

Query Snowball: A Co-occurrence-based Approach to Multi-document Summarization for Question Answering

Hajime Morita^{1 2} and Tetsuya Sakai¹ and Manabu Okumura³

¹Microsoft Research Asia, Beijing, China

²Tokyo Institute of Technology, Tokyo, Japan

³Precision and Intelligence Laboratory, Tokyo Institute of Technology, Tokyo, Japan

morita@lr.pi.titech.ac.jp, tetsuyasakai@acm.org,

oku@pi.titech.ac.jp

Abstract

We propose a new method for query-oriented extractive multi-document summarization. To enrich the information need representation of a given query, we build a co-occurrence graph to obtain words that augment the original query terms. We then formulate the summarization problem as a Maximum Coverage Problem with Knapsack Constraints based on word pairs rather than single words. Our experiments with the NTCIR ACLIA question answering test collections show that our method achieves a pyramid F3-score of up to 0.313, a 36% improvement over a baseline using Maximal Marginal Relevance.

1 Introduction

Automatic text summarization aims at reducing the amount of text the user has to read while preserving important contents, and has many applications in this age of digital information overload (Mani, 2001). In particular, *query-oriented multi-document summarization* is useful for helping the user satisfy his information need efficiently by gathering important pieces of information from multiple documents.

In this study, we focus on *extractive* summarization (Liu and Liu, 2009), in particular, on sentence selection from a given set of source documents that contain relevant sentences. One well-known challenge in selecting sentences relevant to the information need is the vocabulary mismatch between the query (i.e. information need representation) and the candidate sentences. Hence, to enrich the information need representation, we build a co-occurrence

graph to obtain words that augment the original query terms. We call this method *Query Snowball*.

Another challenge in sentence selection for query-oriented multi-document summarization is how to avoid redundancy so that diverse pieces of information (i.e. *nuggets* (Voorhees, 2003)) can be covered. For penalizing redundancy across sentences, using single words as the basic unit may not always be appropriate, because different nuggets for a given information need often have many words in common. Figure 1 shows an example of this word overlap problem from the NTCIR-8 ACLIA2 Japanese question answering test collection. Here, two gold-standard nuggets for the question “*Sen to Chihiro no Kamikakushi (Spirited Away)* is a full-length animated movie from Japan. The user wants to know how it was received overseas.” (in English translation) is shown. Each nugget represents a particular award that the movie received, and the two Japanese nugget strings have as many as three words in common: “批評 (review/critic)”, “アニメ (animation)” and “賞 (award).” Thus, if we use single words as the basis for penalising redundancy in sentence selection, it would be difficult to cover both of these nuggets in the summary because of the word overlaps.

We therefore use *word pairs* as the basic unit for computing sentence scores, and then formulate the summarization problem as a Maximum Cover Problem with Knapsack Constraints (MCKP) (Filatova and Hatzivassiloglou, 2004; Takamura and Okumura, 2009a). This problem is an optimization problem that maximizes the total score of words covered by a summary under a summary length limit.

- Question
Sen to Chihiro no Kamikakushi (Spirited Away) is a full-length animated movie from Japan. The user wants to know how it was received overseas.
- Nugget example 1
 全米映画批評会議のアニメ賞
 National Board of Review of Motion Pictures Best Animated Feature
- Nugget example 2
 ロサンゼルス批評家協会賞のアニメ賞
 Los Angeles Film Critics Association Award for Best Animated Film

Figure 1: Question and gold-standard nuggets example in NTCIR-8 ACLIA2 dataset

We evaluate our proposed method using Japanese complex question answering test collections from NTCIR ACLIA–Advanced Cross-lingual Information Access task (Mitamura et al., 2008; Mitamura et al., 2010). However, our method can easily be extended for handling other languages.

2 Related Work

Much work has been done for generic multi-document summarization (Takamura and Okumura, 2009a; Takamura and Okumura, 2009b; Celikyilmaz and Hakkani-Tur, 2010; Lin et al., 2010a; Lin and Bilmes, 2010). Carbonell and Goldstein (1998) proposed the Maximal Marginal Relevance (MMR) criteria for non-redundant sentence selection, which consist of document similarity and redundancy penalty. McDonald (2007) presented an approximate dynamic programming approach to maximize the MMR criteria. Yih et al. (2007) formulated the document summarization problem as an MCKP, and proposed a supervised method. Whereas, our method is unsupervised. Filatova and Hatzivassiloglou (2004) also formulated summarization as an MCKP, and they used two types of concepts in documents: single words and *events* (named entity pairs with a verb or a noun). While their work was for generic summarization, our method is designed specifically for query-oriented summarization.

MMR-based methods are also popular for query-oriented summarization (Jagarlamudi et al., 2005; Li et al., 2008; Hasegawa et al., 2010; Lin et al., 2010b). Moreover, graph-based methods for summarization and sentence retrieval are popular (Otterbacher et al., 2005; Varadarajan and Hristidis, 2006;

Bosma, 2009). Unlike existing graph-based methods, our method explicitly computes indirect relationships between the query and words in the documents to enrich the information need representation. To this end, our method utilizes within-sentence co-occurrences of words.

The approach taken by Jagarlamudi et al. (2005) is similar to our proposed method in that it uses word co-occurrence and dependencies within sentences in order to measure relevance of words to the query. However, while their approach measures the generic relevance of each word based on *Hyperspace Analogue to Language* (Lund and Burgess, 1996) using an external corpus, our method measures the relevance of each word within the document contexts, and the query relevance scores are propagated recursively.

3 Proposed Method

Section 3.1 introduces the Query Snowball (QSB) method which computes the query relevance score for each word. Then, Section 3.2 describes how we formulate the summarization problem based on word pairs.

3.1 Query Snowball method (QSB)

The basic idea behind QSB is to close the gap between the query (i.e. information need representation) and relevant sentences by enriching the information need representation based on co-occurrences. To this end, QSB computes a *query relevance score* for each word in the source documents as described below.

Figure 2 shows the concept of QSB. Here, Q is the set of query terms (each represented by q), $R1$ is the set of words ($r1$) that co-occur with a query term in the same sentence, and $R2$ is the set of words ($r2$) that co-occur with a word from $R1$, excluding those that are already in $R1$. The imaginary root node at the center represents the information need, and we assume that the need is propagated through this graph, where edges represent within-sentence co-occurrences. Thus, to compute sentence scores, we use not only the query terms but also the words in $R1$ and $R2$.

Our first clue for computing a word score is the query-independent importance of the word.

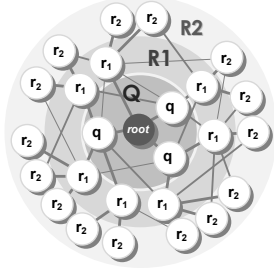


Figure 2: Co-occurrence Graph (Query Snowball)

We represent this *base word score* by $s_b(w) = \log(N/ctf(w))$ or $s_b(w) = \log(N/n(w))$, where $ctf(w)$ is the total number of occurrences of w within the corpus and $n(w)$ is the document frequency of w , and N is the total number of documents in the corpus. We will refer to these two versions as *itf* and *idf*, respectively. Our second clue is the weight propagated from the center of the co-occurrence graph shown in Figure 1. Below, we describe how to compute the word scores for words in $R1$ and then those for words in $R2$.

As Figure 2 suggests, the query relevance score for $r1 \in R1$ is computed based not only on its base word score but also on the relationship between $r1$ and $q \in Q$. To be more specific, let $freq(w, w')$ denote the within-sentence co-occurrence frequency for words w and w' , and let $distance(w, w')$ denote the *minimum dependency distance* between w and w' : A dependency distance is the path length between nodes w and w' within a dependency parse tree; the minimum dependency distance is the shortest path length among all dependency parse trees of source-document sentences in which w and w' co-occur. Then, the query relevance score for $r1$ can be computed as:

$$s_r(r1) = \sum_{q \in Q} s_b(q) \left(\frac{s_b(q)}{sum_Q} \right) \left(\frac{freq(q, r1)}{distance(q, r1) + 1.0} \right) \quad (1)$$

where $sum_Q = \sum_{q \in Q} s_b(q)$. It can be observed that the query relevance score $s_r(r1)$ reflects the base word scores of both q and $r1$, as well as the co-occurrence frequency $freq(q, r1)$. Moreover, $s_r(r1)$ depends on $distance(q, r1)$, the minimum dependency distance between q and $r1$, which reflects the strength of relationship between q and $r1$. This quantity is used in one of its denominators in Eq.1 as small values of $distance(q, r1)$ imply a strong relationship between q and $r1$. The 1.0 in the denominator avoids division by zero.

Similarly, the query relevance score for $r2 \in R2$ is computed based on the base word score of $r2$ and the relationship between $r2$ and $r1 \in R1$:

$$s_r(r2) = \sum_{r1 \in R1} s_b(r2) \left(\frac{s_r(r1)}{sum_{R1}} \right) \left(\frac{freq(r1, r2)}{distance(r1, r2) + 1.0} \right) \quad (2)$$

where $sum_{R1} = \sum_{r1 \in R1} s_r(r1)$.

3.2 Score Maximization Using Word Pairs

Having determined the query relevance score, the next step is to define the summary score. To this end, we use word pairs rather than individual words as the basic unit. This is because word pairs are more informative for discriminating across different pieces of information than single common words. (Recall the example mentioned in Section 1) Thus, the word pair score is simply defined as: $s_p(w_1, w_2) = s_r(w_1)s_r(w_2)$ and the summary score is computed as:

$$f_{QSBP}(S) = \sum_{\{w_1, w_2 | w_1 \neq w_2 \text{ and } w_1, w_2 \in u \text{ and } u \in S\}} s_p(w_1, w_2) \quad (3)$$

where u is a textual unit, which in our case is a sentence. Our problem then is to select S to maximize $f_{QSBP}(S)$. The above function based on word pairs is still submodular, and therefore we can apply a greedy approximate algorithm with performance guarantee as proposed in previous work (Khuller et al., 1999; Takamura and Okumura, 2009a). Let $l(u)$ denote the length of u . Given a set of source documents D and a length limit L for a summary,

Require: D, L

- 1: $W = D, S = \phi$
- 2: **while** $W \neq \phi$ **do**
- 3: $u = \arg \max_{u \in W} \frac{f(S \cup \{u\}) - f(S)}{l(u)}$
- 4: **if** $l(u) + \sum_{u_S \in S} l(u_S) \leq L$ **then**
- 5: $S = S \cup \{u\}$
- 6: **end if**
- 7: $W = W / \{u\}$
- 8: **end while**
- 9: $u_{max} = \arg \max_{u \in D} f(u)$
- 10: **if** $f(u_{max}) > f(S)$ **then**
- 11: **return** u_{max}
- 12: **else return** S
- 13: **end if**

where $f(\cdot)$ is some score function such as f_{QSBP} . We call our proposed method QSBP: Query Snowball with Word Pairs.

4 Experiments

4.1 Experimental Environment

	ACLIA1		ACLIA2
	Development	Test	Test
#of questions	101	100	80*
#of avg. nuggets	5.8	12.8	11.2*
Question types	DEFINITION, BIOGRAPHY, RELATIONSHIP, EVENT		+WHY
Articles years	1998-2001		2002-2005
Documents	Mainichi Newspaper		

*After removing the factoid questions.

Table 1: ACLIA dataset statistics

We evaluate our method using Japanese QA test collections from NTCIR-7 ACLIA1 and NTCIR-8 ACLIA2 (Mitamura et al., 2008; Mitamura et al., 2010). The collections contain complex questions and their answer nuggets with weights. Table 1 shows some statistics of the data. We use the ACLIA1 development data for tuning a parameter for our baseline as shown in Section 4.2 (whereas our proposed method is parameter-free), and the ACLIA1 and ACLIA2 test data for evaluating different methods. The results for the ACLIA1 test data are omitted due to lack of space. As our aim is to answer complex questions by means of multi-document summarization, we removed factoid questions from the ACLIA2 test data.

Although the ACLIA test collections were originally designed for Japanese QA evaluation, we treat them as query-oriented summarization test collections. We use all the candidate documents from which nuggets were extracted as input to the multi-document summarizers. That is, in our problem setting, the relevant documents are already given, although the given document sets also occasionally contain documents that were eventually never used for nugget extraction (Mitamura et al., 2008; Mitamura et al., 2010).

We preprocessed the Japanese documents basically by automatically detecting sentence boundaries based on Japanese punctuation marks, but we also used regular-expression-based heuristics to detect glossary of terms in articles. As the descriptions of these glossaries are usually very useful for answering BIOGRAPHY and DEFINITION questions, we treated each term description (generally multiple sentences) as a single sentence.

We used Mecab (Kudo et al., 2004) for morphological analysis, and calculated base word scores $s_b(w)$ using Mainichi articles from 1991 to 2005. We also used Mecab to convert each word to its base form and to filter using POS tags to extract content words. As for dependency parsing for distance computation, we used Cabocha (Kudo and Matsumoto, 2000). We did not use a stop word list or any other external knowledge.

Following the NTCIR-9 one click access task setting¹, we aimed at generating summaries of Japanese 500 characters or less. To evaluate the summaries, we followed the practices at the TAC summarization tasks (Dang, 2008) and NTCIR ACLIA tasks, and computed pyramid-based precision with an allowance parameter of C , recall, $F\beta$ (where β is 1 or 3) scores. The value of C was determined based on the average nugget length for each question type of the ACLIA2 collection (Mitamura et al., 2010). Precision and recall are computed based on the nuggets that the summary covered as well as their weights. The first author of this paper manually evaluated whether each nugget matches a summary. The evaluation metrics are formally defined as follows:

$$\begin{aligned}
 precision &= \min\left(\frac{C \cdot (\# \text{ of matched nuggets})}{\text{summary length}}, 1\right), \\
 recall &= \frac{\text{sum of weights over matched nuggets}}{\text{sum of weights over all nuggets}}, \\
 F\beta &= \frac{(1 + \beta^2) \cdot precision \cdot recall}{\beta^2 \cdot precision + recall}.
 \end{aligned}$$

4.2 Baseline

MMR is a popular approach in query-oriented summarization. For example, at the TAC 2008 opinion summarization track, a top performer in terms of pyramid F score used an MMR-based method. Our own implementation of an MMR-based baseline uses an existing algorithm to maximize the following summary set score function (Lin and Bilmes, 2010):

$$\begin{aligned}
 f_{MMR}(S) &= \gamma \left(\sum_{u \in S} Sim(u, v_D) + \sum_{u \in S} Sim(u, v_Q) \right) \\
 &\quad - (1 - \gamma) \sum_{\{(u_i, u_j) | i \neq j \text{ and } u_i, u_j \in S\}} Sim(u_i, u_j) \quad (4)
 \end{aligned}$$

where v_D is the vector representing the source documents, v_Q is the vector representing the query terms, Sim is the cosine similarity, and γ is a parameter.

¹<http://research.microsoft.com/en-us/people/tesakai/1click.aspx>

Thus, the first term of this function reflects how the sentences reflect the entire documents; the second term reflects the relevance of the sentences to the query; and finally the function penalizes redundant sentences. We set γ to 0.8 and the scaling factor used in the algorithm to 0.3 based on a preliminary experiment with a part of the ACLIA1 development data. We also tried incorporating sentence position information (Radev, 2001) to our MMR baseline but this actually hurt performance in our preliminary experiments.

4.3 Variants of the Proposed Method

To clarify the contributions of each components, the minimum dependency distance, QSB and the word pair, we also evaluated the following simplified versions of QSBP. (We use the itf version by default, and will refer to the idf version as QSBP(idf).) To examine the contribution of using minimum dependency distance, We remove $distance(w, w')$ from Eq.1 and Eq.2. We call the method QSBP(nodist). To examine the contribution of using word pairs for score maximization (see Section 3.2) on the performance of QSBP, we replaced Eq.3 with:

$$f_{QSB}(S) = \sum_{\{w|w \in u_i \text{ and } u_i \in S\}} s_r(w). \quad (5)$$

To examine the contribution of the QSB relevance scoring (see Section 3.1) on the performance of QSBP, we replaced Eq.3 with:

$$f_{WP}(S) = \sum_{\{w_1, w_2 | w_1 \neq w_2 \text{ and } w_1, w_2 \in u_i \text{ and } u_i \in S\}} s_b(w_1) s_b(w_2). \quad (6)$$

We will refer to this as WP. Note that this relies only on base word scores and is query-independent.

4.4 Results

Tables 2 and 3 summarize our results. We used the two-tailed sign test for testing statistical significance. Significant improvements over the MMR baseline are marked with a \dagger ($\alpha=0.05$) or a \ddagger ($\alpha=0.01$); those over QSBP(nodist) are marked with a $\#$ ($\alpha=0.05$) or a $\#\#$ ($\alpha=0.01$); and those over QSB are marked with a \bullet ($\alpha=0.05$) or a $\bullet\bullet$ ($\alpha=0.01$); and those over WP are marked with a \star ($\alpha=0.05$) or a $\star\star$ ($\alpha=0.01$). From Table 2, it can be observed that both QSBP and QSBP(idf) significantly outperforms QSBP(nodist), QSB, WP and the baseline in terms of all evaluation metrics. Thus, the minimum dependency distance, Query Snowball and the use of word

pairs all contribute significantly to the performance of QSBP. Note that we are using the ACLIA data as summarization test collections and that the official QA results of ACLIA should not be compared with ours.

QSBP and QSBP(idf) achieve 0.312 and 0.313 in F3 score, and the differences between the two are not statistically significant. Table 3 shows the F3 scores for each question type. It can be observed that QSBP is the top performer for BIO, DEF and REL questions on average, while QSBP(idf) is the top performer for EVENT and WHY questions on average. It is possible that different word scoring methods work well for different question types.

Method	Precision	Recall	F1 score	F3 score
Baseline	0.076 \star	0.370 \star	0.116 \star	0.231 \star
QSBP	0.107 $\dagger\bullet\bullet\#\#\#$	0.482 $\dagger\bullet\bullet\#\#\#$	0.161 $\dagger\bullet\bullet\#\#\#$	0.312 $\dagger\bullet\bullet\#\#\#$
QSBP(idf)	0.106 $\dagger\bullet\bullet\#\#\#$	0.485 $\dagger\bullet\bullet\#\#\#$	0.161 $\dagger\bullet\bullet\#\#\#$	0.313 $\dagger\bullet\bullet\#\#\#$
QSBP(nodist)	0.083 $\dagger\star$	0.396 $\dagger\star$	0.125 $\dagger\star$	0.248 $\dagger\star$
QSB	0.086 $\dagger\star$	0.400 $\dagger\star$	0.129 $\dagger\star$	0.253 $\dagger\star$
WP	0.053	0.222	0.080	0.152

Table 2: ACLIA2 test data results

Type	BIO	DEF	REL	EVENT	WHY
Baseline	0.207 \star	0.251 \star	0.270	0.212	0.213
QSBP	0.315 $\bullet\bullet$	0.329 $\dagger\bullet\bullet\#\#\#$	0.401 \dagger	0.258 $\dagger\bullet\bullet\#\#\#$	0.275 $\#\#$
QSBP(idf)	0.304 $\#\#\#\#$	0.328 $\dagger\bullet\bullet\#\#\#$	0.397 \dagger	0.268 $\dagger\bullet\bullet\#\#\#$	0.280 $\#\#\#$
QSBP(nodist)	0.255	0.281 $\dagger\star$	0.329	0.196	0.212 $\dagger\star$
QSB	0.245 $\dagger\star$	0.273 $\dagger\star$	0.324	0.217	0.215
WP	0.109	0.037	0.235	0.141	0.161

Table 3: F3-scores for each question type (ACLIA2 test)

5 Conclusions and Future work

We proposed the Query Snowball (QSB) method for query-oriented multi-document summarization. To enrich the information need representation of a given query, QSB obtains words that augment the original query terms from a co-occurrence graph. We then formulated the summarization problem as an MCKP based on word pairs rather than single words. Our method, QSBP, achieves a pyramid F3-score of up to 0.313 with the ACLIA2 Japanese test collection, a 36% improvement over a baseline using Maximal Marginal Relevance.

Moreover, as the principles of QSBP are basically language independent, we will investigate the effectiveness of QSBP in other languages. Also, we plan to extend our approach to abstractive summarization.

References

- Wauter Bosma. 2009. Contextual salience in query-based summarization. In *Proceedings of the International Conference RANLP-2009*, pages 39–44. Association for Computational Linguistics.
- Jaime Carbonell and Jade Goldstein. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '98*, pages 335–336. Association for Computational Machinery.
- Asli Celikyilmaz and Dilek Hakkani-Tur. 2010. A hybrid hierarchical model for multi-document summarization. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, ACL '10*, pages 815–824. Association for Computational Linguistics.
- Hoa Trang Dang. 2008. Overview of the tac 2008 opinion question answering and summarization tasks. In *Proceedings of Text Analysis Conference*.
- Elena Filatova and Vasileios Hatzivassiloglou. 2004. A formal model for information selection in multi-sentence text extraction. In *Proceedings of the 20th international conference on Computational Linguistics, COLING '04*. Association for Computational Linguistics.
- Takaaki Hasegawa, Hitoshi Nishikawa, Kenji Imamura, Genichiro Kikui, and Manabu Okumura. 2010. A Web Page Summarization for Mobile Phones. *Transactions of the Japanese Society for Artificial Intelligence*, 25:133–143.
- Jagadeesh Jagarlamudi, Prasad Pingali, and Vasudeva Varma. 2005. A relevance-based language modeling approach to duc 2005. In *Proceedings of Document Understanding Conferences (along with HLT-EMNLP 2005)*.
- Samir Khuller, Anna Moss, and Joseph S. Naor. 1999. The budgeted maximum coverage problem. *Information Processing Letters*, 70(1):39–45.
- Taku Kudo and Yuji Matsumoto. 2000. Japanese dependency structure analysis based on support vector machines. In *Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics*, volume 13, pages 18–25. Association for Computational Linguistics.
- Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. 2004. Applying conditional random fields to Japanese morphological analysis. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)*, volume 2004, pages 230–237.
- Wenjie Li, You Ouyang, Yi Hu, and Furu Wei. 2008. PolyU at TAC 2008. In *Proceedings of Text Analysis Conference*.
- Hui Lin and Jeff Bilmes. 2010. Multi-document summarization via budgeted maximization of submodular functions. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, HLT '10*, pages 912–920. Association for Computational Linguistics.
- Hui Lin, Jeff Bilmes, and Shasha Xie. 2010a. Graph-based submodular selection for extractive summarization. In *Automatic Speech Recognition & Understanding, 2009. ASRU 2009. IEEE Workshop on*, pages 381–386. IEEE.
- Jimmy Lin, Nitin Madnani, and Bonnie J. Dorr. 2010b. Putting the user in the loop: interactive maximal marginal relevance for query-focused summarization. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, HLT '10*, pages 305–308. Association for Computational Linguistics.
- Fei Liu and Yang Liu. 2009. From extractive to abstractive meeting summaries: can it be done by sentence compression? In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers, ACLShort '09*, pages 261–264. Association for Computational Linguistics.
- Kevin Lund and Curt Burgess. 1996. Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior Research Methods*, 28:203–208.
- Inderjeet Mani. 2001. *Automatic summarization*. John Benjamins Publishing Co.
- Ryan McDonald. 2007. A study of global inference algorithms in multi-document summarization. In *Proceedings of the 29th European conference on IR research, ECIR'07*, pages 557–564. Springer-Verlag.
- Teruko Mitamura, Eric Nyberg, Hideki Shima, Tsuneaki Kato, Tatsunori Mori, Chin-Yew Lin, Ruihua Song, Chuan-Jie Lin, Tetsuya Sakai, Donghong Ji, and Noriko Kando. 2008. Overview of the NTCIR-7 ACLIA tasks: Advanced cross-lingual information access. In *Proceedings of the 7th NTCIR Workshop*.
- Teruko Mitamura, Hideki Shima, Tetsuya Sakai, Noriko Kando, Tatsunori Mori, Koichi Takeda, Chin-Yew Lin, Ruihua Song, Chuan-Jie Lin, and Cheng-Wei Lee. 2010. Overview of the NTCIR-8 ACLIA tasks: Advanced cross-lingual information access. In *Proceedings of the 8th NTCIR Workshop*.
- Jahn Otterbacher, Güneş Erkan, and Dragomir R. Radev. 2005. Using random walks for question-focused sentence retrieval. In *Proceedings of the conference on Human Language Technology and Empirical Methods*

- in Natural Language Processing, HLT '05*, pages 915–922. Association for Computational Linguistics.
- Dragomir R. Radev. 2001. Experiments in single and multidocument summarization using mead. In *First Document Understanding Conference*.
- Hiroya Takamura and Manabu Okumura. 2009a. Text summarization model based on maximum coverage problem and its variant. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 781–789. Association for Computational Linguistics.
- Hiroya Takamura and Manabu Okumura. 2009b. Text summarization model based on the budgeted median problem. In *Proceeding of the 18th ACM conference on Information and knowledge management, CIKM '09*, pages 1589–1592. Association for Computing Machinery.
- Ramakrishna Varadarajan and Vagelis Hristidis. 2006. A system for query-specific document summarization. In *Proceedings of the 15th ACM international conference on Information and knowledge management, CIKM '06*, pages 622–631. ACM.
- Ellen M. Voorhees. 2003. Overview of the TREC 2003 Question Answering Track. In *Proceedings of the Twelfth Text REtrieval Conference (TREC 2003)*, pages 54–68.
- Wen-tau Yih, Joshua Goodman, Lucy Vanderwende, and Hisami Suzuki. 2007. Multi-document summarization by maximizing informative content-words. In *Proceedings of the 20th international joint conference on Artificial intelligence*, pages 1776–1782. Morgan Kaufmann Publishers Inc.