5J-3

A Machine Learning Approach to Sentence Ordering for Multi Document Summarization

Danushka Bollegala* **Naoaki Okazaki†** **Mitsuru Ishizuka†**

Abstract

Ordering information is a difficult but an important task for natural language generation applications. A wrong order of information not only makes it difficult to understand but also conveys enitrely different idea to the reader. In this paper we propose an algorithm that will learn orderings from a set of human ordered texts. Our model consists of a set of ordering experts. Each expert gives its precedence preference between two sentences. We combine these preferences and order sentences. We also propose two new metrics for the evaluation of sentence orderings. Our experimental results show that the proposed algorithm outperforms the existing methods in all eveluation metrices.

1 Introduction

Multidocument summarization(MDS) is the task of generating a human readable summary from a given set of documents. It can be considered as a two-stage process. On the first stage we must extract a set of sentences from the given document set. Researchers have already investigated this stage of MDS and designed efficient algorithms for this task. The second stage of MDS is creating a comprehensible summary from this extract. A good ordering of sentences improves coherence of a summary. Unlike in single document summarization, extracted sentences belong to different documents. Barzilay (2002) proposes a chronology oriented approach and Lapata (2003) gives a probabilistic text structuring approach to sentence ordering. However, to order a set of sentences correctly, we must consider many other features besides chronology and probabilisitc co-occurances. An algorithm which is able to learn such rules of ordering is needed. Therefore we used a combination of ordering methods and designed an algorithm which can be trained to order sentences.

2 Method

For sentences taken from the same document we keep the order in that document as done in single document summarization. However, we have to be careful when ordering sentences which belong to different documents. To decide the order among such sentences, we implemented five ranking experts. These experts return precedence preference between two sentences. Cohen (1999) proposes an elegant learning model that works with preference functions and we adopt this learning model to our task. Each expert, e generates a pair-wise preference function defined as follows,

$$\mathrm{PREF}_e(u, v, Q) \in [0, 1]. \quad (1)$$

Here,u, v are two sentences that we want to order; Q is the set of sentences which has been already ordered. The expert returns its preference of u to v. The linear weighted sum of these individual preference functions is taken as the total preference by the set of experts.

$$\mathrm{PREF}_{total}(u, v, Q) = \sum_{e \in E} w_e \mathrm{PREF}_e(u, v, Q) \quad (2)$$

Here, E is the set of experts and w_e is the weight associated to expert $e \in E$. These weights are normalized so that the sum of them is 1. We use the Hedge learning algorithm to learn the weights associated with each expert's preference function. Then we use the greedy algorithm proposed by Cohen (1999) to get an ordering according to the total preference.

2.1 Chronological expert

Chronological expert orders sentences according to the publication date and sentence position.

$$\mathrm{PREF}_{chro}(u, v, Q) = \begin{cases} 1 & T(u) < T(v) \\ 1 & [D(u) = D(v)] \wedge [N(u) < N(v)] \\ 0.5 & [T(u) = T(v)] \wedge [D(u) \neq D(v)] \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

Therein: $T(u)$ is the publication date of sentence u, $D(u)$ presents the unique identifier of the document to which sentence u belongs; $N(u)$ denotes the line number of sentence u in $D(u)$.

2.2 Probabilistic expert

Lapata (2003) calculates the conditional probability $P(S_i|S_j)$, that the sentence S_i appearing after the sentence S_j in the summary, using some selected features. In our calculations we limited these features to verbs and nouns.

$$\mathrm{PREF}_{prob}(u, v) = \frac{1 + P(v|u) - P(u|v)}{2} \quad (4)$$

2.3 Topical relevance expert

This expert prefers sentences which are more similar to the ones that have been already ordered. For each

*Department of Information Engineering, The University of Tokyo

†Graduate School of Information Sciences, The University of Tokyo

sentence l in the extract we define its topical relevance, $\mathrm{topic}(l)$ as,

$$\mathrm{topic}(l) = \max_{q \in Q} \mathrm{sim}(l, q). \quad (5)$$

We use cosine similarity to calculate $\mathrm{sim}(l, q)$.

$$\mathrm{PREF}_{topic}(u, v, Q) = \begin{cases} 0.5 & [Q = \Phi] \vee [\mathrm{topic}(u) = \mathrm{topic}(v)] \\ 1 & [Q \neq \Phi] \wedge [\mathrm{topic}(u) > \mathrm{topic}(v)] \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

2.4 Precedent expert

Okazaki (2005) proposes precedence relations as a method to improve the chronological ordering of sentences. He considers the information stated in the documents preceding extract sentences to judge the order. Based on this idea, we define the the precedence $\mathrm{pre}(l)$ of extract sentence l as follows,

$$\mathrm{pre}(l) = \max_{p \in P, q \in Q} \mathrm{sim}(p, q) \quad (7)$$

Here, P is the set of sentences preceding the extract sentence l in the original document. By substituting pre for topic, we can define the preference function for precedent expert as we did in (6).

2.5 Succedent expert

When extracting sentences from documents, sentences which are similar to ones already extracted are usually ignored. However, this information is valuable when ordering sentences. We design an expert which uses this information to order sentences. When r is the lastly ordered sentence in the summary so far, we find the block K of sentences that appear after r in the original document. For each of the unordered extract sentence l, we define its succedence $\mathrm{succ}(l)$ as follows,

$$\mathrm{succ}(l) = \max_{k \in K} \mathrm{sim}(l, k) \quad (8)$$

The preference function for the succedent expert is defined by substituting succ for topic in (6). Weights are learnt using Cohen (1999) hedge algorithm. An ordering that approximates this total preference is generated using the Greedy Algorithm proposed by Cohen (1999).

3 Result

Preparing 30 sets of extracted sentences based on the TSC-3 extract data, we ordered each extract by five methods: Random Ordering (RO); Probabilistic Ordering (PO); Chronological Ordering (CO); Learned Ordering (LO); and HO (Human-made Ordering). We measure closeness of respective orderings to the human-made one and evaluate each method. In addition to Spearman's τ_s and Kendall's τ_k rank correlations which are widely used to compare two ranks, we use sentence continuity (Okazaki et al., 2005) and its extension, Average Continuity (AC). Figure 1 shows precisions of n-sentence continuity (i.e., sentence n-gram precisions of an ordering against HO). We define,

Table 1: Experimental Results

	τ_s	τ_k	τ_c	AC
RO	-0.267	-0.160	-0.118	0.024
PO	0.058	-0.019	-0.093	0.019
CO	0.774	0.735	0.629	0.511
LO	0.783	0.746	0.706	0.546
HO	1.000	1.000	1.000	1.000

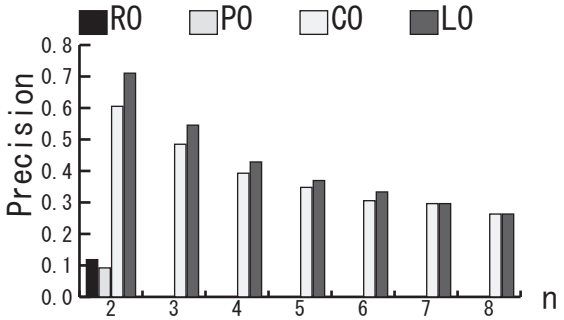

Figure 1: sentence n-gram precision

$$\mathrm{AC} = \exp \sum_{n=2}^{4} \log \frac{\text{number of matched n-grams}}{N - n + 1} \quad (9)$$

As seen from table 3 the proposed algorithm(LO) performs better than the existing base line methods. In Figure 1, for all lengths of continuity, the proposed method has better precisions than the existing methods. From these results we can conclude that our trained algorithm outperforms all the existing methods which are used to order sentences.

References

Regina Barzilay, Noemie Elhadad, and Kathleen McKeown. 2002. Inferring strategies for sentence ordering in multidocument news summarization. *Journal of Artificial Intelligence Research*, 17:35–55.

W. W. Cohen, R. E. Schapire, and Y. Singer. 1999. Learning to order things. *Journal of Artificial Intelligence Research*, 10:243–270.

Mirella Lapata. 2003. Probabilistic text structuring: Experiments with sentence ordering. *Proceedings of the annual meeting of ACL, 2003.*, pages 545–552.

Naoaki Okazaki, Yutaka Matsuo, and Mitsuru Ishizuka. 2005. An integrated summarization system with sentence ordering using precedence relation. *ACM-TALIP, to appear in 2005.*