ can be re-written as

$$\mathcal{G}(D_i, \mathcal{S}_{k-1}) = |D_i \cap (\cup_{j \neq i} D_j) - \mathcal{S}_{k-1}|$$

Let's denote $D_i \cap (\cup_{j \neq i} D_j)$ by $\cap_i$. The following equations prove the theorem.

$$\begin{aligned}
&\mathcal{S}_{k-1} \subseteq \mathcal{S}_k \\
&\mathcal{S}'_{k-1} \supseteq \mathcal{S}'_k \\
&\cap_i \cap \mathcal{S}'_{k-1} \supseteq \cap_i \cap \mathcal{S}'_k \\
&\cap_i - \mathcal{S}_{k-1} \supseteq \cap_i - \mathcal{S}_k \\
&|\cap_i - \mathcal{S}_{k-1}| \geq |\cap_i - \mathcal{S}_k| \\
&\mathcal{G}(D_i, \mathcal{S}_{k-1}) \geq \mathcal{G}(D_i, \mathcal{S}_k) \\
&f(\mathcal{S}_{k-1} \cup D_i) - f(\mathcal{S}_{k-1}) \geq f(\mathcal{S}_k \cup D_i) - f(\mathcal{S}_k)
\end{aligned}$$

Here, $\mathcal{S}'_k$ is the set of all N-grams in the vocabulary that are not present in $\mathcal{S}_k$. The gain of adding a sentence, D_i, to an empty summary is a non-negative value.

$$\mathcal{G}(D_i, \mathcal{S}_0) = C \geq 0$$

By induction, we will get

$$\mathcal{G}(D_i, \mathcal{S}_0) \geq \mathcal{G}(D_i, \mathcal{S}_1) \geq \cdots \geq \mathcal{G}(D_i, \mathcal{S}_k) \geq 0$$

□

Theorem 1 implies the general case of submodularity:

$$\forall m, n, 0 \leq m \leq n \leq |D| \Rightarrow \mathcal{G}(D_i, \mathcal{S}_m) \geq \mathcal{G}(D_i, \mathcal{S}_n)$$

Maximizing this submodular function is an NP-hard problem (Khuller et al., 1999). A common way to solve this maximization problem is to start with an empty set, and in each iteration pick a set that maximizes the gain. It has been shown before in (Kulik et al., 2009) that if f is a submodular, nondecreasing set function and $f(\emptyset) = 0$, then such a greedy algorithm finds a set $\mathcal{S}$, whose gain is at least as high as $(1 - 1/e)$ of the best possible solution. Therefore, we can optimize the keyphrase coverage as described in Algorithm 1.

4 Experimental Setup

We use the annotated data described in Section 2. In summary, the annotation consisted of two parts: nugget extraction and nugget distribution analysis. Five annotators were employed to annotate the sentences in each of the 25 citation summaries and write down the nuggets (non-overlapping contributions) of the target paper. Then using these

Summary generated using bigram-based keyphrases	
ID	Sentence
P06-1048:1	Ziff-Davis Corpus Most previous work (Jing 2000; Knight and Marcu 2002; Riezler et al 2003; Nguyen et al 2004a; Turner and Charniak 2005; McDonald 2006) has relied on automatically constructed parallel corpora for training and evaluation purposes.
J05-4004:18	Between these two extremes, there has been a relatively modest amount of work in sentence simplification (Chandrasekar, Doran, and Bangalore 1996; Mahesh 1997; Carroll et al 1998; Grefenstette 1998; Jing 2000; Knight and Marcu 2002) and document compression (Daume III and Marcu 2002; Daume III and Marcu 2004; Zajic, Dorr, and Schwartz 2004) in which words, phrases, and sentences are selected in an extraction process.
A00-2024:9	The evaluation of sentence reduction (see (Jing, 2000) for details) used a corpus of 500 sentences and their reduced forms in human-written abstracts.
N03-1026:17	To overcome this problem, linguistic parsing and generation systems are used in the sentence condensation approaches of Knight and Marcu (2000) and Jing (2000).
P06-2019:5	Jing (2000) was perhaps the first to tackle the sentence compression problem.

Table 5: Bigram-based summary generated for A00-1043.

Algorithm 1 The greedy algorithm for summary generation

$k \leftarrow$ the number of sentences in the summary
$D_i \leftarrow$ keyphrases in d_i
$\mathcal{S} \leftarrow \emptyset$
for l = 1 to k **do**
  $s_l \leftarrow \arg\max_{D_i \in D} |D_i \cap (\cup_{j \neq i} D_j)|$
  $\mathcal{S} \leftarrow \mathcal{S} \cup s_l$
  for j = 1 to $|D|$ **do**
    $D_j \leftarrow D_j - s_l$
  end for
end for
return $\mathcal{S}$

nugget sets, each sentence was annotated with the nuggets it contains. This results in a sentence-fact matrix that helps with the evaluation of the summary. The summarization goal and the intuition behind the summarizing system is to select a few (5 in our experiments) sentences and cover as many nuggets as possible. Each sentence in a citation summary may contain 0 or more nuggets and not all nuggets are mentioned an equal number of times. Covering some nuggets (contributions) is therefore more important than others and should be weighted highly.

To capture this property, the pyramid score seems the best evaluation metric to use. We use the pyramid evaluation method (Nenkova and Passonneau, 2004) at the sentence level to evaluate the summary created for each set. We benefit from the list of annotated nuggets provided by the annotators as the ground truth of the summarization evaluation. These annotations give the list of nuggets covered by each sentence in each citation summary, which are equivalent to the *summarization content unit (SCU)* as described in (Nenkova and Passonneau, 2004).

The pyramid score for a summary is calculated as follows. Assume a pyramid that has n tiers, T_i, where tier $T_i > T_j$ if $i > j$ (i.e., T_i is not below T_j, and that if a nugget appears in more sentences, it falls in a higher tier.). Tier T_i contains nuggets that appeared in i sentences, and thus has weight i. Suppose $|T_i|$ shows the number of nuggets in tier T_i, and Q_i is the size of a subset of T_i whose members appear in the summary. Further suppose Q shows the sum of the weights of the facts that are covered by the summary. $Q = \sum_{i=1}^{n} i \times Q_i$. In addition, the optimal pyramid score for a summary with X facts, is

$$Max = \sum_{i=j+1}^{n} i \times |T_i| + j \times (X - \sum_{i=j+1}^{n} |T_i|)$$

where $j = \max_i(\sum_{t=i}^{n} |T_t| \geq X)$. The pyramid score for a summary is then calculated as follows.

$$P = \frac{Q}{Max}$$

This score ranges from 0 to 1, and a high score shows the summary contains more heavily weighted facts.

4.1 Baselines and Gold Standards

To evaluate the quality of the summaries generated by the greedy algorithm, we compare its pyramid score in each of the 25 citation summaries with those of a gold standard, a random summary, and four other methods. The gold standards are summaries created manually using 5 sentences. The 5 sentences are manually selected in a way to cover as many nuggets as possible with higher priority for the nuggets with higher frequencies. We also created random summaries using Mead (Radev et al., 2004). These summaries

are basically a random selection of 5 sentences from the pool of sentences in the citation summary. Generally we expect the summaries created by the greedy method to be significantly better than random ones.

In addition to the gold and random summaries, we also used 4 baseline state of the art summarizers: LexRank, the clustering C-RR and C-LexRank, and Maximal Marginal Relevance (MMR). LexRank (Erkan and Radev, 2004) works based on a random walk on the cosine similarity of sentences and prints out the most frequently visited sentences. Said differently, LexRank first builds a network in which nodes are sentences and edges are cosine similarity values. It then uses the eigenvalue centralities to find the most central sentences. For each set, the top 5 sentences on the list are chosen for the summary.

The clustering methods, C-RR and C-LexRank, work by clustering the cosine similarity network of sentences. In such a network, nodes are sentences and edges are cosine similarity of node pairs. Clustering would intuitively put nodes with similar nuggets in the same clusters as they are more similar to each other. The C-RR method as described in (Qazvinian and Radev, 2008) uses a round-robin fashion to pick sentences from each cluster, assuming that the clustering will put the sentences with similar facts into the same clusters. Unlike C-RR, C-LexRank uses LexRank to find the most salient sentences in each cluster, and prints out the most central nodes of each cluster as summary sentences.

Finally, MMR uses the full cosine similarity matrix and greedily chooses sentences that are the least similar to those already selected for the summary (Carbonell and Goldstein, 1998). In particular,

$$MMR = \arg \min_{d_i \in D-A} \Big[\max_{d_j \in A} Sim(d_i, d_j) \Big]$$

where A is the set of sentences in the summary, initially set to $A = \emptyset$. This method is different from ours in that it chooses the least similar sentence to the summary in each iteration.

4.2 Results and Discussion

As mentioned before, we use the text of the abstracts of all the papers in AAN as the background, and each citation set as a separate foreground corpus. For each citation set, we use the method described in Section 3.1 to extract significant N-grams of each sentence. We then use the keyphrase set representation of each sentence to build the summaries using Algorithm 1. For each of the 25 citation summaries, we build 4 different summaries using unigrams, bigrams, 3-grams, and 4-grams respectively. Table 5 shows a 5-sentence summary created using algorithm 1 for the paper A00-1043 (Jing, 2000).

The pyramid scores for different methods are reported in Figure 1 together with the scores of gold standards, manually created to cover as many nuggets as possible in 5 sentences, as well as summary evaluations of the 4 baseline methods described above. This Figure shows how the keyphrase based summarization method when employing N-grams of size 3 or smaller, outperforms other baseline systems significantly. More importantly, Figure 1 also indicates that this method shows more stable results and low variation in summary quality when keyphrases of size 3 or smaller are employed. In contrast, MMR shows high variation in summary qualities making summaries that obtain pyramid scores as low as 0.15.

Another important advantage of this method is that we do not need to calculate the cosine similarity of the pairs of sentences, which would add a running time of $O(|D|^2|V|)$ in the number of documents, $|D|$, and the size of the vocabulary $|V|$ to the algorithm.

5 Conclusion and Future Work

This paper presents a summarization methodology that employs keyphrase extraction to find important contributions of scientific articles. The summarization is based on citation sentences and picks sentences to cover nuggets (represented by keyphrases) or contributions of the target papers. In this setting the best summary would have as few sentences and at the same time as many nuggets as possible. In this work, we use pointwise KL-divergence to extract statistically significant N-grams and use them to represent nuggets. We then apply a new set function for the task of summarizing scientific articles. We have proved that this function is submodular and concluded that a

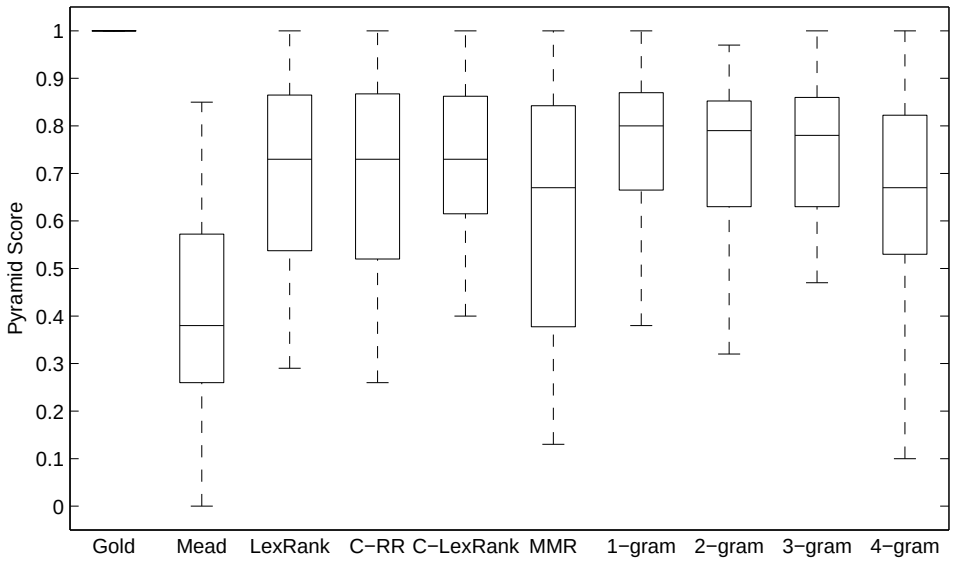

Figure 1: Evaluation Results (summaries with 5 sentences): The median pyramid score over 25 datasets using different methods.

greedy algorithm will result in a near-optimum set of covered nuggets using only 5 sentences. Our experiments in this paper confirm that the summaries created based on the presented algorithm are better than randomly generated summary, and also outperform other state of the art summarization methods in most cases. Moreover, we show how this method generates more stable summaries with lower variation in summary quality when N-grams of size 3 or smaller are employed.

A future direction for this work is to perform post-processing on the summaries and re-generate sentences that cover the extracted nuggets. However, the ultimate goal is to eventually develop systems that can produce summaries of entire research areas, summaries that will enable researchers to easily and quickly switch between fields of research.

One future study that will help us generate better summaries is to understand how nuggets are generated by authors. In fact, modeling the nugget coverage behavior of paper authors will help us identify more important nuggets and discover some aspects of the paper that would otherwise be too difficult by just reading the paper itself.

6 Acknowledgements

This work is in part supported by the National Science Foundation grant “iOPENER: A Flexible Framework to Support Rapid Learning in Unfamiliar Research Domains”, jointly awarded to University of Michigan and University of Maryland as IIS 0705832, and in part by the NIH Grant U54 DA021519 to the National Center for Integrative Biomedical Informatics.

Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the supporters.

References

Bradshaw, Shannon. 2002. *Reference Directed Indexing: Indexing Scientific Literature in the Context of Its Use*. Ph.D. thesis, Northwestern University.

Bradshaw, Shannon. 2003. Reference directed indexing: Redeeming relevance for subject search in citation indexes. In *Proceedings of the 7th European*

Conference on Research and Advanced Technology for Digital Libraries.

Carbonell, Jaime G. and Jade Goldstein. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *SIGIR'98*, pages 335–336.

Clarkson, PR and R Rosenfeld. 1997. Statistical language modeling using the cmu-cambridge toolkit. *Proceedings ESCA Eurospeech*, 47:45–148.

Elkiss, Aaron, Siwei Shen, Anthony Fader, Güneş Erkan, David States, and Dragomir R. Radev. 2008. Blind men and elephants: What do citation summaries tell us about a research article? *Journal of the American Society for Information Science and Technology*, 59(1):51–62.

Erkan, Güneş and Dragomir R. Radev. 2004. Lexrank: Graph-based centrality as salience in text summarization. *Journal of Artificial Intelligence Research*.

Jing, Hongyan. 2000. Sentence reduction for automatic text summarization. In *Proceedings of the sixth conference on Applied natural language processing*, pages 310–315, Morristown, NJ, USA. Association for Computational Linguistics.

Kan, Min-Yen, Judith L. Klavans, and Kathleen R. McKeown. 2002. Using the Annotated Bibliography as a Resource for Indicative Summarization. In *Proceedings of LREC 2002*, Las Palmas, Spain.

Khuller, Samir, Anna Moss, and Joseph (Seffi) Naor. 1999. The budgeted maximum coverage problem. *Inf. Process. Lett.*, 70(1):39–45.

Kulik, Ariel, Hadas Shachnai, and Tami Tamir. 2009. Maximizing submodular set functions subject to multiple linear constraints. In *SODA '09*, pages 545–554.

Kupiec, Julian, Jan Pedersen, and Francine Chen. 1995. A trainable document summarizer. In *SIGIR '95*, pages 68–73, New York, NY, USA. ACM.

Manning, Christopher D. and Hirich Schutze. 2002. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, Massachusetts, London, England.

Mei, Qiaozhu and ChengXiang Zhai. 2008. Generating impact-based summaries for scientific literature. In *Proceedings of ACL '08*, pages 816–824.

Mohammad, Saif, Bonnie Dorr, Melissa Egan, Ahmed Hassan, Pradeep Muthukrishan, Vahed Qazvinian, Dragomir Radev, and David Zajic. 2009. Using citations to generate surveys of scientific paradigms. In *NAACL 2009*, pages 584–592, June.

Nanba, Hidetsugu and Manabu Okumura. 1999. Towards multi-paper summarization using reference information. In *IJCAI1999*, pages 926–931.

Nanba, Hidetsugu, Noriko Kando, and Manabu Okumura. 2004. Classification of research papers using citation links and citation types: Towards automatic review article generation. In *Proceedings of the 11th SIG Classification Research Workshop*, pages 117–134, Chicago, USA.

Nenkova, Ani and Rebecca Passonneau. 2004. Evaluating content selection in summarization: The pyramid method. *Proceedings of the HLT-NAACL conference*.

Qazvinian, Vahed and Dragomir R. Radev. 2008. Scientific paper summarization using citation summary networks. In *COLING 2008*, Manchester, UK.

Radev, Dragomir, Timothy Allison, Sasha Blair-Goldensohn, John Blitzer, Arda Çelebi, Stanko Dimitrov, Elliott Drabek, Ali Hakim, Wai Lam, Danyu Liu, Jahna Otterbacher, Hong Qi, Horacio Saggion, Simone Teufel, Michael Topper, Adam Winkel, and Zhu Zhang. 2004. MEAD - a platform for multidocument multilingual text summarization. In *LREC 2004*, Lisbon, Portugal, May.

Radev, Dragomir R., Pradeep Muthukrishnan, and Vahed Qazvinian. 2009. The ACL anthology network corpus. In *ACL workshop on Natural Language Processing and Information Retrieval for Digital Libraries*.

Siddharthan, Advaith and Simone Teufel. 2007. Whose idea was this, and why does it matter? attributing scientific work to citations. In *Proceedings of NAACL/HLT-07*.

Teufel, Simone. 2005. Argumentative Zoning for Improved Citation Indexing. *Computing Attitude and Affect in Text: Theory and Applications*, pages 159–170.

Tomokiyo, Takashi and Matthew Hurst. 2003. A language model approach to keyphrase extraction. In *Proceedings of the ACL 2003 workshop on Multiword expressions*, pages 33–40.