Class Lecture Summarization Taking into Account Consecutiveness of Important Sentences

Yasuhiwa Fujii, Kazumasa Yamamoto, Norihide Kitaoka, and Seiichi Nakagawa

1Department of Information and Computer Sciences, Toyohashi University of Technology
fujii@lcs.tut.ac.jp, kyama@lcs.tut.ac.jp, nakagawa@lcs.tut.ac.jp
2Department of Media Science Graduate School of Information Science, Nagoya University
kitaoka@nagoya-u.jp

Abstract
This paper presents a novel sentence extraction framework that takes into account the consecutiveness of important sentences using a Support Vector Machine (SVM). Generally, most extractive summarizers do not take context information into account, but do take into account the redundancy over the entire summarization. However, there must exist relationships among the extracted sentences. Actually, we can observe these relationships as consecutiveness among the sentences. We deal with this consecutiveness by using dynamic and difference features to decide if a sentence needs to be extracted or not. Since important sentences tend to be extracted consecutively, we just used the decision made for the previous sentence as the dynamic feature. We used the differences between the current and previous feature values for the difference feature, since adjacent sentences in a block of important ones should have similar feature values to each other, whereas there should be a larger difference in the feature values between an important sentence and an unimportant one. We also present a way to ensure that no redundant summarization occurs. Experimental results on a Corpus of Japanese classroom Lecture Contents (CILC) showed that the dynamic and difference features were complementary, and our proposed method outperformed traditional methods, which did not take context information into account.

Index Terms: automatic speech summarization, sentence extraction, context information, support vector machine

1. Introduction
In the past decade, the amount of on-line academic lecture materials has increased, and the techniques used to transcribe the audio tracks of those materials have been investigated [1, 2]. If we can index and summarize the audio tracks with the transcriptions, the materials would be much easier to reference. Thus, there has been more focus put on studies on automatic speech summarization than ever before [3, 4, 5, 6].

Zhu et al. [3] examined the role of disfluencies and the impact of word error rate (WER) on the individual features for summarization. Chen et al. [4] proposed a unified probabilistic generative framework that combines sentence generative probability with sentence prior probability for sentence ranking. Ricardo et al. [5] compared three methods for extractive summarization of Portuguese broadcast news: feature-based, Latent Semantic Analysis, and Maximal Marginal Relevance. In our previous work [6], we proposed a Cue Phrase (CP) extraction technique with Conditional Random Fields (CRF) for automatic summarization.

Generally, summarization is conducted based on a sentence extraction method such as [3, 4, 5, 6]. Although they assumed the independence of sentences, there must be relationships among the extracted sentences. In [7], Kolluru et al. pointed out that important sentences tend to be consecutively extracted and they attempted to deal with this tendency by extracting the sentences adjacent to a seed sentence. Although they just extracted the adjacent sentences around a seed one, the summarizer used in this method was worse than the ones that did not consider the context, but was better concerning the coherence. Maskey et al. [8] summarized speech using a Hidden Markov Model (HMM) to take into account the context information. The hidden states of the HMM represented whether a sentence should be included in the summary or not. Their summarizer was able to take into account the previous decisions. They used only prosodic and structural information, whereas the linguistic features are also quite informative for speech summarization, even if recognition errors exist in the speech recognition results [9].

We propose a novel sentence extraction method in this paper that aims to capture the consecutiveness of important sentences. To model the consecutiveness we use two features: dynamic and difference features. A dynamic feature is the previous decision of extraction and the difference feature is the difference between the current and previous feature values. Our summarizer uses an SVM to incorporate the different kinds of features [6]. We can find the best sequence of important sentences that comprise a summary by using a dynamic programming technique. We also present a way to reduce redundancy from the entire summarization. Experimental results showed that our proposed method outperforms conventional summarization methods, which take into account no context information.

2. Consecutiveness of important sentences
When human perform summarization by extracting sentences, successive sentences tend to be extracted as important. In this section, we describe our corpus and reference for automatic summarization, and then we show the consecutiveness of important sentences extracted by humans on our corpus.

2.1. Corpus
For our experiments, we used eight lectures performed by four professors in CILC [10]. The lectures in the graduate course are related to spoken language processing, multimodal interface, pattern recognition, and natural language processing. Table 1 shows the statistics details of the data. Each lecture was about 70 minutes long and contained about 1,000 sentences. All speech data were transcribed by humans and a SPHINX ASR system [11]. The recognition performances of the speech were 49.1% in word accuracy, and 55.8% in word correct.
2.2. Reference for automatic summarization

Six speech research experts who understood the extremely high-level content of the targeted graduate course conducted the important sentence extraction. All subjects were instructed to mark the important sentences in the speech transcriptions as “important” for 1/4 of the sentences of the entire speech (25% rate of summary) in each lecture. In other words, every sentence was classified as important or not. Here “sentence” is defined as a portion between pauses longer than about 200msec. We also prepared reference data called man3/6, which corresponds to the sentences extracted by three or more subjects out of six and reflects the consensus of the subjects. The target values of the summarization shown in Table 1 are the averages of the scores obtained from each subject and man3/6 (sentences extracted by three or more subjects out of all but the evaluated subject), which have higher scores than the inter-subjects [12].

2.3. Consecutiveness of important sentences

Table 2 shows the consecutiveness of the important sentences in the references. The averages number of the important sentences and the isolated ones in a lecture were 268.0 and 80.6, respectively, and the average number of sentences in a consecutively extracted part was 1.83. These results mean that 2/3 of the important sentences were extracted with neighboring ones, whereas 1/3 of the important sentences are isolated. Thus, if we can model such a shallow consecutiveness, there is a possibility that we can generate a more sophisticated summarization.

3. Baseline methods

3.1. Maximal Marginal Relevance

Maximal Marginal Relevance (MMR) is a method for combining document-relevance with information-novelty presented by Carbonell et al. [13]. MMR criterion strives to reduce redundancy while maintaining document relevance when selecting appropriate sentences for text summarization. Although there have been several implementations of MMR, we used the same framework as used in [14], in which the MMR score $S_{C}^{MMR}(i)$ for a given sentence $S_i$ in a document is given by

$$S_{C}^{MMR}(i) = \lambda (\text{Sim}(S_i, D)) - (1 - \lambda)(\text{Sim}(S_i, S_{rk})), \quad (1)$$

where $D$ is the average document vector, $S_{rk}$ is the average vector from the set of sentences already extracted, and $\lambda$ is a factor which trades off between relevance and redundancy. $\text{Sim}$ represents a cosine similarity between two vectors. In our definition $S_i$ is defined as follows:

$$S_i = tf_i = (tf_{i,1}, tf_{i,2}, \ldots, tf_{i,w}), \quad (2)$$

$$tf_{i,w} = f_w \cdot \log \left( \frac{tf_{i,w}}{f_w} \right), \quad (3)$$

where $tf_i$ is a vector of $tf_{i,w}$, which means it is a modified term frequency, and $f_w$ is a frequency of $w$ in a document and $f_w$ means the highest frequency in all words throughout the document. We use MMR as a baseline.

3.2. Feature based extraction

In our previous work [6], we used a feature based summarizer based on SVM that effectively incorporates different kinds of features including linguistic and prosodic features. We use different kinds of features, such as linguistic and prosodic features. Linguistic information is quite informative not only for text summarization, but also for speech summarization. The linguistic features that we use are listed below, where ChaSen [15] is used as a Japanese morphological analyzer.

Repeated words: We extracted sentences according to the rate of summary that includes two or more frequent words, i.e. “repeated words”, which are basically only nouns, except for some fillers or stop words in a document. The number of repeated words in each sentence is used as a feature.

Words in slide texts: Class lectures are usually performed using slides (e.g. Microsoft PowerPoint®). Therefore, words in slide texts and captions are available and may be good cues. The sum of the appearance of such words in a corresponding slide is also used as a feature.

Term Frequency (TF): The value by Equation (3) is used as the TF. In the experiment, we only count the nouns, except for the stop words and fillers.

CP-based: Our previous work [6] shows that Cue Phrases (CP) based on a CRF based labeler are good cues for important sentence extraction. Although we used a binary feature as to whether a sentence was labeled by a CP labeler or not in [6], we use the number of CP labeled by the CP labeler in this paper.

In speech summarization, we can use prosodic information in addition to linguistic ones, and we can improve speech summarization by using prosodic information [12]. The prosodic features that we use are listed below.

Duration: The duration for each sentence is used.

Power and F0: Power and F0 of each sentence are derived by using ESPS [16]. We use the averages of these items in a sentence as features.

Rate of utterance: The rate of utterance (ROU) of each sentence is also used. The ROU for sentence $i$ is computed as follows:

$$ROU(i) = \sum_{w \in S_i} \frac{\text{mora}(w)}{\text{duration}(i)}, \quad (4)$$

where $\text{mora}(w)$ means the number of mora in $w$ and $\text{duration}(i)$ is the duration of sentence $i$.

Pause: The durations of pauses between previous and current sentences, and between current and post ones are used.

3.2.1. Features

The range in raw value of each feature might differ depending on each lecture, thus all the features were normalized to have mean and standard deviation of 0 and 1, respectively, for each lecture as follows:

$$f_{j,norm}(i) = \frac{f_j(i) - \text{mean}_j}{\text{std}_j}, \quad (5)$$

where $f_j(i)$ means the raw value of feature $j$ derived from sentence $i$, and $\text{mean}_j$ and $\text{std}_j$ are the mean and standard deviations of the values of feature $j$ for all the sentences in a lecture.

We used a discretized value of $f_{j,norm}(i)$, which is the same as in [17]. All normalized feature values are represented by $d_{iv}$ Boolean variables. A normalized feature, which has a minimum value $\text{min}_j$ and a maximum value $\text{max}_j$, is discretized as below:

$$f_{j, disc}(i) = \text{rounddown} \left( \frac{f_{j,norm}(i) - \text{min}_j}{\text{max}_j - \text{min}_j} \cdot \text{div} \right), \quad (6)$$

<table>
<thead>
<tr>
<th>No. of Important Snt.</th>
<th>No. of Isolated Important Snt.</th>
<th>Mean</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>268.0</td>
<td>80.6</td>
<td>1.83</td>
<td>7.0</td>
</tr>
</tbody>
</table>
where \textit{rounddown} is a round-down function after the decimal point. According to \( f_j^{\text{disc}}(i) \), we can obtain \( \text{div} \) Boolean variables. In the experiment, we used \( \text{div} = 5 \). For example, \( f_j^{\text{disc}}(i) = 1 \) is represented by “01000”, and \( f_j^{\text{disc}} = 3 \) is represented by “00010”.

### 3.2.3. Classifier

Before discretization, we combined all the features to represent correlations between features, since overfitting appears when using a complex kernel for an SVM after discretization on our small corpus. In the experiment, we used all the possible combinations between the arbitrary features and the second power of each feature in addition to the original features. The score of sentence \( i \) is calculated as below:

\[
\text{Score}(S_i) = w x + b,
\]

where \( w \) is a vector of the weights for combined features, \( x \) is a vector of the values of combined features, and \( b \) expresses a bias. \( w \) and \( b \) are estimated using an SVM, which classifies \( x \) based on the margin maximization. To train \( w \) and \( b \), we use \( \text{svm}^++ \) [18].

### 4. Proposed Methods

#### 4.1. Summarization taking into account consecutiveness of important sentences

As shown in Section 2.3, there is a consecutiveness of important sentences. Thus, in addition to the features described in Section 3.2.1, we use two features to deal with the consecutiveness: dynamic and difference features. The dynamic feature concerns the decisions made in the previous extraction, and the difference feature deals with the difference between the current and previous feature values.

##### 4.1.1. Dynamic feature

Since important sentences tend to be consecutively extracted, the decision made for the previous sentence might be useful as a feature. Therefore, we just use the decision for the previous sentence as a dynamic feature, which has two Boolean values. The dynamic feature vector for sentence \( i \) is determined as follows:

\[
dynamic(i) = \begin{cases} 10 & \text{if } S_{i-1} \text{ is extracted.} \\ 01 & \text{else.} \end{cases}
\]

##### 4.1.2. Difference feature

The adjacent sentences in a block of important ones should have similar feature values to each other, and therefore, there should be a larger difference in the feature values between an important sentence and an unimportant one. Therefore, we use the difference between the current and previous feature values. The difference feature \( \text{dif} f_{i,j} \) for feature \( j \) from sentence \( i \) is computed as follows:

\[
dif f_{i,j} = f_j(i) - f_j(S_{i-1}).
\]

##### 4.1.3. Search

Since decision for the extraction of a certain sentence depends on the decisions for the other sentences, we cannot make the decision sentence-by-sentence. We assume that current decision making depends on only the previous decision, and thus we can obtain the sequence that has the highest score from all the hypotheses using a dynamic programming technique as follows:

\[
g_{0}(i,j) = \max \begin{cases} g_{0}(i-1, j) \\ g_{1}(i-1, j) \end{cases}
\]

\[
g_{1}(i,j) = \max \begin{cases} g_{0}(i-1, j-1) + \text{score}(i|0) \\ g_{1}(i-1, j-1) + \text{score}(i|1) \end{cases},
\]

where \( i \) is a current sentence number and \( j \) is the number of already extracted sentences. \( g_{1}(i,j) \) and \( g_{0}(i,j) \) are the best scores when sentence \( i \) is extracted or not and when \( j \) sentences have already been extracted. \( \text{score}(i|0) \) and \( \text{score}(i|1) \) are the local scores of sentence \( i \) when previous sentence \( i-1 \) is extracted or not, respectively, and those are computed by Equation \( (7) \).

### 4.2. Feature-based summarization concerning global redundancy

In Section 4.1, we proposed a method which takes into account context information, although it does not take into account a redundancy in a summarization. In this section, we propose an extraction method that takes into account the global redundancy over the entire summarization. To cope with this problem, we incorporate the redundancy criterion of MMR into the feature-based summarization.

#### 4.2.1. Redundancy Feature

The second term of Equation \( (1) \) represents the redundancy of already extracted sentences. We use the redundancy as a new feature. Given a document and the important sentences \( \text{imp} \) in the document, the redundancy feature \( \text{rdun} \) for sentence \( i \) is calculated as follows:

\[
\text{rdun}(i) = \text{Sim}(S_i, \text{Imp}),
\]

\[
\text{Imp} = \begin{cases} \sum_{S \in \text{imp} \backslash \{S_i\}} S \backslash \{S_i\} & \text{if } S_i \text{ is important sentence.} \\ \sum_{S \in \text{imp} \backslash \{S_i\}} S & \text{else.} \end{cases}
\]

where \( \text{Sim} \) is a cosine similarity. We assume the redundancy feature eliminates the redundancy throughout the entire summarization.

#### 4.2.2. Search

Since our redundancy modeling assumes a global dependency over the entire summarization, it is intractable to find the best sequence from all the hypotheses. Thus, we use a beam search and replace \( \text{imp} \) in Eq. \( (13) \) with an average vector of the already-extracted sentences. In the experiment, we examine the beam width of 30 for each \( g(i,j) \) from Equations \( (10) \) and \( (11) \).

### 5. Experiments

#### 5.1. Setups

For MMR, we estimated the \( \lambda \) in Equation \( (1) \), which yields the best \( \kappa \) value from all the documents, i.e. \( \lambda \) is estimated under a closed condition. In the experiments, \( \lambda = 0.6 \) is used.

For feature-based extraction, a 4-fold cross validation is used to evaluate the summarization performance. Thus, when using the lectures given by one professor, the lectures given by the other three are used as training data to train \( w \) and \( b \) in Equation \( (7) \).

#### 5.2. Evaluation Metrics

It is a non-trivial task to find the best summarization metrics. In this paper, we used \( \kappa \)-value [19], \( F \)-measure, and a Rouge [20] metric to measure the summarization performance. These are defined as follows:
In this paper, we propose a novel sentence extraction method that effectively deals with the consequitiveness of important sentences that is based on SVM using the dynamic and difference features. The dynamic feature takes into account the previous decision of extraction, and the difference feature is a difference between the current and previous feature values. By using both the dynamic and difference features simultaneously, we could obtain better results than those from MMR and those using only conventional features. We also adopted the redundancy feature, and we could obtain slight improvement on manual transcripts.

In the future, we will search for a method that deals with redundancy better, and aim to use more complex contextual information, such as coreference or the long distance correlation beyond the shallow consequitiveness of important sentences that were dealt with in this paper.

7. References