Surveyor: Generating Scientific Historical Notes

Abstract

We describe Surveyor, a system that automatically generates surveys of collections of papers that represent scientific domains. Surveyor extracts the main contributions of the underlying articles by using heterogeneous sources: source texts and citations. Using Surveyor, we generate summaries of 10 topics in Natural Language Processing corresponding to 10 chapters from the Jurafsky and Martin textbook *Speech and Language Processing* (Jurafsky and Martin, 2008). We evaluate the generated summaries by comparing them to end-of-chapter summaries and historical notes from the textbook and show that using heterogeneous sources results in higher Rouge scores than state-of-art summarization systems. Finally, we present our experiments on a benchmark dataset using nugget-based pyramid score. We show that our proposed model can outperform previous survey generation systems (such as C-LexRank) by 9.2% when citation nuggets are employed.

1 Introduction

Researchers and scholars often face the problem of keeping up with the ever increasing number of publications in their fields of research. In addition, research is increasingly becoming inter-disciplinary, bridging different areas and forcing researchers to familiarize themselves with new areas. For instance, cancer researchers often have to quickly move into a new area in their research; a pathologist may want to learn about new medical devices or drug developments; business researchers are interested in understanding user behavior in an online community; and social scientists may be interested in learning new computational models that explain certain social phenomena. Inter-disciplinary review panels and funding agencies often need to make decisions on proposals from a wide range of newly emerging areas. Thus they have to learn about the development of ideas in a new discipline and be able to relate their expertise to the proposals.

In this paper, we present Surveyor, a summary generation system that addresses such needs by generating surveys of key developments on a research topic. The solution that we propose uses different sources of information (i.e., source texts and citations) and exploits the citation network to produce summaries that compete expert-written surveys.

Previous work has noted the difference between conventional multi-document summarization and summarizing scientific literature (Mohammad et al., 2009). In the case of multi-document summarization, the goal is to produce a readable presentation of multiple documents, whereas in the case of technical survey creation, the goal is to convey the key features and basic underpinnings of a particular field, temporal developments, important contributions, emergence of sub-fields, and basic definitions and examples that enable rapid understanding of a field by non-experts.

One example of expert-written surveys is the set of end-of-chapter summaries and "historical notes" that appear at the end of chapters in the *Speech and Language Processing* textbook (Jurafsky and Martin, 2008). Each summary or historical note is about a sub-field in Natural Language Processing (NLP), and includes information about the background, early and recent developments, state-of-the-art results, etc. Table 1 shows parts of the historical notes that Jurafsky and Martin wrote for "Machine Translation." The example shows that this survey includes various information for non-expert readers including some history, early developments, toolkits, evaluations and additional references and tutorials for further reading.

The goal of our paper is to present a framework that generates summaries similar to the one in Table 1. After a review of related work in Section 2, we

Historical Notes: Machine Translation	
history	*Work on models of the process and goals of translation goes back at least to Saint Jerome in the fourth century (Kelley, 1979)...*
early work	*... At the same time, the IBM group, drawing directly on algorithms for speech recognition (many of which had themselves been developed originally at IBM!) proposed the Candide system, based on the IBM statistical models we have described (Brown et al., 1990, 1993) ...*
tools	*... Progress was made hugely easier by the development of publicly-available toolkits, particularly tools extended from the EGYPT toolkit developed by the Statistical Machine Translation team in during the summer 1999 research workshop at the Center for Language and Speech Processing at the Johns Hopkins University. These include the GIZA++ aligner, developed by Franz Joseph Och by extending the GIZA toolkit (Och and Ney, 2003), which implements IBM models 1-5 as well as the HMM alignment model ...*
evaluations	*... These included the use of doze and Shannon tasks to measure intelligibility as well as a metric of edit distance from a human translation, the intuition that underlies all modern automatic evaluation metrics like BLEU ...*
other resources	*... Nirenburg et al. (2002) is a comprehensive collection of classic readings in MT. Knight (1999b) is an excellent tutorial introduction to statistical MT ...*

Table 1: Part of the historical note in (Jurafsky and Martin, 2008) signifying the history, early and late developments and evaluation in "machine translation"

present our data preparation, including scanning and parsing citations in (Jurafsky and Martin, 2008) and the ACL Anthology Network (Radev et al., 2009), which is used as the source for summary generation in Section 3. We propose a new approach that repurposes both citations and source text of papers and exploits the citation graph to build a survey in Section 4. Finally, Section 5 presents our experiments and results on the Jurafsky and Martin textbook as well as a benchmark dataset from (Mohammad et al., 2009) and Section 6 concludes the paper.

2 Related Work

In this section, we first summarize previous work on citation analysis and summarization, then we review graph-based summarization systems.

2.1 Citation Analysis

Citation patterns and collaboration networks have been studied before (Newman, 2001; Leskovec et al., 2005). Bradshaw (2003) introduces "Reference Directed Indexing" to improve the results of a search engine by using citations. Nanba and Okumura (1999) report that co-citation implies similarities by showing that the textual similarity of co-cited papers is proportional to the proximity of their citations in the citing article (Nanba et al., 2004b). Previous work uses citation role classification for survey generation (Teufel et al., 2006; Nanba and Okumura, 1999). Using 160 pre-defined phrase-based rules, they analyze citation sentences and automatically categorize citations into three types: (1) theories and methods; (2) problems or gaps; (3) neither (Nanba et al., 2004b; Nanba et al., 2004a).

Previous work has used abstracts of scientific articles to produce summaries (Kupiec et al., 1995). However, other work has shown that citation sentences are as important in understanding the main contributions of a paper. Elkiss et al. (2008) perform a large-scale study on citations and their importance. Their experiments on more than $2,000$ articles from the free PubMedCentral repository suggest that the average cosine between sentences in the set of citations to an article is consistently higher than that of its abstract.

Kan et al. (2002) use annotated bibliographies for summarization and suggest that summaries should include metadata and critical document features as well as the prominent content-based features. Siddharthan and Teufel (2007) describe a new reference task and show high annotator agreement as well as an improvement on the performance of *argumentative zoning* (Teufel, 2005). In argumentative zoning—a rhetorical classification task—seven classes (Own, Other, Background, Textual, Aim, Basis, and Contrast) are used to label sentences according to their roles in the author's argument.

Previously, Qazvinian and Radev (2008) show that the set of citations to a paper describe the most important contributions of a given paper. Qazvinian et al. (2010) propose a keyphrase extraction system to extract such contributions and produce a summary that covers them. Mohammad et al. (2009) extend citation summarization to a set of papers that repre-

sent a topic and show that citations can be employed to effectively produce surveys for two Natural Language Processing tasks: Question Answering and Dependency Parsing.

In the Elsevier's Grand Challenge 2009, a new research tool Citation-Sensitive In-Browser Summarizer (CSIBS) (Wan et al., 2010) was designed to facilitate biomedical researchers reviewing academic literature. In the scenario of one reading a document, a large number of citations point to a set of cited documents. Based on user requirements analysis and citation context analysis, the CSIBS will recommend the most relevant articles to the user.

2.2 Graph-based summarization systems

As a representative of graph-based methods applied to summarization, LexRank (Erkan and Radev, 2004) constructs a graph whose vertices are sentences from all the documents in a cluster. The graph is characterized by a sentence connectivity matrix representing the Markov transition probabilities among vertices. Sentences of high centralities are then selected to form the summary C-LexRank (Qazvinian and Radev, 2008) extends the framework by incorporating community clustering to address the need of covering different aspects of contributions in a scientific work. We use both methods as baselines in our experiment.

Motivated by the similar idea of applying PageRank and HITS (Kleinberg, 1999) on graphs of sentences, Mihalcea and Tarau (2004) present TextRank, a system for keyword extraction and sentence extraction, and successfully apply it to producing extractive summaries. The system is proved scalable to multi-document summarization tasks, and also language-independent (Mihalcea, 2005).

More recent work has integrated link analysis and other techniques as re-ranking process to improve the effectiveness of summarization based on graph-based ranking. Wan and Yang (2006) incorporate *information richness* and *information novelty* into the criteria of selecting important sentences. These two parameters are determined by a sentence affinity graph reflecting the semantic relationships between sentences. They also distinguish between intra-document and inter-document links, biasing the latter for information richness computation.

Another optimization ClusterCMRW (and ClusterHITS) proposed in (Wan and Yang, 2008) assumes that a given document set covers a few topic themes or subtopics that are of different degrees of importance. The idea of clustering sentences according to subtopics is comparable to C-LexRank. Designed for summarizing scientific contributions, C-LexRank looks for a comprehensive coverage of each subtopic or contribution aspect, while ClusterCMRW (and ClusterHITS) focus at ranking on the cluster level, so that sentence centralities are scaled by the centralities of the clusters in which they belong.

3 Datasets

In this section, we first describe the ACL Anthology Network, which is used as the source dataset for generating system surveys. We then explain our gold standard preparation from the Jurafsky and Martin textbook.

3.1 The ACL Anthology Network

The ACL Anthology[1] includes all papers published by ACL and related organizations as well as the Computational Linguistics journal over a period of four decades. Radev et al. (2009) have further processed this Anthology to produce the the ACL Anthology Network (AAN)[2]. The AAN includes more than $16,000$ papers, each distinguished with a unique ACL ID, together with their full-texts, abstracts, and citation information. It also includes other valuable meta-data such as author affiliations, citation and collaboration networks, and various centrality measures (Radev et al., 2009; Joseph and Radev, 2007). In our experiments, we generate a set of automatic summaries using the papers in AAN.

3.2 Gold Standard Preparation

We use 2 sets of gold standards both extracted from the Jurafsky and Martin textbook[3] *Speech and Language Processing* (Jurafsky and Martin, 2008): end-of-chapter summaries and the historical notes.

3.2.1 End-of-chapter Summaries

We were fortunate to obtain the end of chapter summaries in the JM book in text format. Each

[1] http://www.aclweb.org/anthology-new/

[2] http://clair.si.umich.edu/clair/anthology/

[3] we use the shorthand "JM book" in the rest of this paper

summary is generally a few paragraphs long and explains the main points discussed in the chapter. We will refer to these gold standards as **chapter summaries**.

3.2.2 Historical Notes

We also use the **historical notes** at the end of each chapter in the JM book as the second set of gold standards. Each historical note, corresponding to one chapter, is generally 1-2 pages long and summarizes the history, early developments and the state-of-art methods in each NLP topic.

In order to prepare this gold standard, we first scanned the historical notes of the chapters as well as the references in the JM book. Next, we used a commercial OCR tool to convert the scanned files to plain text. We further processed the OCR output by removing end-of-line hyphens and fixing sentence fragments and line breaks.[4] Cleaning-up references included identifying entry boundaries and combining multiple lines corresponding to one entry.

We used the extracted references and citations in each historical note to extract the set of papers that are cited by (Jurafsky and Martin, 2008) and are part of AAN. We use these papers as the seed source papers to generate automatic summaries. To extract the list of AAN papers that are cited in each historical note, we first map each reference in the JM book to an AAN paper. First, for each reference we represent it by a vector of metadata that consists of the author names, title (stop words removed), canonical name of the venue, and publication year. We then compare these vectors with with AAN metadata and find the closest match by computing the minimum edit distance of corresponding metadata vectors when the publication dates agree. Finally, we manually verify the output of the above procedure and correct mismatches.

4 Approach

Previous work on scientific survey generation have compared surveys that are generated from different sources such as citations and source paper texts (Qazvinian and Radev, 2008; Mei and Zhai, 2008; Mohammad et al., 2009). However, none of these approaches combine these heterogeneous information sources to produce automatic surveys.

In our approach, we investigate the usefulness of combining different information sources and producing summaries that are both affected by source paper text and citation information. For a set of papers in the same scientific topic, we extract survey worthy sentences from the source texts that cover contributions recognized by other scholars in citations, and extract citations that cover contributions that are recognized by the authors in the source text.

In our algorithm, we model the set of papers in a scientific topic t as a bi-partite graph, $\mathcal{B}$ with a left and a right component ($\mathcal{B}_L$, $\mathcal{B}_R$). Each node in $\mathcal{B}_L$ is a citation sentence to one or more papers in t extracted from AAN, and each node in $\mathcal{B}_R$, represents a sentence extracted from the source text of a paper in t. We construct the edges in $\mathcal{B}$ by connecting each citing sentence to all the source sentences in the papers it cites. Each edge in $\mathcal{B}$ is assigned a weight equal to the cosine similarity of the TF-IDF term vectors of the two sentences it connects. Figure 1 illustrates part of the bi-partite graph for built for the "Part-of-Speech Tagging" chapter in the JM book.

To build the summaries we are interested in citations and source sentences that cover important contributions in the given scientific topics. Intuitively, contributions that both the paper authors and other scholars recognize as significant are important and should be extracted. Surveyor extracts citations that cover important contributions mentioned in the source papers as well as source sentences that discuss important factoids recognized by others in citations.

4.1 Ranking

The inherent duality in the source papers and citations suggests that the problem could be addressed by applying the HITS algorithm (Kleinberg, 1999) to iteratively assign hub and authority scores to citations and source sentences respectively. The induction process is as follows. Each citation sentence $c \in \mathcal{B}_L$ is associated with a hub score h_c, and each source sentence $s \in \mathcal{B}_R$ is associated with an au-

[4] Parsing the bibliographies from the OCR output is more challenging than historical notes because of the smaller fonts and frequent out-of-vocabulary words such as author names. However, OCR errors are tolerated in bibliographies since we use minimum edit distance to find the corresponding papers in AAN.

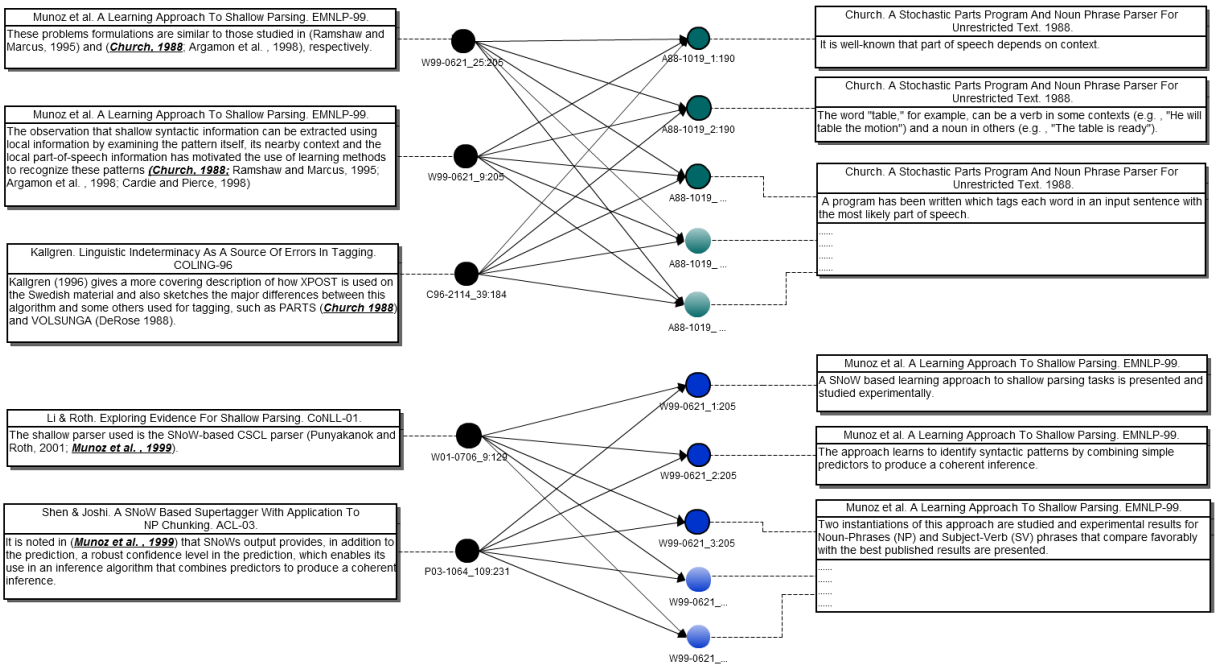

Figure 1: A mini-model of the bi-partite graph for Chapter 5 (Part-of-Speech Tagging)

thority score a_s. These scores are initialized with a value of 1.0. Hub and authority scores are iteratively updated using the following equations.

$$a_s^{(i+1)} = \sum_{c \in nei(s)} \frac{h_c^{(i)}}{H^{(i)}} \quad (1)$$

$$h_c^{(i+1)} = \sum_{s \in nei(c)} \frac{a_s^{(i)}}{A^{(i)}} \quad (2)$$

where a source sentence s is in a citation sentence, c's neighborhood ($s \in nei(c)$) if there is an edge between s and c in $\mathcal{B}$ (c cites the paper that contains s), and their cosine similarity is greater than a threshold (i.e., $cos(s,c) > \theta$). Here, $H^{(i)}$ and $A^{(i)}$ are normalization factors:

$$H^{(i)} = (\sum_{c \in \mathcal{B}_L} {h_c^{(i)}}^2)^{1/2} \quad (3)$$

$$A^{(i)} = (\sum_{s \in \mathcal{B}_R} {a_s^{(i)}}^2)^{1/2} \quad (4)$$

In our experiments, we set $\theta = 0.1$. This ranking gives us top authorities (source sentences) and top hubs (citations) with which we build two different summaries: **HITS$_{\text{src}}$** and **HITS$_{\text{cit}}$**. Although these summaries are built from different sources (i.e., source papers and citations) they are affected by each other. In other words, the scores and thus extraction of top citations affects the extraction of top source sentences and vice versa.

4.2 Adding Weights

In previous section, we described the basic version of our system in which the edges are considered as binary connections (if the cosine similarity is above a threshold). We would like to investigate the effect of similarity on sentence extraction. In other words, instead of applying a threshold we use the actual edge weights and modify Equations 1, 2 as follows.

$$a_s^{(i+1)} = \sum_{c \in nei(s)} \frac{w_{cs} \cdot h_c^{(i)}}{H^{(i)}} \quad (5)$$

$$h_c^{(i+1)} = \sum_{s \in nei(c)} \frac{w_{sc} \cdot a_s^{(i)}}{A^{(i)}} \quad (6)$$

Chapter	src	cit	$\lvert\mathcal{B}_L\rvert$	$\lvert\mathcal{B}_R\rvert$	$E_\mathcal{B}$
Words and Transducers	14	255	489	3,484	202,940
N-grams	5	73	97	2,083	32,690
Part-of-Speech Tagging	16	657	1,261	3,385	344,886
Hidden Markov and Maximum Entropy Models	2	432	659	525	187,905
Phonetics	1	12	81	216	17,496
Speech Synthesis	4	54	126	920	29,357
Automatic Speech Recognition	2	27	103	401	21,566
Speech Recognition: Advanced Topics	7	189	445	2,007	96,467
Syntactic Parsing	4	131	246	763	63,673
Dialog and Conversational Agents	11	170	368	3,745	114,281

Table 2: List of chapter historical notes used in our experiments together with the number of source papers extracted from historical notes (src), the number of citing papers extracted from AAN (cit), size of the left ($\mathcal{B}_L$) and right ($\mathcal{B}_R$) components in the bi-partite graph, and number of edges in the graph ($E_\mathcal{B}$).

where w_{sc} is the is the edge weight between vertices s and c, calculated as the TF-IDF based cosine similarity between their corresponding sentences.

Intuitively, this modification will take into account the similarity of sentence with its neighbors rather than the number of connections, and would result in summaries that contain more *lexically* salient sentences. The weighted ranking gives us top authorities (source sentences) and top hubs (citations) with which we build two different summaries: **HITS$_{\text{src}}$ with weights** and **HITS$_{\text{cit}}$with weights**.

4.3 Citation Bias

The downside of the current HITS-based sentence extraction is that it assumes equal importance for the papers in a given topic. However, contributions from highly cited papers are intuitively more important. To address this issue, we propose an improvement inspired by (Mei et al., 2010) and modify equations 1, 2 to include a prior distribution of prestige.

$$a_s^{(i+1)} = (1-\lambda)\cdot p^*(s) + \lambda \cdot \sum_{c\in nei(s)} \frac{w_{cs}\cdot h_c^{(i)}}{H^{(i)}} \quad (7)$$

$$h_c^{(i+1)} = (1-\lambda)\cdot p^*(c) + \lambda \cdot \sum_{s\in nei(c)} \frac{w_{sc}\cdot a_s^{(i)}}{A^{(i)}} \quad (8)$$

Here, $p^*(v)$ is a distribution which represents the prior preference of vertex v. When $p^*(v)$ is uniform, the left component is similar to the random jumping probabilities in PageRank. Other possible choices for $p^*(v)$ include a topic sensitive distribution, inspired by personalized jumping in personalized PageRank (Haveliwala, 2002; Haveliwala, 2003). In Equations 7, 8 λ obtains a value between 0 and 1. When $\lambda = 1$, Equations 7, 8 lead to the standard HITS algorithm. In our experiments, we set $\lambda = 0.75$.

The prior distribution allows us to favor citation sentences that are from more impactful papers. Therefore we define the prior distributions as the normalized citation frequency of the paper

$$p^*(v) = \frac{C_v + 1}{\sum_{v\in\mathcal{B}} C_v + |\mathcal{B}|} \quad (9)$$

where C_v is the number of citations to the paper that contains sentence v. Equations 7, 8 give us top authorities (source sentences) and top hubs (citations) with which we build two different summaries: **HITS$_{\text{src}}$ with weights/priors** and **HITS$_{\text{cit}}$with weights/priors**.

5 Experiments

Using the procedure described in section 3.2.2, we extract the list of source papers from 10 chapters' historical notes in the JM book. For each chapter, the papers cited in its historical note are used as the source papers (**src**) and the set of AAN papers that cite them are used as citing papers (**cit**). Table 2 summarizes the list of chapter historical notes used in our experiments together with the number of source papers, citing papers extracted from AAN, the size of the left ($\mathcal{B}_L$) and right ($\mathcal{B}_R$) components in the bi-partite graph, and number of edges in the graph ($E_\mathcal{B}$).

For each chapter we generate 2×2 summaries using the **cit** and **src** papers with a length equal to the length of chapter's **chapter summaries** and that of

	System Performance: Rouge-1 Gold Standard: Historical Notes				
Method	**src**	95% C.I.	**cit**	95% C.I.	Mean
LexRank	0.150	[0.110, 0.190]	0.212	[0.189, 0.235]	0.181
C-LexRank	0.183	[0.147, 0.220]	0.187	[0.158, 0.217]	0.185
HITS	0.202	[0.162, 0.243]	0.152	[0.120, 0.185]	0.177
HITS with weights	**0.216**	**[0.195, 0.237]**	0.200	[0.178, 0.222]	0.208
HITS with weights/priors	0.204	[0.187, 0.221]	**0.215**	**[0.181, 0.249]**	**0.209**

Table 3: Average Rouge-1 scores of automatic surveys of the 10 chapters listed in Table 2 evaluated using historical notes as reference (C.I.: Confidence Interval).

chapter's **historical notes**. We evaluate these summaries using Rouge (Lin, 2004), and compare them with two state-of-the-art methods in scientific survey generation: LexRank and C-LexRank.

5.1 Baseline Methods

5.1.1 LexRank

LexRank (Erkan and Radev, 2004) works by first building a graph of all the documents (D_i) in a cluster. The edges between corresponding nodes (d_i) represent the cosine similarity between them if the cosine value is above a threshold (0.10 following (Erkan and Radev, 2004)). Once the network is built, the system finds the most central sentences by performing a random walk on the graph.

$$p(d_j) = (1-\lambda)\frac{1}{|D|} + \lambda \sum_{d_i} p(d_i) P(d_i \rightarrow d_j) \quad (10)$$

5.1.2 C-LexRank

C-LexRank is a clustering-based summarization system that is proposed by (Qazvinian and Radev, 2008) to summarize different scientific perspectives. create a full connected network in which nodes are sentences and edges are cosine similarities. To create summaries, C-LexRank constructs a fully connected network in which vertices are sentences and edges are cosine similarities calculated using the TF-IDF vectors of citation sentences. It then employs a hierarchical agglomeration clustering algorithm proposed by (Clauset et al., 2004) to find communities of sentences that discuss the same scientific contributions.

Once the graph is clustered and communities are formed, Qazvinian and Radev (2008) extract sentences from different clusters to build a summary.

Part of the automatic summary	
early developments	*During the early stages of the Penn Treebank project, the initial automatic POS assignment was provided by PARTS (Church 1988), a stochastic algorithm developed at AT&T Bell Labs.*
methods	*As shown by Klein and Manning (2002, 2004), the extension to inducing trees for words instead of p-o-s tags is rather straight-forward since there exist several unsupervised part-of-speech taggers with high accuracy, which can be combined with unsupervised parsing (see e.g. Schutze 1996; Clark 2000).*
ambiguity problem	*Jardino and Adda (1994), Schutze (1997) and Clark (2000) have attempted to address the ambiguity problem to a certain extent.*

Table 5: Part of the automatic survey generated using **HITS with weights** for "part-of-speech tagging" signifying early work, state-of-the-art, etc. (The labels are manually extracted for better illustration of the summary quality, and are not a by-product of the system)

They start with the largest cluster and extract sentences using LexRank within each cluster. In other words, for each cluster Ω_i they make a lexical network of *the sentences in that cluster* (N_i). LexRank extracts the most central sentences in N_i as salient sentences of Ω_i to include in the main summary. For each cluster Ω_i, the most salient sentence of Ω_i is extracted until the summary length limit is reached. The cluster selection is in order of decreasing size.

5.2 Results and Discussion

Table 3 lists the average Rouge-1 scores of different automatic summaries with each chapter's **historical notes** chosen as the gold standard and its length as the automatic summary length. Similarly, Table 4 summarizes the average Rouge-1 scores of different system summaries when **chapter summaries** are used as reference.

Both of these tables show that in general the HITS

	System Performance: Rouge-1 Gold Standard: Chapter Summaries				
Method	**src**	95% C.I.	**cit**	95% C.I.	Mean
LexRank	0.205	[0.141, 0.269]	0.232	[0.203, 0.260]	0.218
C-LexRank	0.188	[0.129, 0.246]	0.198	[0.140, 0.256]	0.193
HITS	0.233	[0.191, 0.274]	0.161	[0.122, 0.200]	0.197
HITS with weights	**0.242**	**[0.215, 0.268]**	0.222	[0.183, 0.260]	0.232
HITS with weights/priors	0.235	[0.198, 0.271]	**0.241**	**[0.198, 0.284]**	**0.238**

Table 4: Average Rouge-1 scores of automatic surveys of the 10 chapters listed in Table 2 evaluated using chapter summaries as reference (C.I.: Confidence Interval).

method that employs weights on graph edges leads to significantly better results than other methods both when the summaries are generated from citations (cit) or source texts (src). Moreover, these tables suggest that the HITS method when employing weights and priors outperforms the state-of-the-art methods and baselines when summaries are generated using citations (cit). Table 5 shows part of the automatic survey generated using **HITS with weights** for "part-of-speech tagging" signifying some early work, state-of-the-art, etc. For better illustration of the quality of this survey, we have manually labeled each sentence with its role (i.e., early developments, methods, etc.)

We repeat the same experiments using Rouge-L. Figure 2 summarizes the average Rouge-L score for automatic summaries generated using source texts (src) and citations (cit). This figure confirms that Rouge-L results follow a similar pattern as Rouge-1. The results in Figure 2 and Tables 4, 3 also suggest that surveys generated using citations are consistently better that those generated from source texts in LexRank and C-LexRank. However, when the two summaries are generated using both sources affecting each other in a bi-partite graph, summaries from source and citations obtain similar qualities on average.

In summary, we observe that using semantic relatedness and adding weights to the bi-partite citation graph increases the quality of the produced summaries. One explanation is that weights enforce citation sentences (source sentences) to obtain high scores only when they are connected to important source sentences (citation sentences) that are also semantically similar similar to them.

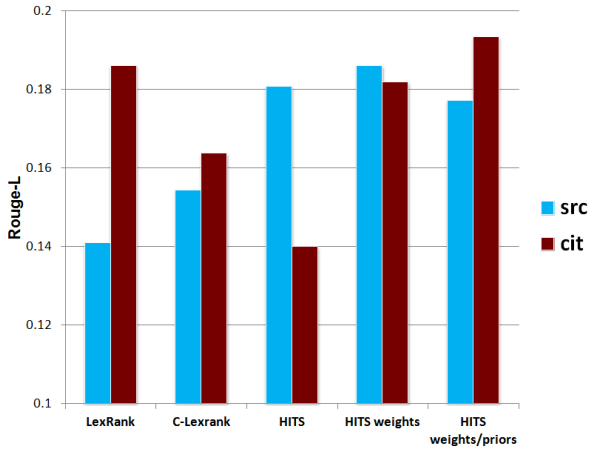

Figure 2: Average Rouge-L scores of automatic surveys of the 10 chapters listed in Table 2 using chapter summaries and historical notes as reference

5.3 Nugget-based Evaluations

In addition to Rouge, we evaluate the quality of the automatic summaries using the *pyramid score*. This evaluation metric relies on human judgments and manual nugget annotations (Lin and Demner-Fushman, 2006; Nenkova and Passonneau, 2004; Hildebrandt et al., 2004; Voorhees, 2003). In pyramid evaluation, different factoids (Qazvinian and Radev, 2011) obtain different weights, and the quality of a summary is measured using the F-measure calculated from its factoid recall and precision values.

In order to evaluate the performance of the proposed graph-based algorithm with respect to human extracted nuggets, we use the QA dataset from (Mohammad et al., 2009). In their work, Mohammad et al. (2009) performed their experiments on a set of papers in the research area of Question Answering (QA). They selected the papers in each corpus by

System Performance: Pyramid F-measure		
	Nuggets:	
System	QA–CT	QA–AB
Random	0.321	0.395
LexRank	0.295	0.320
C-LexRank	0.434	0.388
HITS	0.421	0.347
HITS with weights	**0.474**	**0.462**
HITS with weights/priors	0.149	0.101

Table 6: Pyramid F-measure scores of automatic surveys of QA data. The surveys are evaluated using nuggets drawn from QA citation texts (QA–CT) and QA abstracts (QA–AB).

matching the phrases "Question Answering" in the title and the content of AAN papers. The QA dataset contains 10 papers cited in 146 sentences from 62 papers in AAN.

Mohammad et al. (2009) created a set of gold standards for the QA data from citation texts and abstracts, respectively. They asked three human judges to identify important nuggets of information worth including in a survey from QA citations and QA abstract. More particularly, they instruct annotators to extract prioritized lists of 2–8 nuggets from abstracts and citations of each paper. These lists are them merged to build the pyramid model that can be used to evaluate automatically generated summaries.

We use this dataset to evaluate the performance of the HITS algorithm using the nugget-based pyramid method. We obtain the citations and the nuggets for the QA papers, which were extracted from AAN's 2009 release. We also obtained the source text of the QA papers from the most recent AAN papers to build the graph. Since the nuggets that were extracted by Mohammad et al. (2009) were from citations and abstracts, and that our source papers may not be identical to the original set, we only evaluate automatic summaries that are generated using citations in this section, and not the full-text.

When evaluated on this data, the HITS algorithm outperforms LexRank and C-LexRank. Table 6 shows that the summaries generated using the HITS model that employs weights on network edges produces higher quality summaries that LexRank and C-LexRank as well as the Random summarizer, which pick sentences from citation sets randomly.

6 Conclusion and Future Work

In this paper we present a framework based on the HITS algorithm that employs heterogeneous information (i.e., citations and source texts) to generate surveys of scientific paradigms. Using both Rouge and nugget-based evaluations, we show that our proposed system, Surveyor, generates summaries that have higher quality than the state-of-the-art methods when compared with end of chapter summaries and historical notes in Jurafsky and Martin NLP textbook.

In our work, we have used Jurafksy and Martin's end of chapter summaries as the gold standard written by written experts. We believe that the area of text summarization, and especially summarizing scholarly work can benefit from a wide range of expert written summaries that are produced more naturally, outside the context of multi-document summarization experiments. Other examples of such a gold standard source include "further reading" sections in the leading Information Retrieval textbook (Manning et al., 2008), or survey papers published occasionally in journals such as Computational Linguistics.

One of the authors of this paper organized an NLP seminar previously. As part of the seminar, the students in the class took turns to present surveys of specific topics in NLP and Information Retrieval (IR) and wrote chapter-length surveys of their topics. In future work, we plan to make use of the surveys written by NLP students as gold standard in evaluations. Compared to the chapters from JM book, these topics are more specific and close to the latest development in NLP and IR. Examples include Sentiment and Polarity Extraction, Science Maps, Spectral graph-based methods for NLP, Information Diffusion In Graphs, Financial Networks and Query Expansion.

In current work, we are using the papers cited in each chapter of the JM textbook as seed source papers (i.e. we assume that the set of seminal papers on each topic are known). However in the science community, there are thousands more papers that are related to a given topic. In the future, we will work on a method of automatically identifying the most influential papers that represent a specific topic from the vast range of publications.

References

Shannon Bradshaw. 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*.

Aaron Clauset, Mark E. J. Newman, and Cristopher Moore. 2004. Finding community structure in very large networks. *Phys. Rev. E*, 70(6):066111, Dec.

Aaron Elkiss, 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.

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

Taher H. Haveliwala. 2002. Topic-sensitive pagerank. In *Proceedings of the 11th international conference on World Wide Web*, WWW '02, pages 517–526.

Taher H. Haveliwala. 2003. Topic-sensitive pagerank: A context-sensitive ranking algorithm for web search. *Knowledge and Data Engineering, IEEE Transactions on*, 15(4):784–796.

Wesley Hildebrandt, Boris Katz, and Jimmy Lin. 2004. Overview of the trec 2003 question-answering track. In *Proceedings of the 2004 Human Language Technology Conference and the North American Chapter of the Association for Computational Linguistics Annual Meeting (HLT/NAACL 2004)*.

Mark T. Joseph and Dragomir R. Radev. 2007. Citation analysis, centrality, and the ACL Anthology. Technical Report CSE-TR-535-07, University of Michigan. Department of Electrical Engineering and Computer Science.

Daniel Jurafsky and James H. Martin. 2008. *Speech and Language Processing: An Introduction to Natural Language Processing, Speech Recognition, and Computational Linguistics (2nd edition)*. Prentice-Hall.

Min-Yen Kan, 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.

Jon M. Kleinberg. 1999. Authoritative sources in a hyperlinked environment. *J. ACM*, 46:604–632, September.

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

Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. 2005. Graphs over time: densification laws, shrinking diameters and possible explanations. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 177–187. ACM.

Jimmy J. Lin and Dina Demner-Fushman. 2006. Methods for automatically evaluating answers to complex questions. *Information Retrieval*, 9(5):565–587.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Proceedings of the ACL workshop on Text Summarization Branches Out*.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press.

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

Qiaozhu Mei, Jian Guo, and Dragomir Radev. 2010. Divrank: the interplay of prestige and diversity in information networks. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1009–1018.

Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing order into texts. In *Proceedings of EMNLP-04and the 2004 Conference on Empirical Methods in Natural Language Processing*, July.

Rada Mihalcea. 2005. Language independent extractive summarization. In *Proceedings of the ACL 2005 on Interactive poster and demonstration sessions*, ACLdemo '05, pages 49–52, Stroudsburg, PA, USA. Association for Computational Linguistics.

Saif Mohammad, 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 *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 584–592, Boulder, Colorado, June. Association for Computational Linguistics.

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

Hidetsugu Nanba, Takeshi Abekawa, Manabu Okumura, and Suguru Saito. 2004a. Bilingual presri: Integration of multiple research paper databases. In *Proceedings of RIAO 2004*, pages 195–211, Avignon, France.

Hidetsugu Nanba, Noriko Kando, and Manabu Okumura. 2004b. 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.

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

Mark E. J. Newman. 2001. The structure of scientific collaboration networks. *PNAS*, 98(2):404–409.

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

Vahed Qazvinian and Dragomir R. Radev. 2011. Learning from collective human behavior to introduce diversity in lexical choice. In *ACL '11*.

Vahed Qazvinian, Dragomir R. Radev, and Arzucan Özgür. 2010. Citation summarization through keyphrase extraction. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 895–903. Association for Computational Linguistics.

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

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

Simone Teufel, Advaith Siddharthan, and Dan Tidhar. 2006. Automatic classification of citation function. In *Proceedings of the EMNLP*, Sydney, Australia, July.

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

Ellen M. Voorhees. 2003. Overview of the trec 2003 question answering track. In *Proceedings of the Twelfth Text Retrieval Conference (TREC 2003)*.

Xiaojun Wan and Jianwu Yang. 2006. Improved affinity graph based multi-document summarization. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, NAACL-Short '06, pages 181–184, Stroudsburg, PA, USA. Association for Computational Linguistics.

Xiaojun Wan and Jianwu Yang. 2008. Multi-document summarization using cluster-based link analysis. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pages 299–306, New York, NY, USA. ACM.

Stephen Wan, Cecile Paris, and Robert Dale. 2010. Supporting browsing-specific information needs: Introducing the citation-sensitive in-browser summariser. *J. Web Sem.*, 8(2-3):196–202.