Sentence Extraction with Support Vector Machine Ensemble

Minh Le Nguyen, Akira Shimazu, Xuan Hieu Phan
Tu Bao Ho and Susumu Horiguchi
Japan Advanced Institute of Science and Technology
1-1 Asahidai, Nomi, Ishikawa 923-1292, Japan

ABSTRACT

This paper addresses a support vector machine model for text summarization problem. First, we formulate the text summarization problem as the problem of extracting a set of importance sentences. We then employ a support vector model for sloving that problem. Although the SVM are shown to be very suitable for sentence extraction because of the abillty in dealing with a very large of feature demision. The limitation of it is that in practical some approxiamtion algorithm are used. It might reduce the accuracy of classification. To overcome the above drawback, a SVM ensemble is clearly sutiable. This was because when combining each individual SVM has been traiend independently from the random chosen training samples and the correctly classifed area in the sapce of data samples of each SVM becomes limited to a certain area. We can expect that a combination of several SVMs will exapand the correctly classified area incrementlly. This paper initialy presents the use of ensemble SVM to text summarization and shows that the performance of SVM ensemble will be better than that of convential SVM.
Keywords: Text summarization, sentence extraction, SVM, Ensemble learning, SVM ensemble.

1. INTRODUCTION

Sentence extraction is the task of identifying important sentences in the text. The majority of early extraction research focused on the development of relatively simple surface-level techniques that tend to signal important passages in the source text. Typically, a set of features is computed for each passage, and ultimately these features are normalized and summed. The passages with the highest resulting scores are sorted and returned the extract. Early techniques for sentence extraction computed a score for each sentence based on feature such as position in the text [1], word and phrase frequency [2], key phrase (e.g., "In conclusion") [3]. Recent extraction approaches use more sophisticated techniques for deciding which sentences to extract; these techniques often rely on machine learning to identity important feature, on natural languages analysis to identify key passages, or on relations between words rather than bags of words. Approaches involving more sophisticated natural language analysis to identify key passages rely on either of word relatedness or of discourse structure. Some research uses the degree of the correctness between potential passages and the remainder of the text document. The correctness may be measured by the number of share words, synonyms, or anaphora. Other research rewards passages that include topic words; that is, words that have been determined to correlate well with the topic of interest to the user or with the general theme of the source text [4].

The application of machine learning to summarization was pioneered by Kupiec, Pedersen, and Chen [5]. In these work they developed a summarizer using a Bayesian Classifier to combine features from a corpus of scientific articles and their abstracts. Aone et al. [5] and Lin [6] experimented with other forms of machine learning and its effectiveness. Learning individual features has been also reported by Lin and Hovy [7] and Mital [8]. In these tasks, the affect of position sentences, important words and phrases to the selection of sentences were investigated. Some recent works has turned to the use of hidden Markov Model (HMMs) and pivoted QR decomposition to reflect the fact that the probability of inclusion of a sentence in an extract depends on whether the previous sentence has been included as well. An alternative to sentence extraction using learning approach are

proposed by [9]. In this method, the author indicated that using Support Vector Machine was well suited for sentence extraction. It also showed an advantage in comparing with earlier sentence extraction methods due to using high dimension space of features. However, the SVM has a drawback that since learning of the SVM is a very time consuming for a large scale of data, so some approximate algorithm are used. Although it has an advantage that reducing the computation time, but degrade the classification performance. To overcome the above drawback, a SVM ensemble is clearly sutiable. This was because when combining each individual SVM has been traiend independently from the random chosen training samples and the correctly classifed area in the sapce of data samples of each SVM becomes limited to a certain area. We can expect that a combination of several SVMs will exapand the correctly classified area incrementlly. This paper initially proposes the use of ensemble SVM to text summarization and shows that the performance of SVM ensemble will be better than that of conventional SVM. We also initially build a Vietnamese text summarization corpus which is helpful for studying text summarization with Vietnamese language. Experimental results on that corpus show that the performance of SVM ensemble is better than that of conventional SVM for text summarization

2. SUPPORT VECTOR MACHINENSEMBLE

This section introduces the support vector machine ensemble [12] which is based on a boosting strategy to combine several support vector machines. It is known that an ensemble often shows much better performance than the individual classifiers that make it up. For example, an ensemble of *n* classifiers: $\{f_1,f_2,...,f_n\}$ for a test data *x* returning the decision $f_i(x)$ $i=1,n$ with their error are uncorrelated can be better than individual classifier by using a simple voting method. There are two phases in a structured SVM ensemble which are training and testing phase. During the training phase, each individual SVM is trained independently by its own replicate training data set via a bootstrap method. All constituent SVMs will be aggregated by various combination strategies such as boosting or bagging. In the testing phase, a test example is applied to all SVMs simultaneously and a collective decision is obtained based on the aggregation strategy. Our aggeration strategy is simply using a major voting method.

3. SENTENCE EXTRACTION WITH SVM ENSEMBLE

The SVM ensemble method is introduced in [12] showed that it improve the accuacry for the data in the UCI tasks. This section shows the SVM ensemble method for sentence extraction problem.

3.1 Learning with SVM Ensemble

Input: A set of TR of l labelled examples:
$S = (x_i; y_i), i = 1, 2, ..., l$ x_i is the sentence and y_i is the tree structure
$p_0(x) := 1/l;$
for $k = 1$ *to* K **do**
  Build $TR_{boost_k} = (x_i; y_i)|i = 1, 2, ..., l'$ based on the $p_{k-1}(x_i)$;
  Train the kth SVM h_k using TR_{boostk};

$$\varepsilon_k = \sum_{i=1}^{l} p_i(i)|\{i|h_i(x_i) \neq y_i\}|;$$
$$\alpha_k = \frac{1}{2}\ln(\frac{\varepsilon_k}{1-\varepsilon_k});$$

  for $i = 1$ *to* l **do**

$$p_{k+1}(x_i) = \frac{p_k(x_i)}{Z_k} \times \begin{cases} \exp(-\alpha_k) \text{if} h_k(x_i) = y_i \\ \exp(\alpha_k) \text{if} h_k(x_i) \neq y_i \end{cases}$$

    where Z_k is a normalization factor to make $\sum_{i=1}^{l} p_{k+1}(x_i) = 1$
  end
end

Algorithm 1: The AdaBoost algorithm

Our method for learning is based on the boosting stratagy as described following: We follow the behaviour of Addboosting to select training data for each individula SVM. In the first step, Each SVM is trained using a different training set. Assuming that we have a training set $TR = \{(x_i; y_i) \mid i = 1, 2, ..., l\}$ consisting of *l* whose samples and each sample in the *TR* is assigned to have the same value of weight $p_0(x_i) = \frac{1}{l}$. For training the *k*th SVM classifier, we build a set of training samples

$TR_{boost_k} = \{(x_i; y_i) \mid i = 1,2,...,l'\}$ that is obtained by selecting *l' (<l)* samples among the whole data set *TR* according to the weight value $p_{k-1}(x_i)$ at the *(k-1)th* iteration. The training samples is used for training the *kth* SVM classifier. Then, we obtained the updated weight values $p_k(x_i)$ of the training samples as follows. The weight values of the incorrectly classified samples are increased but the weight values of the correctly classified samples are decreased. This shows that the samples which are hard to classify are selected more frequently. This updated weight values will be used for building the training samples $TR_{boost_{k+1}} = \{(x_i; y_i) \mid i = 1,2,...,l'\}$ of the *(k+1)*th SVM classifier. The sampling procedure will be repeated until *k* training samples set has been build for the *kth* SVM classifier.

3.2 Testing with SVM Ensemble

In the testing method, we need to classify a given input sentence to label "true" or label "false" to indicate that whether the sentence is important or not. The method here is mainly based Majority voting.

Let f_k (k=1,2,…,K) be a decision function of *k*th SVM in the SVM ensemble and C_j (*j=1,2,..,C*) denote a label of the jth class. Then let N_j is the number of SVMs whose decisions are know to the jth class. Here we need only two N_0 and N_1, in which N_0 is the number of SVMs which have label true and N_1 is the number of SVMs which have label "false". If N_1 is gerater than N_0 then we obtained a class label true, otherwise we obtain a class label false.

3.3 Feature for SVM ensemble

The most important problem in sentence extraction is designed feature sets. In this paper we present a set of feature for an individual SVM. Our features including one some method bellow:

- **Location method**: Including the position of sentences within documents. These sentences in the beginning or in the end of a given text document are highly relevant to the text's gist meaning.
- **Length method**: These short sentences are preferred to these important sentences. The length here means the number of words in the sentence.
- **Relevant to title**: These sentences are closed to the title of a given texts are more important.
- **term frequent and document frequent**
- **cue phrase:** The term 'cue phrase' covers the kinds of stock phrases which are frequently good indicators of rhetorical status (e.g. phrases such as *The aim of this study* in the scientific article domain and *It seems to me that*).
- **distance of a word within** a sentence to its previous occupance.
- **Words** information with its frequent is gerater than a specific threshold.

4. EXPERIMENTAL RESULTS

This section show the experimental results when using support vector machine ensemble for Vietnamese documents. We collected 900 documents on the website http://www.vnexpress.net. Those documents are mainly on the domain of informatic. We annotated sentences in a document using two class labels:
label +1 (stands for an important sentence) and label -1 stands for a un-importance sentence.

Table 1. Example of a document in the corpus

```
<?xml version="1.0" encoding="utf-8"
standalone="yes" ?>
<document>
<title>Công nghệ bảo mật mới và phần mềm dành cho
Cluster Server</title>
<fpa>
<s label="+1">Trong nỗ lực tăng tính bảo mật cho các
máy tính nối mạng, phòng nghiên cứu Bell Lap thuộc
Licent Technologies đã công bố phần mềm bảo mật mới
chạy trên các hệ điều hành Plan9, Unix, Linux, Solaris và
cả Windows.</s>
<s label="-1">Điểm nổi bật của công nghệ mới là lưu trữ
dữ liệu và quá trình xác thực người sử dụng
(Authentication) sẽ được phân làm hai công đoạn tách biệt
do hai phần mềm đảm trách.</s>
<s label="-1">Qua đó, người truy cập có thể hoàn toàn chủ
động khi đưa ra những thông tin cá nhân của mình trong
các cuộc giao dịch trực tuyến.</s>
</fpa>
<p>
</document>
```

In order to use the SVM ensemble to obtain a set of important sentences within a document. We build our own Vietnamese text summarization corpus in which a set of 500 text documents is made. Each document has approximately 50 sentences, the average length of a sentence is 10. We used the SVM-Light (available at http://svmlight.joachims.org/) to train our SVM model for the corpus. The ensemble version of SVM is based on the Algorithm 1, in which we set the parameter *K* to 10. The testing is simply a voting method as presented in section 3.2. To test our result, we used ten-folds cross validation test and the evaluation result is

measured by precision and recall. The equation bellow shows how to compute the precision and recall value for evaluating our sentence extraction problem.

$$precision = \frac{\#\ corrected\ labels}{\#\ extracted\ sentences}$$

$$recall = \frac{\#\ corrected\ labels}{\#\ standard\ extracted\ sentences}$$

$$F\text{-}measure = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

Interestingly, our SVM ensemble achived 0.53 F-measure in comprision with indiviudal SVM 0.51 F-measure. It shows that SVM ensemble are promising to be used as a novel method for text summarization.

5. CONCLUSIONS

In this paper, we propose a novel text summarization method based on the ensemble SVM classification. We initialy build a Vietnamese corpus in order to test the performance of SVM ensemble. It shows that the ensemble method is better than that of the individual SVM.

REFERENCES

[1] Aone, Chinatsu, Mary Ellen Okurowski, Jame Gorlinsky, and Bjornar Larsen: "A Trainable summarizer with knowledge acquired from robust NLP techniques", In I. Mani and M. T. Maybury, editors, Advances in Automatic Text Summarization. MIT Press, Cambiridege, pp. 71-80.

[2] P.B. Baxendale: "Man-made index for technical literature- An experiment", IBM Journal of Research and Development, vol. 2(4), pp.354-361 (1958).

[3] H.P. Edmunson: "New methods in automatic extracting", Journal of the Association for Computing Machinery 16(2): 264-285 (1958).

[4] H.P. Luhn, "The automatic creation of literature abstracts", IBM Jouranal of Research Development, vol. 2(2): 1959-165 (1958).

[5] J. Kupiec, Jan O. Pedersen, and F. Chen: "A trainable document summarizer", In Research and Development in Information Retrieval", pp 68-73 (1995).

[6] J.M. Conroy et al: "Using HMM and Logis-tic Regression to Generate Extract Summaries for DUC", Proceeding Document Understanding Conference, 2001.

[7] S. Teufel and M. Moens: "Sentence extraction as a classification task", Proceeding ACL/EACL-97 Workshop on Intelligent and Scalable Text Summarization, 1997.

[8] T. Hirao, H. Isozaki, E. Maeda, and Y. Matsumoto: "Extracting important sentences with support vector machines", In Proceeding of COLING 2002, pages 342-348, 2002.

[9] Stralkowski, Tomek, G. Stein, J. Wang, and B. Wise, "A robust practical text summarizer", In I. Mani and M.T. Maybury editor, Advances in Automatic text summarization. MIT Press, Cambridge, pp. 137-154 (1999).

[10] C. Kim, et.al., "Constructing support vector machine ensemble", Pattern Recognition 36 (2003) 2757-2767