Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence

# Surveyor: A System for Generating Coherent Survey Articles for Scientific Topics

**Rahul Jha** and **Reed Coke**
Department of EECS
University of Michigan
Ann Arbor, MI, 48109

**Dragomir Radev**
Department of EECS & School of Information
University of Michigan
Ann Arbor, MI, 48109

## Abstract

We investigate the task of generating coherent survey articles for scientific topics. We introduce an extractive summarization algorithm that combines a content model with a discourse model to generate coherent and readable summaries of scientific topics using text from scientific articles relevant to the topic. Human evaluation on 15 topics in computational linguistics shows that our system produces significantly more coherent summaries than previous systems. Specifically, our system improves the ratings for coherence by 36% in human evaluation compared to C-Lexrank, a state of the art system for scientific article summarization.

## Introduction

This paper is about generating coherent summaries of scientific topics. Given a set of input papers that are relevant to a specific topic such as *question answering*, our system called Surveyor extracts and organizes text segments from these papers into a coherent and readable survey of the topic. There are many applications for automated surveys thus generated. Human surveys do not exist for all topics and quickly become outdated in rapidly growing fields like computer science. Therefore, an automated system for this task can be very useful for new graduate students and cross-disciplinary researchers who need to quickly familiarize themselves with a new topic.

Our work builds on previous work on summarization of scientific literature (Mohammad et al. 2009; Qazvinian and Radev 2008). Prior systems for generating survey articles for scientific topics such as C-Lexrank have focused on building informative summaries but no attempt has been made to ensure the coherence and readability of the output summaries. Surveyor on the other hand focuses on generating survey articles that contain well defined subtopics presented in a coherent order. Figure 1 shows part of the output of Surveyor for the topic of *question answering*.

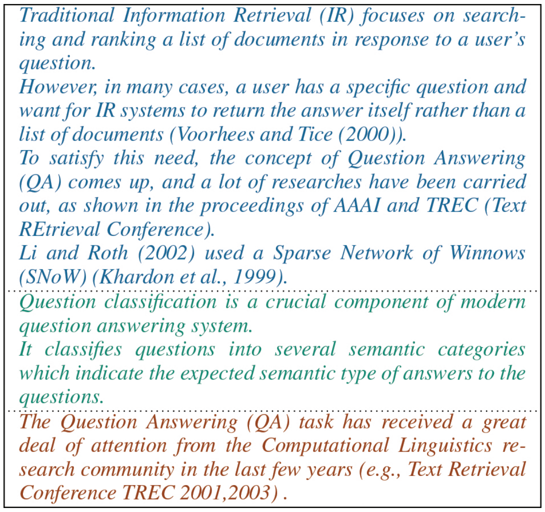
Our experimental results on a corpus of computational linguistics topics show that Surveyor produces survey articles that are substantially more coherent and readable compared to previous work. The main contributions of this paper are:

- We propose a summarization algorithm that combines a content model and a discourse model in a modular way to build coherent summaries.
- We introduce the notion of *Minimum Independent Discourse Contexts* as a way of flexibly modeling discourse relationships in a summarization system.
- We conducted several experiments for evaluating coherence and informativeness of Surveyor on a dataset of 15 topics in computational linguistics with 297 articles and 30 human-written gold summaries (2 per topic). All data used for our experiments is available at http://clair.si.umich.edu/corpora/surveyor_aaai_15.tgz.

We first give an overview of our summarization approach. This is followed by details about our experimental setup and a discussion of results. Finally, we summarize the related work and conclude the paper with pointers for future work.

## Overview of Summarization Approach

We first describe the two main components of our system and then describe our summarization algorithm that is built



*Traditional Information Retrieval (IR) focuses on searching and ranking a list of documents in response to a user's question.*
*However, in many cases, a user has a specific question and want for IR systems to return the answer itself rather than a list of documents (Voorhees and Tice (2000)).*
*To satisfy this need, the concept of Question Answering (QA) comes up, and a lot of researches have been carried out, as shown in the proceedings of AAAI and TREC (Text REtrieval Conference).*
*Li and Roth (2002) used a Sparse Network of Winnows (SNoW) (Khardon et al., 1999).*

*Question classification is a crucial component of modern question answering system.*
*It classifies questions into several semantic categories which indicate the expected semantic type of answers to the questions.*

*The Question Answering (QA) task has received a great deal of attention from the Computational Linguistics research community in the last few years (e.g., Text Retrieval Conference TREC 2001,2003) .*

Figure 1: Example output of Surveyor for the topic of *question answering*. The survey contains three distinct subtopics illustrated by different colors and separated by dashed lines.

| **subtopic 1** |
|---|
| *BB constructs classifiers for English-to-Chinese translation disambiguation by repeating the following two steps: (1) Construct a classifier for each of the languages on the basis of classified data in both languages, and (2) use the constructed classifier for each language to classify unclassified data, which are then added to the classified data of the language.* |
| *In translation from English to Chinese, for example, BB makes use of unclassified data from both languages.* |
| **subtopic 2** |
| *Word sense disambiguation (WSD) is the problem of assigning a sense to an ambiguous word, using its context.* |
| *The task of Word Sense Disambiguation (WSD) is to identify the correct sense of a word in context.* |
| **subtopic 3** |
| *We extend previously reported work in a number of different directions: We evaluate the method on all parts of speech (PoS) on SemCor.* |
| *Previous experiments evaluated only nouns on SemCor, or all PoS but only on the Senseval2 and Senseval3 data.* |

Figure 2: Example sentences from three subtopics learned by the HMM for *word sense disambiguation*.

on top of them.

**Content Model** Given a set of research papers relevant to a scientific topic, each of them focusing on a specific aspect of the problem. For example, a paper on supervised word sense disambiguation might describe the background on word sense disambiguation followed by a review of supervised methods for the problem. Similarly, a paper on unsupervised word sense disambiguation may give some general overview of the field, then briefly describe supervised approaches followed by a more detailed overview of unsupervised methods. We capture these subtopics in the input documents and their transitions using a Hidden Markov Model (HMM) where the states of the HMM correspond to subtopics. Given the set of $k$ subtopics $S = (s_1 \cdots s_k)$, the state transitions of the HMM are defined as:

$$p(s_j|s_i) = \frac{Count(s_i, s_j) + \delta}{Count(s_i) + \delta * m}$$

Where $Count(s_i, s_j)$ is the number of times a sentence from subtopic $s_j$ appears immediately after a sentence from subtopic $s_i$ in the input document collection and $Count(s_i)$ is the total number of times the subtopic $s_i$ appears in the input document set. $\delta$ is a smoothing parameter and $m$ is the number of sentences in $s_i$.

To initialize the states of the HMM, we use a network based clustering approach. We build a lexical network where the sentences represent the nodes of the network and the edge weights are the tf*idf similarity between each pair of sentences [1]. Given this lexical network, we use the Louvain clustering method (De Meo et al. 2011) to partition the lexical network into clusters. Each cluster in the network is then initialized to a sub-topic. Louvain is a hierarchical clustering algorithm that does not need the number of output clusters as a parameter. The HMM is then learned through Viterbi decoding. Our HMM model is similar to (Barzilay and Lee 2004), but we take the novel step of using the transition matrix to guide the summarization output, as described below.

| | **subtopic 1** | **subtopic 2** | **subtopic 3** |
|---|---|---|---|
| $start$ | 0.35 | 0.50 | 0 |
| **subtopic 1** | 0.49 | 0.22 | 0 |
| **subtopic 2** | 0.24 | 0.41 | 0.02 |
| **subtopic 3** | 0.25 | 0.25 | 0.50 |

Table 1: A partial table of transition probabilities between three subtopics for *word sense disambiguation*. The probabilities do not add up to 1 because the table only shows a few states from a larger transition matrix.

Figure 2 shows sentences from three of the subtopics learned for the topic of *word sense disambiguation*. In a coherent summary, subtopic 2 containing background sentences should appear before subtopic 1 that contains details about a specific method. We use the transition matrix of the learned HMM to model these subtopic transitions in the original documents and use it to guide the summarizer output. As an example, a partial table of transition probabilities learned for the subtopics in Figure 2 is shown in Table 1, where $start$ is a pseudo-state representing the beginning of the document. The highest outgoing probability from $start$ is to subtopic 2, which allows the summarizer to include background information about the topic at the beginning followed by sentences from more specific subtopics represented by subtopic 1 and subtopic 3.

**Discourse Model** A common problem with extractive summaries is that the sentences used from the original input documents may not be understandable when pulled out of their original context. To avoid such problems, we introduce the idea of *Minimum Independent Discourse Contexts (MIDC)*.

**Definition.** *Given a text segment $T$ containing $n$ sentences $(s_1 \cdots s_n)$, the minimum independent discourse context (midc) of a sentence $s_i$ is defined as the minimum set of $j$ sentences $midc(s_i) = (s_{i-j} \cdots s_i)$ such that given $midc(s_i)$, $s_i$ can be interpreted independently of the other sentences in $T$.*

Figure 3 shows how this idea works in practice. Sentences $s1$ and $s4$ can be included in a summary without requiring additional context sentences. Sentences $s2$, $s3$ and $s4$ on the other hand, depend on a set of previous sentences in order to be understandable. A summary that includes sentence $s3$,

[1] The idfs are computed over the entire input corpus.

| | |
|---|---|
| s1 | Opinion words are words that convey positive or negative polarities. |
| s2 | They are critical for opinion mining (Pang et al., 2002; Turney, 2002; Hu and Liu, 2004; Wilson et al., 2004; Popescu and Etzioni, 2005; Gamon et al., 2005; Ku et al., 2006; Breck et al., 2007; Kobayashi et al., 2007; Ding et al., 2008; Titov and McDonald, 2008; Pang and Lee, 2008; Lu et al., 2009). |
| s3 | The key difficulty in finding such words is that opinions expressed by many of them are domain or context dependent. |
| s4 | Several researchers have studied the problem of finding opinion words (Liu, 2010). |
| s5 | The approaches can be grouped into corpus-based approaches (Hatzivassiloglou and McKeown, 1997; Wiebe, 2000; Kanayama and Nasukawa, 2006; Qiu et al., 2009) and dictionary-based approaches (Hu and Liu 2004; Kim and Hovy, 2004; Kamps et al., 2004; Esuli and Sebastiani, 2005; Takamura et al., 2005; Andreevskaia and Bergler, 2006; Dragut et al., 2010). |

$$\begin{aligned} midc(s1) &= \emptyset \\ midc(s2) &= \{s1\} \\ midc(s3) &= \{s1, s2\} \\ midc(s4) &= \emptyset \\ midc(s5) &= \{s4\} \end{aligned}$$

Figure 3: A paragraph from an input paper on the topic of *opinion mining* along with the *midc* for each sentence on the right.

| **Discourse relationship** | **Dependency rule** |
|---|---|
| Coreference | Add a dependency between $s_i$ and $s_j$ if they belong to a coreference chain. |
| Discourse Transition | Add a dependency between $s_{i-1}$ and $s_i$ if $s_i$ contains an explicit discourse marker. |
| Entity Transition | Add a dependency between $s_i$ and $s_j$ if they both share a prominent entity. |

Table 2: Discourse rules used to create *minimum independent discourse contexts*.

for example, must include sentences $s1$ and $s2$ for it to be understood outside of its original text.

To calculate the *midc*s for sentences in practice, we use discourse rules that are triggered by coreference dependencies, explicit discourse dependencies and entity links between sentences. These rules are summarized in Table 2. Every time a discourse rule is triggered, a dependency is added between two sentences. The *midc* for a sentence $s_i$ is all the sentences preceding $s_i$ in the input document to which it has a dependency edge. The coreference chains are found using the Stanford dependency parser (de Marneffe, MacCartney, and Manning 2006) and the discourse markers are obtained from the Penn Discourse TreeBank (Prasad et al. 2008). The prominent entities used for creating entity links are nouns that appear in the syntactic role of subject or object in any sentence in the input.

Treating *midc*s as the units of content that are combined to build the extractive summaries allows us to take into account discourse level dependencies and generate locally coherent summaries without complicating the optimization procedure.

**Summarization Algorithm** We now describe how our summarization algorithm works given the output of these two components. The pseudocode for the algorithm is presented in Figure 4.

The algorithm accepts a set of input documents $docs$ and a maximum summary length $maxlen$. It first learns the subtopics and their transition matrix by running HMM on the input document set. After initializing the first subtopic to the pseudo-subtopic $start$, it iteratively picks the next subtopic by using the HMM transition matrix. Given each subtopic, it runs a salience algorithm on all the sentences of the subtopic to find the most central sentence of the subtopic. In the current implementation, this is done using Lexrank (Erkan and Radev 2004). Given the subtopic's most central sentence, it calculates the *midc* for this sentence and if the *midc* is valid, it is added to the output summary. An *midc* can be invalid if it exceeds a maximum threshold number of sentences [2] The *midc* is then removed from the subtopic so it will not be picked if we visit this subtopic again. This procedure continues until we obtain a summary of the desired length. Important subtopics in the input can get more than one *midc* in the summary because the transition matrix contains high probabilities for transitioning to these subtopics.

```
input  : docs, maxlen
output: summary of length maxlen
summary $\leftarrow \emptyset$;
transitionMatrix $\leftarrow$ HMM(docs);
curSubtopic $\leftarrow$ start;
while len(summary) $<$ maxlen do
    nextSubtopic $\leftarrow$ transitionMatrix.next(curSubtopic);
    nextSent $\leftarrow$ getMostSalient(sents(nextSubtopic)));
    nextMidc $\leftarrow$ midc(nextSent);
    if valid(nextMidc) then
        summary.add(nextMidc);
        sents(nextSubtopic).remove(nextMidc)
    end
    curSubtopic $\leftarrow$ nextSubtopic;
end
return summary;
```

Figure 4: Summarization Algorithm.

[2] This constraint is added so that a highly salient sentence with a long *midc* does not dominate most of the output summary.

## Experimental Setup

The main research questions that we want to answer using our experiments are:

1. Are the summaries created using Surveyor more coherent than previous state-of-the-art methods for survey article generation?
2. What are the individual contributions of the content model and the discourse model?
3. How does Surveyor compare against state-of-the-art systems for coherent news summarization applied to the survey generation problem?

For research question 1, we compare our system with C-Lexrank (Mohammad et al. 2009), a state-of-the-art system for survey generation. For research question 2, we measure the effects of HMM and MIDC models in isolation on the quality of output summaries. For research question 3, we compare our system with G-FLOW (Christensen et al. 2013), a state-of-the-art system for coherent summarization of news articles. We now describe the data used in our experiments.

### Data

We used the ACL Anthology Network (AAN) (Radev et al. 2013) as a corpus for our experiments and selected 15 established topics in computational linguistics for our evaluation. The input documents used for summarization of a research topic should be research papers that describe the most relevant research in the topic. Since the focus of this paper is on summarization, we used an oracle method for selecting the initial set of papers for each topic. We collected at least three human-written surveys on each topic. The bibliographies of all the surveys were processed using Parscit (Luong, Nguyen, and Kan 2010) and any document that appeared in the bibliography of more than one survey was added to the initial document set $D_i$.

An ideal survey article on the topic should describe the research represented by $D_i$. These sentences are actually found in papers that cite papers in $D_i$ and thus describe their contributions. Therefore to create the final document set $D_f$, we collect all the papers in AAN that cite the papers in $D_i$. [3] The citing documents are then ordered based on the number of papers in $D_i$ that they cite and the top $n$ documents are added to $D_f$. The text input for the summarization system is extracted from $D_f$. For our current experiments, the value of $n$ is set to 20.

For the task of survey article generation, the most relevant text is found in the introduction sections of $D_f$ since this is where researchers describe the prior work done by subsets of papers in $D_i$. Therefore, we extract the sentences in the introductions of each of the papers in $D_f$ as the text input for our summarizer. Table 3 shows the set of 15 topics and size of summarizer input for each topic.

| Topic | # Sentences |
|---|---|
| coreference resolution | 397 |
| dependency parsing | 487 |
| grammar induction | 407 |
| information extraction | 495 |
| information retrieval | 560 |
| machine translation | 552 |
| named entity recognition | 383 |
| question answering | 452 |
| semantic role labeling | 466 |
| semi supervised learning | 506 |
| sentiment analysis | 613 |
| speech recognition | 445 |
| summarization | 507 |
| topic modeling | 412 |
| word sense disambiguation | 425 |

Table 3: List of topics used in our experiments.

### Experiments

**Coherence Evaluation with C-Lexrank** For coherence evaluation, we generated fixed length 2000 character summaries using both C-Lexrank and Surveyor. Six assessors with background in computational linguistics manually evaluated pairs of output summaries. Given two summaries, the assessors were asked to mark which summary they preferred, or mark "indifferent" if they could not choose one against the other. The presentation of summary pairs to assessors as well as the order of summaries in each pair was randomized. Thus, the assessors had no way of telling which systems produced the pair of summaries they saw.

Compared to C-Lexrank, the assessors preferred a summary generated by Surveyor 67% of the time and were indifferent 20% of the time (Table 4).

| Surveyor | Indifferent | C-Lexrank |
|---|---|---|
| 67% | 20% | 13% |

Table 4: Overall summary preference for Surveyor compared to C-Lexrank.

Additionally, the assessors were asked to rate each summary based on the standard DUC quality questions [4]. The DUC quality questions are a standard benchmark used for evaluating summaries on the aspects of overall coherence, avoiding useless text, avoiding repetitive information, avoiding bad referents and avoiding overly explicit referents. For each of the questions, the assessors can assign a score from 1 to 5 with higher being better.

As shown in Table 5, the assessors also assigned much higher scores to summaries generated by Surveyor on an average compared to C-Lexrank on all the DUC quality questions [5]. On the metric of coherence, the scores for Surveyor

[3] On average, we found only 33% of the documents in $D_i$ to be in AAN. Since the citation network for AAN contains only citations within AAN documents, we implemented a record matching algorithm to find all the papers in AAN that cite any arbitrary document outside AAN.

[4] http://duc.nist.gov/duc2004/quality.questions.txt

[5] DUC quality responses represent a Likert-type scale. The use of parametric statistics such as mean for such data has been debated, but there are several recent arguments for its validity (Norman 2010).

| Quality Question | C-Lexrank | Surveyor | Surveyor HMM Only | Surveyor MIDC Only |
|---|---|---|---|---|
| coherence | 2.72 ± 0.16 | 3.70 ± 0.22 | 2.57 ± 0.20 | 3.07 ± 0.36 |
| avoid useless text | 3.20 ± 0.15 | 3.90 ± 0.15 | 3.17 ± 0.19 | 3.33 ± 0.30 |
| avoid repetition | 4.07 ± 0.11 | 4.23 ± 0.14 | 3.97 ± 0.19 | 4.40 ± 0.19 |
| avoid bad referents | 3.43 ± 0.16 | 4.17 ± 0.14 | 3.60 ± 0.18 | 3.47 ± 0.27 |
| avoid overly explicit referents | 4.23 ± 0.12 | 4.47 ± 0.11 | 4.30 ± 0.19 | 4.53 ± 0.22 |

Table 5: Average scores on the DUC quality questions for C-Lexrank and different Surveyor variants along with standard error.

compared to C-Lexrank were higher by 36%. Both on the metrics of avoiding useless text and avoiding bad referents, the scores for Surveyor were higher by about 22%.

**Contribution of Individual Components** To compare the contribution of the content model and the discourse model, we created two additional variants of our system. *Surveyor HMM Only* contains only the HMM component, but does not use the discourse component that adds the *midc*s for the output sentences. *Surveyor MIDC only* uses the discourse component, but instead of relying on the HMM transition matrix to generate the subtopic flow, chooses the subtopics based on their size, where size of a subtopic is the number of sentences assigned to the subtopic. It starts from the largest subtopic and goes through subtopics in order of their size.

We asked our assessors to compare summaries output by each system with the output of C-Lexrank as well rate summaries produced by each system on the DUC quality questions. The results of the direct comparison is summarized in Table 6 and the average DUC ratings are reported in Table 5.

| Surveyor HMM Only | Indifferent | C-Lexrank |
|---|---|---|
| 53% | 27% | 20% |
| **Surveyor MIDC Only** | **Indifferent** | **C-Lexrank** |
| 33% | 27% | 40% |

Table 6: Overall summary preference for the two Surveyor variants compared to C-Lexrank.

Even with just the HMM content model, the summaries from *Surveyor HMM Only* are preferred by assessors compared to C-Lexrank. *Surveyor MIDC Only* does not do as well in direct comparison, which suggests that without a coherent flow of subtopics, the addition of *midc*s to the output sentences does not improve the perceived overall quality of a summary. This shows the importance of the HMM content model and suggests that a summary that jumps between subtopics in an incoherent way will not be perceived as coherent even if the individual sentences in the summary have appropriate context. However, the scores for both of these systems on the DUC quality questions (Table 5) show that the addition of *midc*s does affect the assessors' judgement of specific summary qualities and is an important component of the system. This explains why the combination of both the content model and the discourse model leads to much better results than either of them in isolation (Tables 4 and 6).

**Informativeness Evaluation** We use ROUGE (Lin 2004b) for informativeness evaluation. ROUGE is a standard evaluation metric for automatic evaluation of summaries that uses n-gram co-occurrences between automated summaries and human generated reference summaries to score the automated summaries. ROUGE has been shown to correlate well with human evaluations (Lin 2004a).

For ROUGE evaluation, we asked two assessors to generate 2000 character long gold summaries using the input for each topic. We then did ROUGE evaluation of the summaries generated using C-Lexrank and Surveyor against these gold summaries. The average ROUGE-1 and ROUGE-2 scores are summarized below in Table 7. The improvement in ROUGE scores of Surveyor over C-Lexrank is statistically significant with $p < 0.05$. Thus Surveyor, in addition to producing more coherent summaries, also produces summaries that are more informative given the same input text.

| System | ROUGE-1 | ROUGE-2 |
|---|---|---|
| C-Lexrank | 0.40 | 0.05 |
| Surveyor | 0.44 | 0.19 |
| Surveyor HMM Only | 0.42 | 0.13 |
| Surveyor MIDC only | 0.42 | 0.13 |

Table 7: Average Rouge scores for each of the systems for 15 topics evaluated against two reference summaries per topic.

Previous evaluations for survey generation systems use citing sentences as input as opposed to sentences from the main text. There is no standard summarization evaluation that allows us to evaluate the informativeness of summaries generated using two different input sources. To compare summaries created using citing sentences and source sentences in terms of coherence, we ran C-Lexrank using both citing sentences and introduction sentences as summarizer input and did a coherence evaluation with our assessors. The assessors preferred summaries generated by using introductions as source 60% of the time while preferring summaries generated by using citing sentences as source only 27% of the time. Even though a direct comparison of informativeness is not possible, we posit that since our summaries include background information as part of the survey, our summaries would have to be slightly longer than those based on citing sentences in order to be as informative. However, results from coherence evaluation show that using source sentences allows us to use topical and discourse information in the original papers to generate much more coherent summaries compared to citing sentences.

**Evaluation with G-FLOW** G-FLOW (Christensen et al. 2013) is a recent state of the art system for generating coherent summaries that has been evaluated on newswire data. We compared Surveyor with G-Flow by running the implementation of G-Flow obtained from the original authors on our evaluation data. The coherence evaluation with G-Flow was done in the same way as for C-Lexrank except the output summary length for both systems was limited to 1000 characters. This is because the optimization procedure implemented in G-Flow becomes intractable for output of 2000 characters [6].

In the coherence evaluation, assessors preferred Surveyor 47% of the time compared to 40% of the time for G-Flow (Table 8).

| Surveyor | Indifferent | G-Flow |
|---|---|---|
| 47% | 13% | 40% |

Table 8: Overall summary preference for Surveyor compared to G-Flow.

Surveyor also obtains higher scores than G-Flow on the DUC quality questions (Table 9). The scores for Surveyor and G-Flow are summarized below separately because of the difference in the output length compared to the previous evaluation. The numbers are reported with standard error.

| Quality Question | Surveyor | G-Flow |
|---|---|---|
| coherence | 3.53 ± 0.36 | 3.40 ± 0.25 |
| avoid useless text | 3.60 ± 0.36 | 3.47 ± 0.17 |
| avoid repetition | 4.93 ± 0.07 | 4.53 ± 0.19 |
| avoid bad referents | 3.93 ± 0.33 | 3.80 ± 0.22 |
| avoid overly explicit referents | 4.73 ± 0.12 | 4.47 ± 0.19 |

Table 9: Average scores on the DUC quality questions for Surveyor compared with G-Flow.

In informativeness evaluation with ROUGE, the 1000 character summaries generated by Surveyor got an average ROUGE-1 score of 0.41 compared to a score of 0.36 obtained by G-Flow. The ROUGE-2 score of Surveyor was 0.13 compared to 0.07 for G-Flow. p-values for the ROUGE-1 and ROUGE-2 improvements of Surveyor over G-Flow are 0.12 and 0.11 respectively. These results indicate that Surveyor does slightly better than G-Flow in terms of coherence evaluation while also producing informative summaries. This suggests that the HMM based content model does a better job of modeling the flow of subtopics in scientific articles compared to G-Flow which does not include such a component.

## Related Work

Multi-document summarization of scientific articles has been studied by Nanba, Kando, and Okumura (2004). Mohammad et al. (2009) compared several algorithms for generating automated surveys of scientific topics. Jha, Abu-Jbara, and Radev (2013) implemented a system that can summarize a topic starting from a query as input. However, none of these papers focused on evaluating the coherence of resulting summaries. In the medical domain, several summarization systems have been proposed that take advantage of the rich ontological data available for medical concepts (Elhadad and McKeown 2001; Kan, McKeown, and Klavans 2001; Yoo, Hu, and Song 2006). A different stream of research has looked at summarizing scientific research using the metaphor of maps (Fried and Kobourov 2013; Shahaf, Guestrin, and Horvitz 2012). For the work on single document summarization for scientific literature, we refer the readers to the review in Nenkova and McKeown (2011).

Ordering the sentences in summarization output for improving readability has been studied by Barzilay and McKeown (2005) and Bollegala, Okazaki, and Ishizuka (2010). Automatic metrics for estimating coherence for summarization evaluation have also been studied (Lapata 2005; Lin et al. 2012). More recently, Christensen et al. (2013) presented an algorithm called G-Flow for joint sentence selection and ordering for news summarization.

Barzilay and Lee (2004) and Fung and Ngai (2006) have presented HMM based content models that use the HMM topics as features in a supervised summarization system to produce informative summaries. LDA (Latent Dirichlet Allocation) based content models for summarizing documents (Daumé and Marcu 2006; Haghighi and Vanderwende 2009) have also been explored, but they focus on maximizing informativeness instead of coherence.

## Conclusion and Future Work

In this paper, we present Surveyor, a system for generating coherent surveys of scientific articles. We describe our algorithm and present experimental results on a corpus of 15 topics in computational linguistics. Our results show that our system leads to more coherent summaries than C-Lexrank, a state-of-the-art system for survey article generation and G-Flow, a state-of-the-art system for coherent summarization. In particular, in human evaluation for coherence, Surveyor outperforms the performance of C-Lexrank by 36% and outperforms the performance of G-Flow by 4%.

This work suggests several possible future directions for research. The first is developing more sophisticated content models that better capture the distribution of topics in scientific documents across genres. The second is building a corpus of discourse relationships between sentences in scientific documents as well as improving the algorithm for creating minimum independent discourse context. Finally, automatic sentence compression, fusion and rewriting strategies can be applied to sentences of the output summary to remove irrelevant text segments and improve the informativeness of the summaries.

## Acknowledgments

We thank the anonymous reviewers for their feedback on earlier versions of this paper. Also, thanks to Janara Christensen for providing results of running G-Flow on our data. Finally, thanks to Inderjeet Mani, Zornitsa Kozareva, Or Biran, Ben King, Veronica Perez Rosas, Charlie Welch, Shibamouli Lahiri, Max Kaufmann and Shiwali Mohan for several useful discussions and help with evaluations.

[6] Personal communication with Christensen et al.

# References

Barzilay, R., and Lee, L. 2004. Catching the drift: Probabilistic content models, with applications to generation and summarization. In Susan Dumais, D. M., and Roukos, S., eds., *HLT-NAACL 2004: Main Proceedings*, 113–120. Boston, Massachusetts, USA: Association for Computational Linguistics.

Barzilay, R., and McKeown, K. R. 2005. Sentence fusion for multidocument news summarization. *Computational Linguistics* 31(3):297–328.

Bollegala, D.; Okazaki, N.; and Ishizuka, M. 2010. A bottom-up approach to sentence ordering for multi-document summarization. *Information Processing & Management* 46(1):89 – 109.

Christensen, J.; Mausam; Soderland, S.; and Etzioni, O. 2013. Towards coherent multi-document summarization. In *Proceedings of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2013)*.

Daumé, III, H., and Marcu, D. 2006. Bayesian query-focused summarization. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, ACL-44, 305–312. Stroudsburg, PA, USA: Association for Computational Linguistics.

de Marneffe, M.-C.; MacCartney, B.; and Manning, C. D. 2006. Generating typed dependency parses from phrase structure parses. In *The International Conference on Language Resources and Evaluation (LREC)*, volume 6, 449–454. Citeseer.

De Meo, P.; Ferrara, E.; Fiumara, G.; and Provetti, A. 2011. Generalized louvain method for community detection in large networks. *CoRR* abs/1108.1502.

Elhadad, N., and McKeown, K. R. 2001. Towards generating patient specific summaries of medical articles. In *In Proceedings of NAACL-2001 Automatic Summarization Workshop*.

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

Fried, D., and Kobourov, S. G. 2013. Maps of Computer Science. *ArXiv e-prints*.

Fung, P., and Ngai, G. 2006. One story, one flow: Hidden markov story models for multilingual multidocument summarization. *ACM Trans. Speech Lang. Process.* 3(2):1–16.

Haghighi, A., and Vanderwende, L. 2009. Exploring content models for multi-document summarization. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL '09, 362–370. Stroudsburg, PA, USA: Association for Computational Linguistics.

Jha, R.; Abu-Jbara, A.; and Radev, D. R. 2013. A system for summarizing scientific topics starting from keywords. In *Proceedings of The Association for Computational Linguistics (short paper)*.

Kan, M.; McKeown, K. R.; and Klavans, J. L. 2001. Domain-specific informative and indicative summarization for information retrieval. In *Proc. of the Document Understanding Conference (DUC*, 6.

Lapata, M. 2005. Automatic evaluation of text coherence: models and representations. In *In the Intl. Joint Conferences on Artificial Intelligence*, 1085–1090.

Lin, Z.; Liu, C.; Ng, H. T.; and Kan, M.-Y. 2012. Combining coherence models and machine translation evaluation metrics for summarization evaluation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1*, ACL '12, 1006–1014. Stroudsburg, PA, USA: Association for Computational Linguistics.

Lin, C. Y. 2004a. Looking for a few good metrics: Automatic summarization evaluation - how many samples are enough? In *Proceedings of the NTCIR Workshop 4*.

Lin, C.-Y. 2004b. ROUGE: A package for automatic evaluation of summaries. In *Proceedings of the ACL workshop on Text Summarization Branches Out*.

Luong, M.-t.; Nguyen, T. D.; and Kan, M.-y. 2010. Logical Structure Recovery in Scholarly Articles with Rich Document Features. *IJDLS*.

Mohammad, S.; Dorr, B.; Egan, M.; Hassan, A.; Muthukrishan, P.; Qazvinian, V.; Radev, D.; and Zajic, D. 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*, NAACL '09, 584–592. Stroudsburg, PA, USA: Association for Computational Linguistics.

Nanba, H.; Kando, N.; and Okumura, M. 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*, 117–134.

Nenkova, A., and McKeown, K. 2011. Automatic summarization. *Foundations and Trends in Information Retrieval* 5(2-3):103–233.

Norman, G. 2010. Likert scales, levels of measurement and the laws of statistics. *Advances in Health Sciences Education* 15(5):625–632.

Prasad, R.; Dinesh, N.; Lee, A.; Miltsakaki, E.; Robaldo, L.; Joshi, A.; and Webber, B. 2008. The penn discourse treebank 2.0. In *In Proceedings of LREC*.

Qazvinian, V., and Radev, D. R. 2008. Scientific paper summarization using citation summary networks. In *COLING 2008*.

Radev, D. R.; Muthukrishnan, P.; Qazvinian, V.; and Abu-Jbara, A. 2013. The acl anthology network corpus. *Language Resources and Evaluation* 1–26.

Shahaf, D.; Guestrin, C.; and Horvitz, E. 2012. Metro maps of science. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, 1122–1130. New York, NY, USA: ACM.

Yoo, I.; Hu, X.; and Song, I.-Y. 2006. A coherent biomedical literature clustering and summarization approach through ontology-enriched graphical representations. In Tjoa, A., and Trujillo, J., eds., *Data Warehousing and Knowledge Discovery*, volume 4081 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg. 374–383.