NTT's Multiple Document Summarization System for DUC2003

Tsutomu HIRAO, Jun SUZUKI, Hideki ISOZAKI and **Eisaku MAEDA**
NTT Communication Science Laboratories, NTT Corp.
2-4,Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0237 Japan
{hirao,jun,isozaki,maeda}@cslab.kecl.ntt.co.jp

Abstract

We participated in the Document Understanding Conference 2003 (DUC-2003) in order to confirm the effectiveness of our multiple document summarization system using Support Vector Machines (SVMs) and kernel-based Maximum Marginal Relevance (MMR). The results of the task-2 and task3 show that our system performed very well.

1 Introduction

In this paper, we provide a description of our system, summarizing multiple documents as an extension of a single-document summarization. Our system employs Support Vector Machines (SVMs) (Vapnik, 1995) for sentence extraction from each document contained in a given document set, and uses Maximum Marginal Relevance (Carbonell and Goldstein, 1998; Goldstein et al., 2000) to minimize redundancy for extracted sentences. In addition, we propose a novel similarity measure based on a kernel for MMR.

The results of the multiple document summarization tasks (task-2 and task-3) showed that our system achieved high grades in both coverage and readability metrics.

The remainder of this paper is organized follows: Section 2 describes the sentence extraction method based on SVMs and MMR with Extended String Subsequence Kernel (ESSK), and in Section 3, we give our evaluation results at DUC-2003. Section 4 concludes this paper.

2 Kernel-based approach to Multiple Document Summarization

In this section, we describe a multiple document summarization based on important sentence extraction. We employ the decision function of SVMs as a significance score for a sentence and MMR for minimizing redundancy (Figure 1).

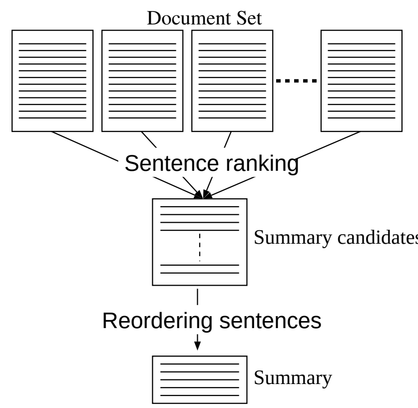

Figure 1: Summarization Process.

Next, we describe our method based on Support Vector Machines (SVMs). Following that, we describe a method of minimizing redundancy by MMR with the proposed similarity measure.

2.1 Sentence Ranking by SVMs

The extraction of important sentences from a document can be regarded as a two-class problem. Therefore, a single-document summarization method using SVMs has shown good performance for both Japanese (Hirao et al., 2002a) and English documents (Hirao et al., 2002b). This approach uses $g(\mathbf{x})$ sentence ranking rather than classifying the sentence as important or not. This approach is similar to interpreting $g(\mathbf{x})$ as a likelihood measure (Isozaki and Kazawa, 2002; Kudo and Matsumoto, 2000). Theoretical justification of the use of $g(\mathbf{x})$ as a likelihood measure is shown in (Kazawa et al., 2002; Kwok, 1999).

In this paper, we also use $g(\mathbf{x})$ for sentence ranking.

2.2 Features for Summarization

We use the following features of sentence S_i by taking past studies into account (Kupiec et al., 1995; Sekine and Nobata, 2001).

Position of Sentences

Posd is S_i's position in a document. The first sentence obtains the highest score, the last obtains the lowest score:

$$\mathrm{Posd}(S_i) = 1 - \frac{BD(S_i)}{|D|}.$$

Here, $|D|$ is the number of words in the document D that contains S_i and $BD(S_i)$ is the number of words before S_i in D.

Length of Sentences

We define a feature function that addresses the length of sentences as

$$\mathrm{Len}(S_i) = |S_i|.$$

Here, $|S_i|$ is the number of words in S_i.

Weight of Sentences

We define the feature function that weights sentences based on the weight of a word as

$$\begin{aligned} \mathrm{Score}(S_i) &= \sum_t tf(t, S_i) \cdot w(t, D), \\ \mathrm{Score}_{\mathrm{HL}}(S_i) &= \sum_{t' \in \mathrm{HL}} tf(t', S_i) \cdot w(t', D), \end{aligned}$$

where, $\mathrm{Score}(S_i)$ is the summation of weighting $w(t, D)$ of words that appear in sentence S_i. $\mathrm{Score}_{\mathrm{HL}}(S_i)$ is the summation of weighting $w(t', D)$ of words that appear in headline (HL). In addition, we define the weight of a word based on frequency:

$$w(t, D) = \log(1 + tf(t, D)).$$

Here, $tf(t, D)$ is the frequency of t in D.

Named Entity

The boolean value '1' is given to S_i, which includes a certain Named Entity class. As listed below, there are nine Named Entity classes as follows:

PERSON, LOCATION, ORGANIZATION, GPE[1], FACILITY, DATE, MONEY, PERCENT, TIME.

[1]Geo-Political Entity.

$A = \{\}$;
$R = \{S_1, S_2, \cdots, S_\ell\}$;
N =The number of sentences as output;
While($|A| < N$){

$S^* = \mathrm{MMR}(A, R)$;
$A = A \cup \{S^*\}$;
$R = R - \{S^*\}$;

}
Output A, where

$$\mathrm{MMR}(A, R) = \begin{cases} \underset{S_i \in R}{\operatorname{argmax}}\, s(g(S_i)) & \text{if} \quad A = \phi \\ \underset{S_i \in R}{\operatorname{argmax}} \big(\alpha s(g(S_i)) \; - \\ \quad (1-\alpha) \max_{S_j \in A} \mathrm{Sim}(S_i, S_j)\big) \end{cases}$$

Figure 2: Reordering algorithm by MMR

Prepositions

The boolean value '1' is given to S_i, which includes a certain preposition.

Verbs

The boolean value '1' is given to S_i if it includes a certain verb.

2.3 Redundancy Minimization by MMR with Extended String Subsequence Kernel

It is said that a document set includes redundant sentences. In order to minimize redundancy, Carbonell proposed Maximum Marginal Relevance (MMR) (Carbonell and Goldstein, 1998; Goldstein et al., 2000). The MMR deals with two factors: a significance score of a sentence and a similarity between the sentence and sentences already selected for summary.

Figure 2 shows a reordering algorithm based on MMR. In the figure, R is the set of all sentences in a given document set. A is the set of sentences selected for summary and $s(x)$ is a sigmoid function defined as $1/(1+\exp(-\beta x))$. Here, we set β as 1. We use the sigmoid function to normalize the output of the decision function $g(\mathbf{x})$, while $\mathrm{Sim}(S_i, S_j)$ provides similarity between sentence S_i and sentence S_j, and α is a trade-off parameter for the two arguments.

Here, we propose a similarity measure for $\mathrm{Sim}(S_i, S_j)$ with Fig. 3 showing the key idea. We measure the similarity between sentences based on a node which includes a word and its senses. In the figure, s_i $(i = 1, 2, 3)$ and t_j $(j = 1, 2, 3)$ indicate a node. In Fig. 3, w indicates a word and x indicates

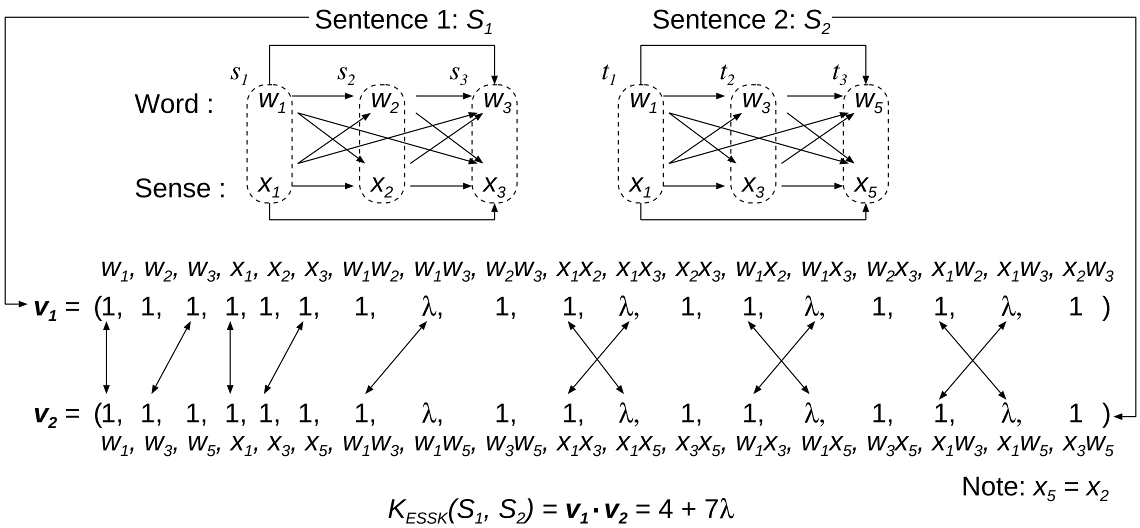

Figure 3: Extended String Subsequence Kernel.

w's sense. Here, we use WordNet (Fellbaum, 1998) to obtain word senses.

The similarity measure is a simple extension of the String Subsequence Kernel (SSK) (Lodhi et al., 2002), and we call this similarity the *Extended String Subsequence Kernel* (ESSK). SSK measures the similarity between two sequences of "alphabets." In ESSK, each "alphabet" in SSK is replaced by a disjunction of an "alphabet" and its alternatives, and once a sequence pair is given, ESSK measures their similarity by adding the SKK-like similarity measures of all the possible "alphabet" sequences. Here, each word in a sentence is considered an "alphabet," and its alternatives are its senses given by WordNet. We give decay parameter λ for a sequence pair skipping alphabets according to the number of skips. When a sequence that consists of "abcde" is given, we assign subsequences 'ab' $\lambda^0(=1)$, 'ac' λ^1, 'ad' λ^2 and 'ae' λ^3.

Formally, we define the kernel as follows.

$$K_{ESSK}(S_x, S_y) = \sum_{m=1}^{n} \sum_{s_i \in S_x} \sum_{t_j \in S_y} K_m(s_i, t_j). \quad (1)$$

$$K_m(s_i, t_j) = \begin{cases} val(s_i, t_j) & \text{if} \quad m = 1 \\ K'_{m-1}(s_i, t_j) \cdot val(s_i, t_j). & \end{cases} \quad (2)$$

Here, $K'_m(s_i, t_j)$ is defined as follows.

$$K'_m(s_i, t_j) = \begin{cases} 0 & \text{if} \quad j = 1 \\ \lambda K'_m(s_i, t_{j-1}) + K''_m(s_i, t_{j-1}) & \end{cases} \quad (3)$$

$K''_m(s_i, t_j)$ is defined as follow:

$$K''_m(s_i, t_j) = \begin{cases} 0 & \text{if} \quad i = 1 \\ \lambda K''_m(s_{i-1}, t_j) + K_m(s_{i-1}, t_j) & \end{cases} \quad (4)$$

The function $val(s, t)$ returns the number of matching between nodes s and t[2]. In Figure 3, $K_{ESSK}(S_1, S_2)$ equals $\mathbf{v}_1 \cdot \mathbf{v}_2$.

Finally, the similarity between S_i and S_j is defined as follows:

$$\text{Sim}(S_i, S_j) = \frac{K_{ESSK}(S_i, S_j)}{\sqrt{K_{ESSK}(S_i, S_i) K_{ESSK}(S_j, S_j)}}.$$

An efficient algorithm is shown Figure 4. In addition, the characteristic point of this kernel is that the inputs are raw sentences.

2.4 Summarization Procedure

Summarization procedure is as follows.

Step 0: Train SVMs using single document summarization data[3].

Step 1: Give a significance score for each sentence in a given document set by SVMs.

Step 2: Select top N[4] sentences according to $g(\mathbf{x})$ from the document set as summary candidates.

[2] In the Figure 3, $val(s_1, t_1) = 2$, $val(s_2, t_3) = 1$ and $val(s_1, t_3) = 0$.

[3] We use data set at DUC-2001

[4] We set N 20.

for $(i = 1; i \leq |D_1|; i{+}{+})$
  for $(j = 1; j \leq |D_2|; j{+}{+})$

    if$(i = 1)$
      $K''_m(s_i, t_j) = 0$
    else

    for $(m = 1; m \leq n; m{+}{+})$
      $K''_m[s_i, t_j] \mathrel{+}= \lambda K''_m[s_{i-1}, t_j] + K_m[s_{i-1}, t_j]$
    end

    if$(j = 1)$
      $K''_m(s_i, t_j) = 0$
    else

    for $(m = 1; m \leq n; m{+}{+})$
      $K'_m[s_i, t_j] \mathrel{+}= \lambda K'_m[s_i, t_{j-1}] + K''_m[s_i, t_{j-1}]$
    end

    $K_1[s_i, t_j] = val(s_i, t_j)$
    for $(m = 2; m \leq n; m{+}{+})$
      $K_m[s_i, t_j] = K'_{m-1}[s_i, t_j] \cdot val(s_i, t_j)$
    end
  end
end
return $\sum_{m=1}^{n} \sum_{s_i \in s} \sum_{t_j \in t} K_m[s_i, t_j]$

Figure 4: Algorithm for ESSK

Step 3: Reorder candidate sentences by MMR.

Step 4: Select sentences for the summary from the candidate until the specified summarization rate (word limit) is reached.

3 Results

Tables 1 and 2 show the results of subjective evaluation in task-2 and task-3, respectively. In the tables, MC and MLAC are coverage metrics, while CQQ and MQQ are readability metrics; the higher score represents better performance. MC denotes "Mean coverage," MLAC denotes "Mean length-adjusted coverage," CQQ denotes "Count of quality questions," and MQQ denotes "Mean of the quality question scores." In these cases, a lower score means better performance. Our system's ID is "11."

Our system achieved 1st in MQQ at both task-2 and task-3 and 9th and 4th in LAC in task-2 and task-3, respectively. In coverage metrics, our system's performance in task-2 was lower than that in task-3. One reason for this result is that the style of the texts in task-2 was different from our training data. In readability metrics, our system's performance achieved the highest. We consider that the summary length strongly influences readability because the length of our system's summary is short.

4 Conclusion

We described our system based on Support Vector Machines and kernel-based MMR, which participated in the multiple document summarization tasks at DUC-2003, and provided evaluation results.

Acknowledgements

We would like to thank Dr. Sekine and Dr. Nobata for helping us to use their algorithm that aligns abstracts with extracts, and for allowing us to use the English sentence analyzer OAK System.

References

J. Carbonell and J Goldstein. 1998. The use of mmr, diversity-based reranking for reordering document and producing summaries. *Proc. of the 21th Annual International ACM SIGIR Conference on Research and Development in the Information Retrieval*, pages 335–336.

C. Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

J. Goldstein, V. Mittal, and J. Carbonell, J.and Callan. 2000. Creating and evaluating multi-document sentence extract summaries. *Proc. of the 9th International Conference on Information and Knowledge Management*, pages 165–172.

T. Hirao, H. Isozaki, E. Maeda, and Y. Matsumoto. 2002a. Extracting important sentences with support vector machines. *Proc. of the 19th International Conference on Computational Linguistic*, pages 342–348.

T. Hirao, Y. Sasaki, H. Isozaki, and E. Maeda. 2002b. NTT's text summarization system for duc-2002. *Proc. of the 2nd Document Understanding Conference*, pages 104–107.

H. Isozaki and H. Kazawa. 2002. Efficient support vector classifiers for named entity recognition. *Proc. of the 19th International Conference on Computational Linguistic*, pages 390–396.

H. Kazawa, T. Hirao, and E. Maeda. 2002. Ranking svm and its application to sentence selection (in Japanese). *Proc. 2002 Workshop on Information-Based Induction Science (IBIS2002)*, pages 37–42.

T. Kudo and Y. Matsumoto. 2000. Japanese dependency structure analysis based on support vector machines. *Proc. of Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 18–25.

Table 1: Evaluation Results in task-2

SYSTEM-ID	MC	MLAC	CQQ	MQQ
2	0.157	0.113	1.333	0.767
3	0.191	0.133	2.900	1.033
6	0.277	0.191	1.400	0.817
10	0.186	0.198	1.833	0.900
11	0.198	0.187	0.967	0.533
12	0.243	0.163	1.600	0.800
13	0.290	0.195	3.200	1.028
14	0.247	0.191	2.533	0.919
15	0.077	0.070	7.867	1.335
16	0.276	0.196	1.267	0.833
17	0.171	0.117	1.800	0.917
18	0.222	0.179	2.567	0.933
19	0.141	0.142	2.500	0.967
20	0.262	0.187	2.333	1.083
21	0.187	0.146	2.667	0.978
22	0.217	0.259	1.200	0.633
23	0.250	0.197	2.333	0.960
26	0.222	0.152	2.000	0.950

Table 2: Evaluation Results in task-3

SYSTEM-ID	MC	MLAC	CQQ	MQQ
2	0.130	0.090	1.833	0.911
3	0.172	0.119	2.633	1.185
10	0.166	0.171	1.367	0.744
11	0.152	0.164	1.000	0.700
13	0.129	0.088	2.433	0.983
15	0.102	0.093	6.300	1.269
16	0.193	0.136	2.033	0.948
17	0.190	0.127	1.833	0.733
18	0.170	0.165	2.033	0.883
20	0.199	0.138	2.100	1.036
21	0.140	0.116	2.667	0.946
22	0.166	0.173	1.467	0.767
23	0.157	0.141	2.233	1.031

J. Kupiec, J. Pedersen, and F. Chen. 1995. A trainable document summarizer. *Proc. of the 18th Annual International ACM SIGIR Conference on Research and Development in the Information Retrieval*, pages 68–73.

J.T. Kwok. 1999. Moderating the outputs of support vector machine classifiers. *IEEE Transactions on Neural Networks*, 10(5):1018–1031.

H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins. 2002. Text classification using string kernel. *Journal of Machine Learning Research*, 2(Feb):419–444.

S. Sekine and C. Nobata. 2001. Sentence extraction with information extraction technique. *Proc. of the 1st Document Understanding Conference*.

V. Vapnik. 1995. *The Nature of Statistical Learning Theory*. Springer, New York.