Improving Chronological Ordering of Sentences Extracted from Multiple Newspaper Articles

NAOAKI OKAZAKI
The University of Tokyo
YUTAKA MATSUO
Cyber Assist Research Center, AIST Tokyo Waterfront
and
MITSURU ISHIZUKA
The University of Tokyo

It is necessary to determine a proper arrangement of extracted sentences to generate a well-organized summary from multiple documents. This paper describes our Multi-Document Summarization (MDS) system for TSC-3. It specifically addresses an approach to coherent sentence ordering for MDS. An impediment to the use of chronological ordering, which is widely used by conventional summarization system, is that it arranges sentences without considering the presupposed information of each sentence. We propose a method to improve chronological ordering by resolving precedent information of arranging sentences. Combining the refinement algorithm with topical segmentation and chronological ordering, we address our experiments and metrics to test the effectiveness of MDS tasks. Results demonstrate that the proposed method significantly improves chronological sentence ordering. At the end of the paper, we also report an outline/evaluation of important sentence extraction and redundant clause elimination integrated in our MDS system.

Categories and Subject Descriptors: I.7.2 [**Text Processing**]: Document Preparation—*languages*; *photocomposition*

General Terms: Algorithms

Additional Key Words and Phrases: Multi-document summarization, sentence ordering, order, arrange, coherence

Authors' addresses: N. Okazaki, Graduate School of Information Science and Technology, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan; email: okazaki@mi.ci.i.u-tokyo.ac.jp; Y. Matsuo, Cyber Assist Research Center, National Institute of Advanced Industrial Science and Technology (AIST) Tokyo Waterfront, 2-41-6 Aomi, Koto-ku, Tokyo 135-0064, Japan; email: y.matsuo@aist.go.jp; M. Ishizuka, Graduate School of Information Science and Technology, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan; email: ishizuka@i.u-tokyo.ac.jp.

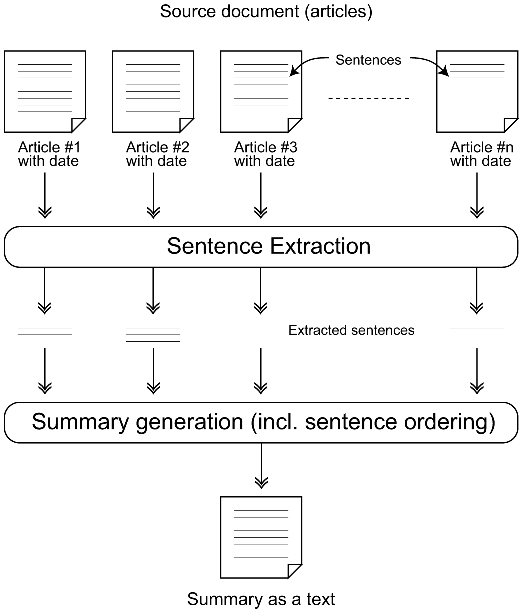


Fig. 1. A simplified MDS system.

1. INTRODUCTION

Numerous computerized documents are accessible on-line. With the help of search engines, we can obtain relevant documents that fit our interests. Notwithstanding, we are often disappointed with the quantity of retrieved documents despite having narrowed the range of documents to be read through the search phase. It is necessary to establish technologies that facilitate information utilization because many documents are gathered dynamically by a user. Automatic text summarization [Mani 2001] is a challenge to the information overload problem. Text summarization provides a condensed text for a given document. Multi-Document Summarization (MDS) [Radev and McKeown 1998], which is an extension of summarization to related documents, has attracted much attention in recent years.

Figure 1 illustrates an example of a typical MDS system. Given a number of documents, an MDS system yields a summary by gathering information from original documents. When a user wishes to gain an understanding of retrieved documents, the summary should include as much important information as possible. Therefore, to produce an effective summary, an MDS system should identify information in source documents to determine which information is important for inclusion and which information is unimportant or redundant. Most existing summarization systems make use of such extraction techniques to find relevant textual segments in source documents. Numerous studies have examined extraction since the early stage of natural language processing [Luhn 1958] because the quality of extraction greatly affects overall performance of MDS systems.

However, to maintain the readability of extracts, we need to ensure that sentences in the extracts are in proper order. Barzilay et al. [2002] conducted experiments to examine the impact of sentence ordering on summary readability. That study showed that sentence ordering markedly affects readers' text comprehension. Sentence position in the original document, which yields a good clue to sentence arrangement for single-document summarization, is insufficient for multi-document summarization because we must simultaneously consider interdocument order. For this reason, it is necessary to establish a good ordering strategy for MDS.

From among components in our MDS system for TSC-3, this paper specifically examines a method to arrange sentences that are extracted by important sentence extraction. This paper is organized as follows. The following section (Section 2) reviews the sentence ordering problem in MDS and previous attempts to tackle the problem. Section 3 describes the issue of chronological ordering and presents our approach to generate an acceptable ordering by resolving antecedent information. The subsequent section (Section 4) addresses our experiments and evaluation metrics to test the effectiveness of the proposed method. After reporting an outline/evaluation of other components in our MDS system, such as important sentence extraction and redundant clause elimination in Section 5, we conclude this paper.

2. SENTENCE ORDERING PROBLEM

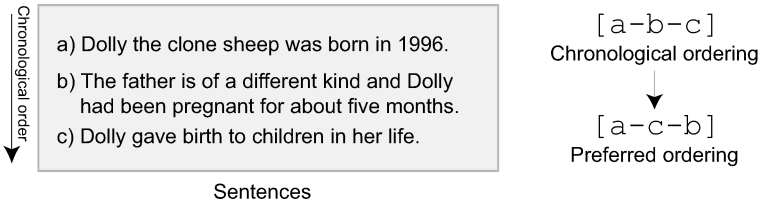
Our goal is to either determine a most probable permutation of sentences or to reconstruct a discourse structure of sentences gathered from multiple sources. When asked to arrange sentences, a human may perform such a task without difficulty just as we put our thought in writing. However, we must consider what accomplishes this task, because computers are, by their nature, unaware of ordering. Discourse coherence, typified by rhetorical relation [Mann and Thompson 1988] and coherence relation [Hobbs 1990], are helpful to resolve this question. Hume [1748] claimed that qualities from which association arises and by which the mind is conveyed from one idea to another are three: *resemblance*, *contiguity in time or place*, and *cause and effect*. That is to say, we should organize a text from fragmented information on the basis of topical relevancy, chronological and spatial orders, and a cause–effect relation. The fact is especially true for sentence ordering of newspaper articles, because we must typically arrange a large number of time-series events that are related to several topics.

The strategy for sentence ordering that most MDS systems use is *chronological ordering* [McKeown et al. 1999; Lin and Hovy 2001], which arranges sentences in the order of their publication dates. Barzilay et al. [2002] addressed the problem of sentence ordering in the context of multi-document summarization. They first demonstrated the remarkable impact of sentence ordering on summary readability through their experiment with human subjects. They also used human experiments to identify orderings of patterns that can improve two naive sentence-ordering techniques, such as majority ordering (examines ordering according to relative frequency in the original documents) and chronological ordering. Based on those experiments, they proposed a strategy that combines constraints from chronological order of events and topical relatedness. Evaluation in which they asked human judges to grade summaries showed remarkably improved quality of orderings from the chronological ordering to the proposed method.

Lapata [2003] proposed an approach to information ordering. She introduced three assumptions for learning constraints on sentence order from a corpus of domain specific texts: the probability of any sentence is dependent on its previously arranged sentences; the probability of any given sentence is determined only by its previous sentence; and transition probability from a sentence to its subsequent sentence is estimated by the Cartesian product defined over the features expressing the sentences. For describing sentences by their features, she used verbs (precedent relationships of verbs in the corpus), nouns (entity-based coherence by keeping track of the nouns), and dependencies (structure of sentences). Lapata also proposed the use of Kendall's rank coefficient for an automatic evaluation that quantifies the difference between orderings produced by the proposed method and a human. Although she did not describe performance comparison of the proposed method with chronological ordering, her approach is applicable to documents without publication dates.

Barzilay and Lee [2004] investigated the utility of domain-specific *content structure* for representing topics and topic shifts. They applied content models to sentence ordering and extractive summarization. Content models are Hidden Markov Models (HMMs) wherein states correspond to types of information characteristics to the domain of interest (e.g., earthquake magnitude or previous earthquake occurrences) and state transitions capture possible information-presentation orderings within the domain. They employed an EM-like Viterbi reestimation procedure that repeats: *creating topical clusters of text spans* and *computing models of word distributions and topic changes from the clusters*. Creating initial topical clusters by complete-link clustering via sentence similarity (cosine coefficient of word bigrams), they constructed a content model: a state represents a topical cluster; the state's sentence-emission probabilities are estimated as the product of word-bigram probabilities; and state-transition probability are estimated by how sentences from the same article are distributed across clusters. Barzilay and colleagues conducted an experiment of ordering sentences that were unseen in test texts and arranged in the actual text. They proposed the use of an *original-source-order (OSO) prediction rate*, which measures the percentage of test cases in which the model under consideration yields the highest probability to the OSO from among all possible permutations, along with Kendall's metric. The evaluation result showed that their method outperformed Lapata's method [2003] by a wide margin. They did not address performance comparison with chronological ordering because they did not apply their approach to sentence ordering for MDS.

These previous attempts could be classified into two groups: use of chronological information [McKeown et al. 1999; Lin and Hovy 2001; Barzilay et al. 2002]; and learning natural ordering of sentences from large corpora [Lapata 2003; Barzilay and Lee 2004]. Advantages of the former group are that such methods are fast, easy-to-implement, and amenable to processing of newspaper articles.


a) Dolly the clone sheep was born in 1996.

b) The father is of a different kind and Dolly had been pregnant for about five months.

c) Dolly gave birth to children in her life.

Fig. 2. A problem case of chronological sentence ordering.

Methods of the latter group are applicable to various source documents, including newspaper articles. Against the background of these studies, we propose the use of antecedent sentences for coherent arrangement of sentences, which integrates ideas of the above two approaches. We take Barzilay's chronological ordering with topical segmentation as a starting point for newspaper articles. We consider its practical refinement using in-document preceding sentences for an evaluation criterion to arrange a sentence.

3. IMPROVING CHRONOLOGICAL ORDERING

3.1 Overview of the Proposed Method

Let us consider the example shown in Figure 2. There are three sentences, a, b, and c, that are extracted from different documents and refer to the clone sheep Dolly. Suppose that we infer an order [a-b-c] by chronological ordering. When we read these sentences in this order, we find that sentence b is positioned incorrectly because sentence b is written on the presupposition that the reader may know that Dolly had a child. An interpretation of this situation is that there were some precedent sentences prior to sentence b in the original document, but sentence extraction did not choose such sentences as summary candidates. Lack of presupposition obscures what a sentence is intended to convey, thereby confusing readers. Although we may hit upon a possible solution by which we include such preceding sentences into summary candidates as an exceptional case, the solution is not appropriate in terms of stability (i.e., preceding sentences are not always required) and redundancy (i.e., including sentences may generate redundant summaries). Hence, we exclude that approach to expand the output of the sentence extraction, which is presumed to be tuned independently.

We observe the example in Figure 2 again. When reading sentence c, we note that it can include presuppositional information of sentence b. In addition, sentence c also requires no presupposition other than Dolly's existence, which was already mentioned in sentence a. Based on the analyses, we can refine the chronological order and revise the order to [a-c-b], putting sentence c before sentence b. This revision enables us to assume sentence b to be an elaboration of sentence c; thereby, we improve summary readability.

The rest of this section addresses improvement of chronological ordering using in-document preceding sentences followed by a detailed description of chronological ordering itself. Then we describe the entire algorithm along with topical segmentation.

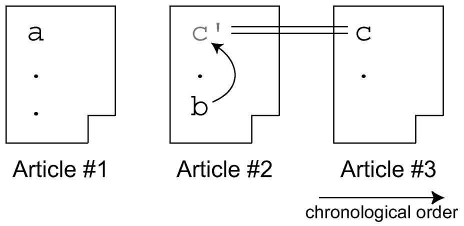


Fig. 3. Background idea of ordering refinement by precedence relation.

3.2 Chronological Ordering

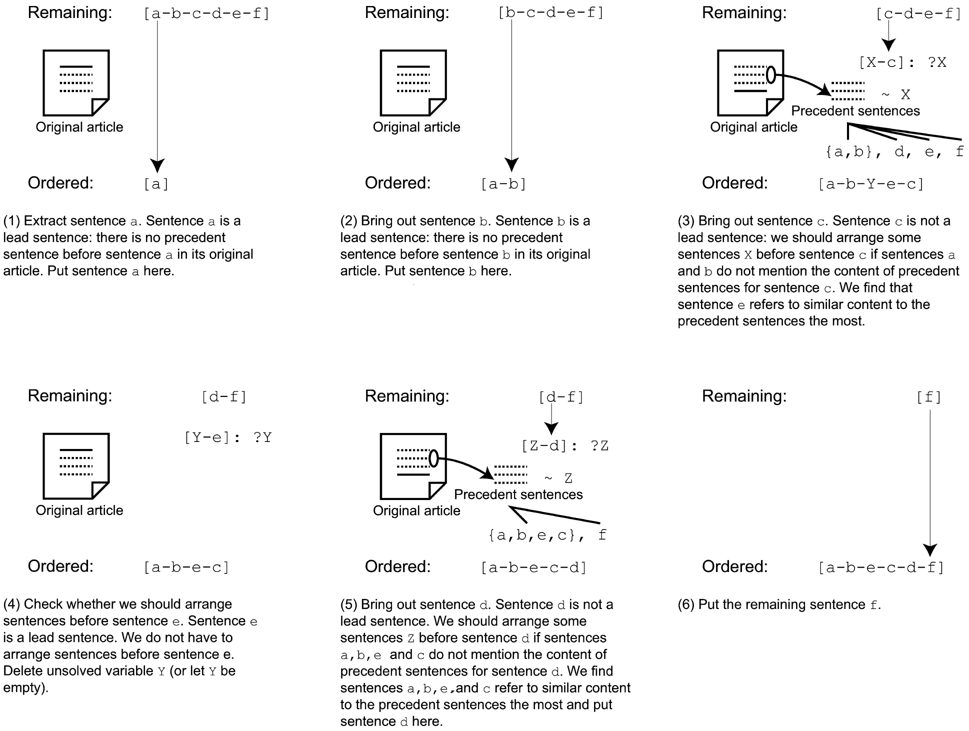
It is difficult for computers to find a resemblance or cause–effect relation between two phenomena: numerous possible relations must be classified in detail; moreover, we do not have conclusive evidence whether a pair of sentences that we arbitrarily gather from multiple documents have some relation. A newspaper usually disseminates descriptions of novel events that have occurred since the last publication. Hence, the publication date (time) of each article turns out to be a good estimator of the resemblance relation (i.e., we observe a trend or series of relevant events in a time period), contiguity in time, and a cause–effect relation (i.e., an event occurs as a result of previous events). [Lin and Hovy 2001, 2002] constructed a multi-document summarization system (NeATS) and arranged sentences in chronological order. They also resolved relative time expressions[1] using rules for actual date estimation. Although resolving temporal expressions in sentences [Mani and Wilson 2000; Mani et al. 2003] may allow more precise estimation of sentence relations, it is not an easy task. For this reason, we first order sentences in chronological order, assigning a time stamp for each sentence by its publication date (i.e., the date when the article appeared in the paper).

For sentences having the same time stamp, we generate the order based on the sentence position and connectivity. We restore an original ordering if two sentences have the same time stamp and belong to the same article. If sentences have the same time stamp and are not from the same article, we insert a sentence that is more similar to previously ordered sentences to assure sentence connectivity.

3.3 Improving Chronological Ordering

After we obtain a chronological order of sentences, we make an effort to improve the ordering with the help of antecedent sentences. Figure 3 shows the background idea of ordering refinement using a precedence relation. Just as in the example shown in Figure 2, we have three sentences a, b, and c in chronological order. First, we select sentence a out of the sentences and check its antecedent sentences. Seeing that there are no sentences prior to sentence a in article #1, we deem it acceptable to put sentence a here. Then we select sentence b from

[1] Newspaper articles often use relative date expressions such as “Monday” or “yesterday.” If these expressions were not replaced with actual dates, the summary might mislead the reader because they might lose absolute time references during sentence extraction.



(1) Extract sentence a. Sentence a is a lead sentence: there is no precedent sentence before sentence a in its original article. Put sentence a here.

(2) Bring out sentence b. Sentence b is a lead sentence: there is no precedent sentence before sentence b in its original article. Put sentence b here.

(3) Bring out sentence c. Sentence c is not a lead sentence: we should arrange some sentences X before sentence c if sentences a and b do not mention the content of precedent sentences for sentence c. We find that sentence e refers to similar content to the precedent sentences the most.

(4) Check whether we should arrange sentences before sentence e. Sentence e is a lead sentence. We do not have to arrange sentences before sentence e. Delete unsolved variable Y (or let Y be empty).

(5) Bring out sentence d. Sentence d is not a lead sentence. We should arrange some sentences Z before sentence d if sentences a,b,e and c do not mention the content of precedent sentences for sentence d. We find sentences a,b,e,and c refer to similar content to the precedent sentences the most and put sentence d here.

(6) Put the remaining sentence f.

Fig. 4. Improving chronological ordering using antecedent sentences.

the remaining sentences and check its antecedent sentences. This time, we find several sentences before sentence b in article #2. Grasping what the antecedent sentences are saying by means of cosine similarity of sentence vectors, we confirm, first, whether their subject content is mentioned in previously arranged sentences (i.e., sentence a). If it is mentioned, we put sentence b here and extend the ordering to [a-b]. Otherwise, we search for a substitution for what the precedent sentences are saying from the remaining sentences (i.e., sentence c in this example). In the Figure 3 example, we find that sentence a is not referring to what sentence c' is saying, but that sentence c is approximately referring to that content. Putting sentence c before b, we finally achieve the refined ordering [a-c-b].

As the criterion for selecting the sentence to be inserted, we introduce *distance* to put a sentence after previously arranged sentences. We define the distance as dissimilarity derived from cosine similarity between a vector of the arranging sentence and a vector of its preceding sentences. When a sentence has preceding sentences and their content is not mentioned by previously arranged sentences, this *distance* will be high. When a sentence has no precedent sentences, we define the *distance* to be 0.

Figure 4 illustrates how our algorithm refines a given chronological ordering [a-b-c-d-e-f]. In the Figure 4 example, we do not change the position of sentences a and b because they do not have precedent sentences in their original article (i.e., they are lead sentences[2]). On the other hand, sentence c has some preceding sentences in its original document. This fact presents us with a choice: we should check whether it is safe to put sentence c just after sentences a and b; or we should arrange some sentences before sentence c as a substitute for the precedent sentences. Preparing a term vector of the precedent sentences, we seek a sentence or a set of sentences that is the closest to the precedent content in sentences {a,b}, d, e, and f by the *distance* measure defined above. In other words, we assume sentence ordering to be [a-b-X-c] and find appropriate sentence(s) X, if any. Supposing that sentence e in Figure 4 describes similar content as the precedent sentences for sentence c, we substitute X with Y-e. We then check whether we should put some sentences before sentence e or not. Given that sentence e is a lead sentence, we leave Y as empty (i.e., *distance* is 0) and fix the resultant ordering to [a-b-e-c].

Then we consider sentence d, which, again, is not a lead sentence. Preparing a term vector of the precedent sentences of sentence d, we search for a sentence or a set of sentences which is closest to the precedent content in sentences {a,b,e,c}, f. Supposing that either sentence a, b, e, or c refers to the precedent content closer than sentence f, we make a decision to put sentence d here. In this way, we get the final ordering: [a-b-e-c-d-f].

3.4 Compatibility with Multi-Document Summarization

We describe briefly how our ordering algorithm functions jointly with MDS. Let us reconsider the example shown in Figure 3. In this example, sentence extraction does not select sentence c'; sentence c is very similar to sentence c'. This may appear to be a rare case for explanation, but it could happen, as we optimize a sentence-extraction method for MDS. A method for MDS (e.g., the method described in Section 2, MMR-MD [Carbonell and Goldstein 1998]) makes an effort to acquire information coverage under the condition that a number of sentences exist as summary candidates. This is to say that an extraction method should be capable of refusing redundant information.

When we collect articles that describe a series of events, we may find that lead sentences convey similar information throughout the articles, because the major task of lead sentences is to give a subject. Therefore, it is quite natural that: lead sentences c and c' refer to similar content; an extraction method for MDS does not choose both sentence c' and c in terms of redundancy; and the method also prefers either sentence c or c' in terms of information coverage.

3.5 Implementation

Figure 5 depicts a block diagram of the sentence ordering algorithm. Given nine sentences denoted by (a b...i), the algorithm eventually produces an ordering: [a-b-f-c-i-g-d-h-e].

We categorize sentences by their topics in the first phase. The aim of this phase is to group topically related sentences together. It was applied to sentence ordering by Barzilay et al. [2002]. We use the vector space model

[2] *Lead sentences* are sentences which appear at the beginning of an article.

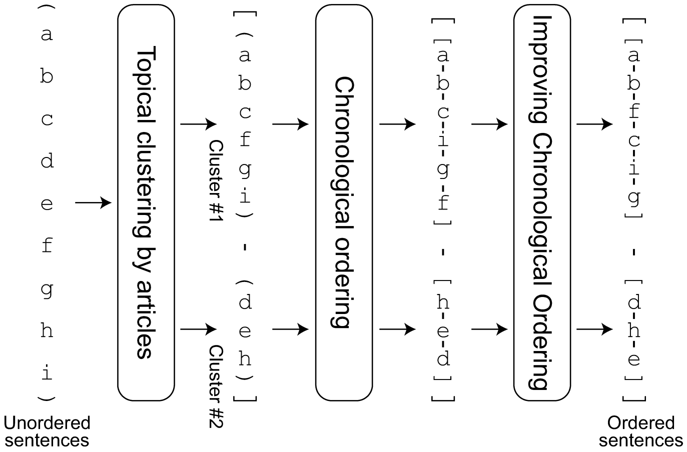

(a b c d e f g h i)
Unordered sentences
Topical clustering by articles
[(a b c f g i) - (d e h)]
Cluster #1
Cluster #2
Chronological ordering
[[a-b-c-i-g-f] - [h-e-d]]
Improving Chronological Ordering
[[a-b-f-c-i-g] - [d-h-e]]
Ordered sentences

Fig. 5. Outline of the ordering algorithm.

[Salton et al. 1975] for sentence representation and apply the nearest-neighbor method [Cover and Hart 1967] to obtain topical clusters. Because sentences in newspaper articles are not always long enough to represent their contents in sentence vectors, we assume that a newspaper article is written for one topic and thereby classify document vectors. Given l articles and m kinds of terms in the articles, we define a document-term matrix D ($l \times m$), whose element D_{ij} represents the frequency of term j in document i,

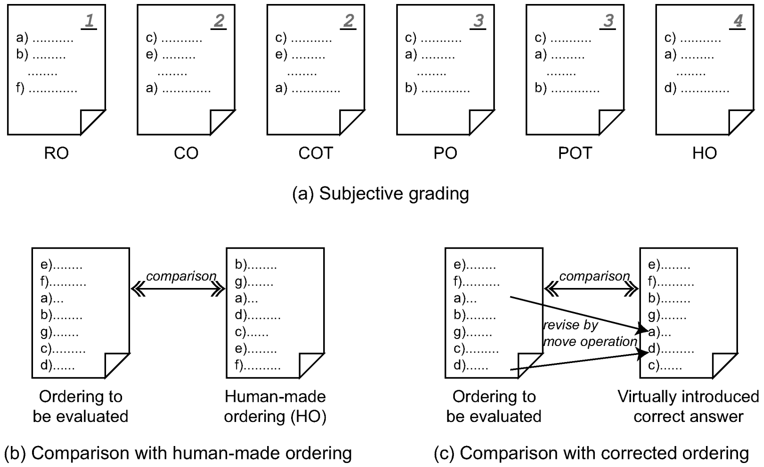
$$D_{ij} = (\text{number of occurrences of term } j \text{ in document } i) \tag{1}$$

Letting D_i denote a term vector (i-component row vector) of document i, we measure the distance or dissimilarity between two articles x and y using a cosine coefficient:

$$\text{distance}(D_x, D_y) = 1 - \frac{D_x \cdot D_y}{|D_x||D_y|} \tag{2}$$

We apply the nearest-neighbor method [Cover and Hart 1967] to merge a pair of articles when their minimum distance is lower than a given parameter $\alpha = 0.3$ (determined empirically). In this manner, we classify sentences according to topical clusters of articles. We determine an order of clusters based on the chronological order of the first publication date of articles in each cluster.

The rest phases of the algorithm, *chronological ordering* and *improving chronological ordering* that we described before, treat the partitioned sentences independently. We arrange sentences within respective topical clusters. In Figure 5 example, we obtain two topical clusters, (a b c f g i) and (d e h), as the output from the topical clustering. The second phase orders sentences in each topical group by the chronological order and sends two orderings, [a-b-c-i-g-f] and [h-e-d], to the third phase. The third phase refines each chronological ordering by the proposed method and outputs the final ordering: [a-b-f-c-i-g-d-h-e].



(a) Subjective grading

(b) Comparison with human-made ordering

(c) Comparison with corrected ordering

Fig. 6. Three evaluation tasks for sentence ordering.

4. EVALUATION

4.1 Experiment

We conducted an experiment of sentence ordering through multi-document summarization to test the effectiveness of the proposed method. Extracting sentences up to a specified number (ca. 10% summarization rate), we created a set of candidate summary sentences for each task. We order the sentences by six methods: *human-made ordering (HO)* as the highest anchor; *random ordering (RO)* as the lowest anchor; *chronological ordering (CO)* as a conventional method; *chronological ordering with topical segmentation (COT)* [similar to Barzilay's et al. method 2002]; *the proposed method without topical segmentation (PO)*; and *the proposed method with topical segmentation (POT)*. Using 28 topics (summarization assignments)[3] in the TSC-3 test collection, we asked three human judges to evaluate these sentence orderings (i.e., each sentence ordering was assigned with three independent judgments). For each summarization topic, we presented six summaries generated by different methods in random order to prevent the bias during the experiment. We describe three tasks to measure the quality of orderings below.

The first evaluation task is a subjective grading by which a human judge marks an ordering of summary sentences on a scale of 1 to 4 (Figure 6(a)). We give clear criteria of scoring to the judges as follows. A perfect (score = 4) summary is a text that we cannot improve any further by reordering. An acceptable (score = 3) summary is one that makes sense and is unnecessary to revise even though there is some room for improvement in terms of readability. A poor summary (score = 2) is one that loses a thread of the story at some places and requires minor amendment to bring it up to an acceptable level. An unacceptable summary (score = 1) is one that leaves much to be improved

[3] We exclude 2 of 30 summaries because they are so long (ca. 30 sentences) that it is hard for judges to evaluate and revise them.

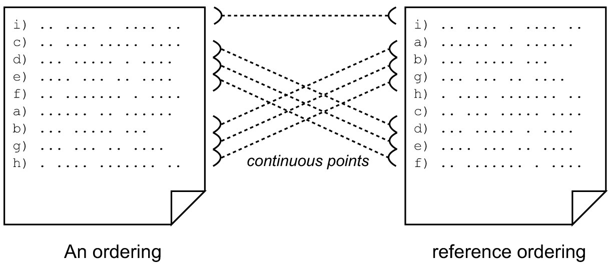


Fig. 7. An example of an ordering and its reference ordering.

and requires overall restructuring rather than partial revision. We inform the judges that summaries were made of the same set of extracted sentences and only sentence ordering made differences between the summaries to improve consistency in rating.

In addition to the rating, it is useful that we examine how close an ordering is to an acceptable one when the ordering is regarded as *poor*. We designed a second task that measures closeness of an ordering to a human-made one (Figure 6(b)). However, this task may be so simplistic that it cannot accept several sentence-ordering patterns for a given summary. We infer that it is valuable to measure the degree of correction because the task virtually requires a human corrector to mentally prepare a correct answer for each ordering. For this reason, we introduce another task in which a human judge is presumed to illustrate how to improve an ordering of a summary when he or she marks the summary as *poor* in the rating task. We restrict applicable operations of corrections to move operations to maintain minimum correction of the ordering. We define a move operation here as removing a sentence and inserting the sentence into an appropriate place (Figure 6(c)).

4.2 Evaluation Metrics

The remainder of the evaluation design entails the comparison of an ordering with its reference ordering. Figure 7 shows an ordering of nine sentences (denoted by `a, b,...,i`) and its reference (correct) ordering. Supposing a sentence ordering to be a rank, we can convert a sentence ordering into a permutation, which represents the rank of each sentence. Let π be a permutation of an ordering to be evaluated and σ be its reference ordering. Expressing sentences `a` in 1, `b` in 2, ..., `i` in 9 respectively, we obtain permutations π and σ for Figure 7:

$$\pi = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ 6 & 7 & 2 & 3 & 4 & 5 & 8 & 9 & 1 \end{pmatrix}, \quad \sigma = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ 2 & 3 & 6 & 7 & 8 & 9 & 4 & 5 & 1 \end{pmatrix} \tag{3}$$

The above formulation transforms closeness measurement of two orderings into calculation of rank correlation of two permutations π and σ. Spearman's rank correlation $\tau_s(\pi, \sigma)$ and Kendall's rank correlation $\tau_k(\pi, \sigma)$ are well-known rank correlation metrics:

$$\tau_s(\pi, \sigma) = 1 - \frac{6}{n(n+1)(n-1)} \sum_{i=1}^{n} (\pi(i) - \sigma(i))^2 \tag{4}$$

$$\tau_k(\pi, \sigma) = \frac{1}{n(n-1)/2} \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \mathrm{sgn}(\pi(j) - \pi(i)) \cdot \mathrm{sgn}(\sigma(j) - \sigma(i)) \tag{5}$$

Therein: n represents the number of sentences; and $\mathrm{sgn}(x) = 1$ for $x > 0$ and -1 otherwise. These metrics range from -1 (an inverse rank) to 1 (an identical rank) via 0 (a noncorrelated rank). For Formula 3, we obtain $\tau_s(\pi, \sigma) = -0.07$ and $\tau_k(\pi, \sigma) = 0.11$ (i.e., the two ranks are approximately noncorrelated). Spearman's rank correlation considers the absolute relation of ranking (i.e., absolute position of sentences), and Kendall's rank correlation considers the relative relation of ranking (i.e., relative position of pairs of sentences). Lapata [2003] and Barzilay and Lee [2004] adopted Kendall's rank correlation for their evaluations, considering that it can be interpreted as the minimum number of adjacent transpositions needed to bring an order to the reference order.

Let us carefully examine the orderings in Figure 7. Spearman's rank correlation and Kendall's rank correlation indicate that they are noncorrelated ranks. However, we notice that the reference ordering can be generated from the ordering by moving a *group* of sentences `c`, `d`, `e`, `f` to the position just after sentence `h`. Although a reader may find the group of sentences `c`, `d`, `e`, `f` to be incorrectly positioned, he or she does not lose the thread of the summary because sentences within two groups, (`c`, `d`, `e`, `f`) and (`a`, `b`, `g`, `h`), are arranged properly.

Sentences in a document are aligned one dimensionally: a reader brings together continuous sentences in a text into his or her mind and interprets their meaning. In other words, when reading a text, a reader prefers local cohesion or sentence continuity as a relative relation of discontiguous sentences. Kendall's rank correlation equally penalizes inverse ranks of sentence pairs that are mutually distant in rank (e.g., sentences `c` and `a`, `c` and `b`). Therefore, we propose another metric to assess the degree of *sentence continuity* in reading. We define sentence continuity as the number of continuous sentence pairs divided by the number of sentences:

$$\text{sentence\_continuity} = \begin{cases} (c+1)/n & \text{(if the first sentences are identical)} \\ c/n & \text{(otherwise)} \end{cases} \tag{6}$$

Therein, c represents the number of continuous sentence pairs. Although there is no sentence prior to the first sentences, we want to measure the appropriateness of the first sentence as a leading sentence.[4] Hence, we define sentence continuity of the first sentence as an agreement of the first sentences between an ordering and its reference. This metric ranges from 0 (no continuity) to 1 (identical). The summary in Figure 7 may interrupt a human's reading after sentences `i`, `f` as the human searches for the next sentence to read. We observe six continuities and an agreement of the first sentences and calculate sentence continuity: $7/9 = 0.78$.

[4] This can be also expressed as *continuity* in the everyday world of readers.

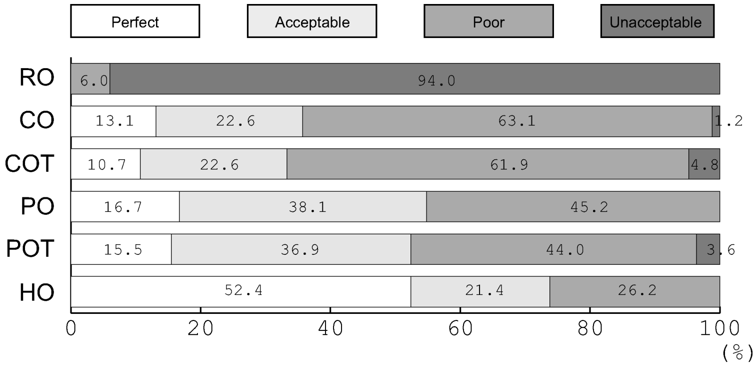


Fig. 8. Distribution of the rating score of orderings (percentage).

Sentence continuity can be expressed through permutations:

$$\tau_c(\pi, \sigma) = \frac{1}{n}\sum_{i=1}^{n} \text{equals}(\pi\sigma^{-1}(i), \pi\sigma^{-1}(i-1)+1). \tag{7}$$

Therein, $\pi(0) = \sigma(0) = 0$; equals$(x, y) = 1$ when x equals y and 0 otherwise. $\sigma^{-1}(i)$ represents a sentence (or index number) of the ith order in the reference; and $\pi\sigma^{-1}(i) = \pi(\sigma^{-1}(i))$ represents a rank in an ordering to be evaluated of the sentence arranged in the ith order in the reference. Hence, equals$(\pi\sigma^{-1}(i), \pi\sigma^{-1}(i-1)+1) = 1$ when sentences of $(i-1)$th and ith order in the reference are also continuous in an ordering.

4.3 Results

Figure 8 shows distribution of rating scores of each method as a percentage of 84 (28 × 3) summaries. Judges marked about 75% of human-made orderings (HOs) as either perfect or acceptable; they rejected as many as 95% of random orderings (ROs). Chronological ordering (CO) did not yield satisfactory results, losing a thread of 63% summaries, although CO performed much better than RO. Topical segmentation did not contribute to ordering improvement of CO either: COT was slightly worse than CO. After taking an in-depth look at the failure orderings, we found that topical clustering did not perform well during this test. We infer that topical clustering did not prove its merits with this test collection because the collection comprises relevant articles that were retrieved by some query and polished well by a human: they exclude articles that are unrelated to a topic. On the other hand, the proposed method (PO) improved chronological ordering much better than topical segmentation: the sum of the perfect and acceptable ratio jumped from 36 (CO) to 55% (PO). This fact shows that ordering refinement by precedence relation improves chronological ordering by pushing poor ordering to an acceptable level. Kendall's coefficient of concordance (W), which assesses the interjudge agreement of overall ratings, indicated that the three judges graded similar score with a high value ($W = 0.756$).

Table I shows the resemblance of orderings to those made by humans. Although we found that RO is clearly the worst, as in other results, we found no

Table I. Comparison with Human-Made Orderings

	Spearman		Kendall		Continuity	
Method	AVG	SD	AVG	SD	AVG	SD
RO	−0.117	0.265	−0.073	0.202	0.054	0.064
CO	0.838	0.185	0.778	0.198	0.578	0.218
COT	0.847	0.164	0.782	0.186	0.571	0.229
PO	0.843	0.180	0.792	0.184	0.606	0.225
POT	0.851	0.158	0.797	0.171	0.599	0.237
HO	1.000	0.000	1.000	0.000	1.000	0.000

Table II. Comparison with Corrected Orderings

	Spearman		Kendall		Continuity	
Method	AVG	SD	AVG	SD	AVG	SD
RO	0.041	0.170	0.035	0.152	0.018	0.091
CO	0.838	0.185	0.870	0.270	0.775	0.210
COT	0.847	0.164	0.791	0.440	0.741	0.252
PO	0.843	0.180	0.921	0.144	0.856	0.180
POT	0.851	0.158	0.842	0.387	0.820	0.240
HO	0.949	0.157	0.947	0.138	0.922	0.138

significant differences among CO, PO, and HO. This result revealed the difficulty of automatic evaluation by preparing a correct ordering.

Table II reports the resemblance of orderings to the corrected ones with average scores (AVG) and standard deviations (SD) of the three metrics τ_s, τ_k, and τ_c. Apparently, average figures have a similar tendency to the rating task with three measures: HO is the best; PO is better than CO; and RO is definitely the worst. We applied one-way analysis of variance (ANOVA) to test the effect of these four different methods (RO, CO, PO, and HO). ANOVA verified the effects of the different methods ($p < 0.01$) for the three metrics. We also applied the Tukey test to compare the differences among these methods. The Tukey test revealed that RO was definitely the worst with all metrics. However, Spearman's rank correlation τ_S and Kendall's rank correlation τ_k failed to show significant differences among CO, PO, and HO. Only sentence continuity τ_c demonstrated that PO is superior to CO; and that HO is better than CO ($\alpha = 0.05$). The Tukey test suggested that sentence continuity has better conformity to the rating results and higher discrimination to make a comparison.

As just described, the proposed method shows a significant improvement. However, evaluation by rating (Figure 8) and comparison with corrected ordering (Table II) also present a great difference between PO and HO. The main reason they made such a difference is the way of arranging lead sentences. The proposed method is intended to preserve chronological order of lead sentences as long as the refinement algorithm does not choose them as a substitution of preceding information for an arranging sentence. Although a human can devise a presentation order of lead sentences based on common sense, it is difficult for computers to grasp preceding information of each lead sentence.

In addition, several cases were found in which the proposed method inserted an unnecessary or inappropriate sentence as presuppositional information of a sentence. Because we do not apply a deep analysis of discourse structure and instead use precedent relation, a sentence does not always require all or any

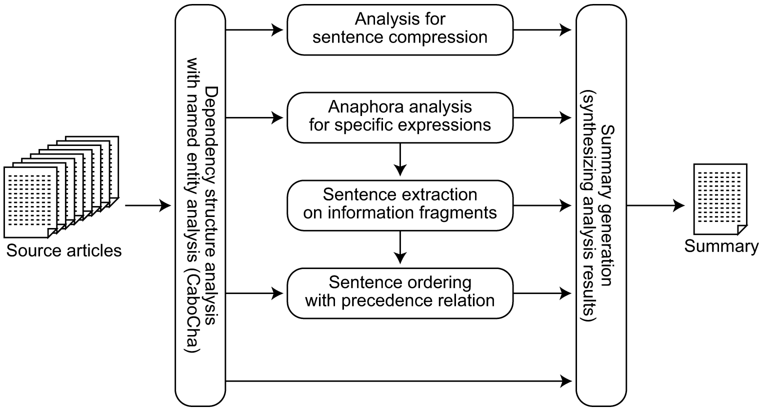

Source articles
Dependency structure analysis with named entity analysis (CaboCha)
Analysis for sentence compression
Anaphora analysis for specific expressions
Sentence extraction on information fragments
Sentence ordering with precedence relation
Summary generation (synthesizing analysis results)
Summary

Fig. 9. Architecture of our MDS system.

preceding sentences as presuppositional information. If the proposed method employs unnecessary preceding sentences as presuppositional information, it may choose a sentence that has little relation to the arranging sentence. The proposed method roughly estimates presuppositional information in this manner, but shows practical improvement for most summaries.

5. OUTLINE OF OUR MULTI-DOCUMENT SUMMARIZATION SYSTEM

Sentence ordering is a component of our MDS system for TSC-3. This section summarizes our MDS system and its evaluation in TSC-3. For more detailed description of this system, refer to the TSC-3 conference paper [Okazaki et al. 2004].

Figure 9 shows the architecture of our summarization system. In the first step, all documents are passed to CaboCha [Kudo and Matsumoto 2002][5] to acquire dependency structures of sentences and extract named entities. We perform two kinds of tasks on the summarization source: *important sentence extraction* and *analyses for generating a summary of good readability*.

Important sentence extraction for MDS [e.g., Carbonell and Goldstein 1998; Radev et al. 2000] should identify information in source documents to determine which information is important for inclusion and which information is unimportant or redundant in a summary. Assuming that a human reader breaks a sentence into several informational phrases to which the sentence is referring, we express each sentence in attributes and their weights [e.g., Salton et al. 1975; Nagao and Hasida 1998; Mani and Bloedorn 1999; Wada et al. 2002; Okazaki et al. 2002]. We convert each source sentence into a set of *information fragments* that consist of dependency relations of two terms and their weights calculated by statistical analysis. Figure 10 demonstrates the procedure for converting a sentence into information-fragment representation. We then determine a set of sentences containing as many important information fragments, formulating the important sentence extraction as a combinational optimization problem

[5] CaboCha is a Japanese dependency structure analyzer including a built-in named-entity analyzer based on Support Vector Machines (SVM).

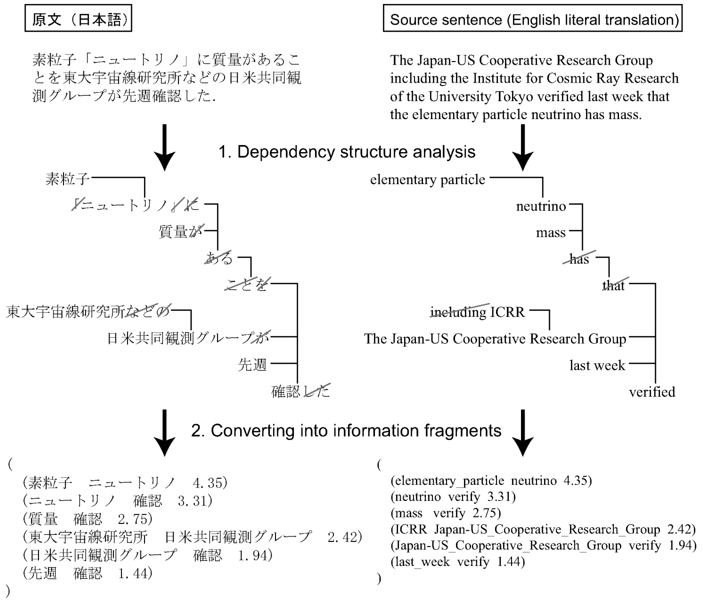

原文（日本語）
素粒子「ニュートリノ」に質量があることを東大宇宙線研究所などの日米共同観測グループが先週確認した.
Source sentence (English literal translation)
The Japan-US Cooperative Research Group including the Institute for Cosmic Ray Research of the University Tokyo verified last week that the elementary particle neutrino has mass.
1. Dependency structure analysis
素粒子
~~「~~ニュートリノ~~」に~~
質量~~が~~
~~ある~~
~~ことを~~
東大宇宙線研究所~~などの~~
日米共同観測グループ~~が~~
先週
確認~~した~~
elementary particle
neutrino
mass
~~has~~
~~that~~
~~including~~ ICRR
The Japan-US Cooperative Research Group
last week
verified
2. Converting into information fragments
(
(素粒子　ニュートリノ　4. 35)
(ニュートリノ　確認　3. 31)
(質量　確認　2. 75)
(東大宇宙線研究所　日米共同観測グループ　2. 42)
(日米共同観測グループ　確認　1. 94)
(先週　確認　1. 44)
)
(
(elementary_particle neutrino 4.35)
(neutrino verify 3.31)
(mass verify 2.75)
(ICRR Japan-US_Cooperative_Research_Group 2.42)
(Japan-US_Cooperative_Research_Group verify 1.94)
(last_week verify 1.44)
)

Fig. 10. Generation of information fragments from a sentence.

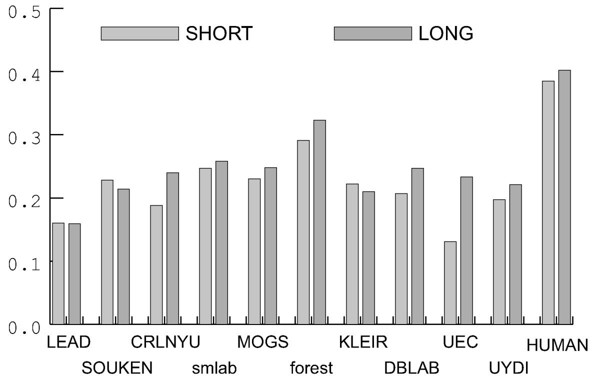


Fig. 11. Results for content evaluation.

under the constraint of the summarization ratio. It is also important to improve summary readability, because MDS gathers information over different documents. From among such components, *redundant clause elimination* deletes redundant or repeated expressions within sentences [Ishizako et al. 2000]. Such processing cannot be achieved by important sentence extraction. We focus attention of repeated expressions peculiar to newspaper articles such as *"– no-jiken-de (on the event of –)"* and *"– mondai-ni-tsuite (as for the problem that –)."* The component extracts clauses, which modify a noun phrase and measures similarity of all pairs of the clauses by dynamic programing (DP) matching. It then deletes clauses which are similar to previously arranged clauses.

Figure 11 shows the evaluation result of content coverage by human subjects. Results demonstrate the quality of important sentence extraction. When

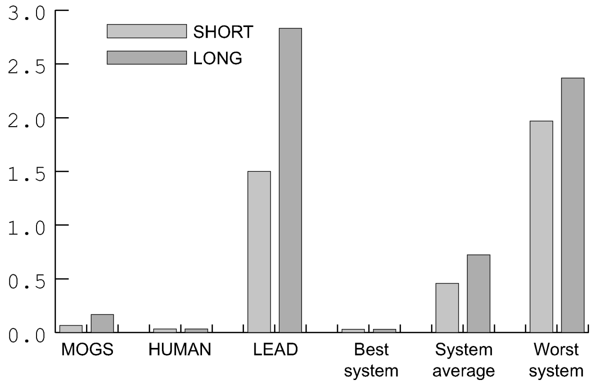


Fig. 12. Number of redundant or unnecessary sentences per summary.

a summary includes all necessary information, the evaluation value will be 1. Although we did not use any question information for summarization, our system (denoted by MOGS) performed well (3rd place; far better than a baseline[6] method; and above average) for both Short and Long summaries. Figure 12 shows the number of redundant or unnecessary sentences in one summary. It indicates the quality of both important sentence extraction and redundant expression elimination. The more a summary includes redundant information, the higher the evaluation value. Our system (MOGS) hardly includes redundant sentences (0.067 redundant sentences for a Short summary and 0.167 sentences for a Long summary on average).

6. CONCLUSION

We described our Multi-Document Summarization (MDS) system for TSC-3, specifically outlining an approach to coherent sentence ordering for the MDS system. We addressed a drawback of chronological ordering, which is widely used by conventional summarization systems: it arranges sentences without considering presupposed information of each sentence. Proposing a method to improve chronological ordering by resolving precedent information of arranging sentences, we conducted an experiment of sentence ordering through MDS. We also proposed an evaluation metric that measures sentence continuity and an amendment-based evaluation task. The proposed method, which utilizes the precedence relations of sentences, achieved good results, raising poor chronological orderings to an acceptable level by 20%. Amendment-based evaluation outperformed an evaluation that compares an ordering with an answer made by a human. The sentence continuity metric, when applied to the amendment-based task, showed good agreement with the rating result.

Future avenues of this study will point toward further improvement of sentence ordering. Assigning the highest priority to preserve chronological ordering, we can remedy situations in which chronological ordering might fail, based on presuppositional information of respective sentence arrangements.

[6] The baseline system is denoted by LEAD.

Although it was practical that the proposed method estimates presuppositional information by preceding sentences, there is room for improving that estimation. Arranging lead sentences in original documents also requires further investigation.

Another direction of this ongoing study will be toward establishment of an evaluation methodology for sentence ordering. This study uncovered the difficulty of automated evaluation of sentence ordering. We adopted amendment-based evaluation for sentence ordering and showed its accuracy and usefulness. Nevertheless, it requires a great deal of time and effort that would be difficult to repeat in a regular basis. We recognize the necessity of automatic evaluation that will probably feature multiple correct orderings for a summary with extended metrics of Kendall's rank correlation or sentence continuity.

Taking a global view of MDS, it may be an interesting approach that incorporates important sentence extraction and sentence ordering. Knowing what kinds of events tend to occur after an event, the extraction scheme can promote peripheral information fragments after inclusion of the fragments. This knowledge of a natural course of events benefits sentence ordering. They are not isolated problems: we plan to pursue their integration for their mutual benefits and improvement of overall quality of MDS.

ACKNOWLEDGMENTS

We used Mainichi Shinbun and Yomiuri Shinbun newspaper articles and the TSC-3 test collection for evaluation. We also wish to thank TSC organizers for organizing their valuable workshops and community. We thank the reviewers for their very useful comments.

REFERENCES

Barzilay, R. and Lee, L. 2004. Catching the drift: Probabilistic content models, with applications to generation and summarization. In *HLT-NAACL 2004: Proceedings of the Main Conference*. 113–120.

Barzilay, R., Elhadad, E., and McKeown, K. 2002. Inferring strategies for sentence ordering in multidocument summarization. *Journal of Artificial Intelligence Research (JAIR) 17*, 35–55.

Carbonell, J. and Goldstein, J. 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*. 335–336.

Cover, T. M. and Hart, P. E. 1967. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory IT-13*, 21–27.

Hobbs, J. 1990. *Literature and Cognition, CSLI Lecture Notes 21*. CSLI.

Hume, D. 1748. *Philosophical Essays concerning Human Understanding*. Printed for A. Millar London.

Ishizako, Y., Kataoka, A., Masuyama, S., Yamamoto, K., and Nakagawa, S. 2000. Reduction of overlapping expressions using dependency relations. *Journal of Natural Language Processing 7*, 4, 119–142. (in Japan.)

Kudo, T. and Matsumoto, Y. 2002. Japanese dependency analysis using cascaded chunking. In *CoNLL 2002: Proceedings of the 6th Conference on Natural Language Learning 2002 (COLING 2002 Post-Conference Workshops)*. 63–69.

Lapata, M. 2003. Probabilistic text structuring: experiments with sentence ordering. In *Proceedings of the 41st Meeting of the Association of Computational Linguistics*. 545–552.

Lin, C.-Y. and Hovy, E. 2001. NEATS: A multidocument summarizer. In *Proceedings of the Document Understanding Conference (DUC01)*.

LIN, C.-Y. AND HOVY, E. 2002. From single to multi-document summarization: A prototype system and its evaluation. In *Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL-02)*. Philadelphia, PA., USA, 457–464.

LUHN, H. P. 1958. The automatic creation of literature abstracts. *IBM journal of Research and Development 2*, 2, 159–165.

MANI, I. 2001. *Automatic Summarization*. John Benjamins, Amsterdam.

MANI, I. AND BLOEDORN, E. 1999. Summarizing similarities and differences among related documents. *Information Retrieval 1*, 1-2, 35–67.

MANI, I. AND WILSON, G. 2000. Robust temporal processing of news. In *Proceedings of the 38th Annual Meeting of ACL'2000*. 69–76.

MANI, I., SCHIFFMAN, B., AND ZHANG, J. 2003. Inferring temporal ordering of events in news. *Proceedings of the Human Language Technology Conference (HLT-NAACL) '03*.

MANN, W. AND THOMPSON, S. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text 8*, 243–281.

MCKEOWN, K., KLAVANS, J., HATZIVASSILOGLOU, V., BARZILAY, R., AND ESKIN, E. 1999. Towards multidocument summarization by reformulation: Progress and prospects. In *Proceedings of the 17th National Conference on Artificial Intelligence*. 453–460.

NAGAO, K. AND HASIDA, K. 1998. Automatic text summarization based on the global document annotation. In *Proceedings of the 17th International Conference on Computational Linguistics / 36th Annual Meeting of the Association for Computational Linguistics (COLING-ACL '98)*. Montreal, Quebec, Canada. 917–921.

OKAZAKI, N., MATSUO, Y., MATSUMURA, N., TOMOBE, H., AND ISHIZUKA, M. 2002. Two different methods at NTCIR3-TSC2: Coverage oriented and focus oriented. In *Working Notes of the Third NTCIR Workshop Meeting, Part V: Text Summarization Challenge 2 (TSC2)*. 39–46.

OKAZAKI, N., MATSUO, Y., AND ISHIZUKA, M. 2004. TISS: An integrated summarization system for TSC-3. In *Working note of the 4th NTCIR Workshop Meeting*. 436–443.

RADEV, D. R. AND MCKEOWN, K. 1998. Generating natural language summaries from multiple on-line sources. *Computational Linguistics 24*, 3, 469–500.

RADEV, D. R., JING, H., AND BUDZIKOWSKA, M. 2000. Centroid-based summarization of multiple documents: Sentence extraction, utility-based evaluation, and user studies. In *The ANLP / NAACL2000 Workshop on Automatic Summarization*. 21–30.

SALTON, G., WONG, A., AND YANG, C. S. 1975. A vector space model for automatic indexing. *Communications of the ACM 18*, 11, 613–620.

WADA, Y., OKUMURA, A., URATANI, N., AND SHIRAI, K. 2002. News sentence summarization based on importance of *bunsetsu* attributes. In *Proceedings of the 8th Annual Meeting of the Association for Natural Language Processing.* 543–546. (*in Japan*).

Received June 2004; revised November 2004; accepted April 2005