Semi-Supervised Extractive Speech Summarization via Co-Training Algorithm

Shasha Xie[1], *Hui Lin*[2], *Yang Liu*[1]

[1]Department of Computer Science, The University of Texas at Dallas, U.S.A
[2]Department of Electrical Engineering, University of Washington, U.S.A
{shasha,yangl}@hlt.utdallas.edu, hlin@ee.washington.edu

Abstract

Supervised methods for extractive speech summarization require a large training set. Summary annotation is often expensive and time consuming. In this paper, we exploit semi-supervised approaches to leverage unlabeled data. In particular, we investigate co-training algorithm for the task of extractive meeting summarization. Compared with text summarization, speech summarization task has its unique characteristic in that the features naturally split into two sets: textual features and prosodic/acoustic features. Such characteristic makes co-training an appropriate approach for semi-supervised speech summarization. Our experiments on ICSI meeting corpus show that by utilizing the unlabeled data, co-training algorithm significantly improves summarization performance when only a small amount of labeled data is available.

Index Terms: extractive meeting summarization, co-training, semi-supervised learning

1. Introduction

Automatic meeting summarization is a very useful technique that can help the users to browse a large amount of meeting recordings. In this paper, we investigate extractive summarization, in which the most representative segments from the original document are selected and concatenated together to form a final summary. This task can be formulated as a binary classification problem and solved using supervised learning approaches. Each training and testing instance (i.e., a sentence) is represented by a set of indicative features, and positive or negative labels are used to indicate whether this sentence is in the summary or not. In previous work, various classification models have been investigated, such as hidden Markov model (HMM), conditional random field (CRF), maximum entropy classifier, and support vector machine (SVM) [1, 2, 3, 4].

Learning a summarization classifier requires a large amount of labeled data for training. Summary annotation is often difficult, expensive, and time consuming. Annotation of meeting recordings is especially hard because the documents to be summarized are transcripts of natural meetings that have very spontaneous style, contain many disfluencies, have multiple speakers, and are less coherent in content. It is very hard to read and understand the document, not to mention extracting the summary. On the contrary, meeting recordings and their transcripts are relatively much easier to collect. This situation creates a good opportunity for semi-supervised learning that can use large amount of unlabeled data, together with the labeled data, to build better classifiers. This technique has been shown to be very promising in many speech and language processing tasks, such as question classification, web classification, word-sense disambiguation and prosodic event detection [5, 6, 7, 8]. However, there is relatively less work of using semi-supervised learning approaches for the automatic summarization task. In [9], the authors used co-training algorithm to exploit unlabeled data for extractive text summarization. Two classifiers (probabilistic SVM and naive Bayesian classifier) were trained on the *same* feature spaces. One assumption of co-training, however, is that features can be split into two sets. To the best of our knowledge, there is little research on the application of semi-supervised learning techniques on extractive speech summarization, except [10] that uses active learning.

In this paper, we use the co-training algorithm [11] to explore semi-surprised learning in speech summarization. Co-training assumes that features can be split into two sets, which are conditionally independent given the class, and each of which is sufficient to train a good classifier. Unlike in text summarization task (where only textual information is available), we can easily extract two different views of features for speech summarization: one from the textual transcripts, and the other from the speech recordings[1]. Previous work [12] showed that each of these two feature sets can achieve good performance for speech summarization. Also, as these two sets of features are essentially different, we can reasonably assume that the conditional independence assumption holds. Therefore, co-training is a vary natural and appropriate approach for semi-supervised speech summarization. Our experimental results on the ICSI meeting corpus show that the co-training algorithm can effectively use the unlabeled data and improve the summarization performance upon only using a small set of labeled data.

2. Corpus

We use the ICSI meeting corpus [13], which contains 75 recordings from natural meetings. Each meeting is about an hour long and has multiple speakers. These meetings have been manually transcribed and annotated with dialog acts (DA), topic segments, and extractive summaries. The same 6 meetings as in [2, 4, 12] are used as the test set. Furthermore, 6 other meetings are randomly selected to construct a development set, and the remaining meetings are used for training. We use three reference summaries from different annotators for each meeting in the test set. For the training and development set, we only have one reference summary. The lengths of the reference summaries are not fixed and vary across annotators and meetings. The average word compression ratio, that is the ratio of the number of words in the summary and the original meeting, is 14.3%, with

[1] Note that in this paper we use textual features to represent the feature set extracted from non-prosodic information, which includes speaker and topic information other than plain texts.

a standard deviation of 2.9% for the test set. In this paper, we extract the summaries with a word compression ratio of 15%.

To evaluate summarization performance, we use ROUGE [14], which has been used in previous studies of speech and text summarization. ROUGE compares the system-generated summary with reference summaries (there can be more than one reference summary), and measures different matches, such as N-gram, longest common sequence, and skip bigrams. In this paper, we report ROUGE-1 F-measure scores to make our research comparable with previous work.

3. Supervised Extractive Summarization

The extractive summarization task can be considered as a binary classification problem and solved using supervised learning approaches. The label for each instance represents whether it is a summary sentence or not. Each training and testing sample (i.e., a sentence) is represented by a large set of indicative features. We use support vector machines (SVM) (the LibSVM implementation [15]) as the classifier because of its superior performance in many binary classification tasks. During training, an SVM model is trained using the labeled training data. Then for each sentence in the test set, we predict its confidence score of being included into the summary. The summary for the test document is obtained by selecting the sentences with highest scores until the desired compression ratio is reached. For meeting summarization, the features we use can be naturally split into two different views, textual and prosodic/acoustic features, which are described below.

3.1. Textual Features

The textual features we use are described in details in [4], including lexical, discourse, structural and topic-related information. The lexical features include the sentence length, the number of words in each sentence after removing stop words, the number of frequent words and bigrams, and the number of nouns or pronouns that appear for the first time in a sentence. In addition, we derive various TF (term frequency) and IDF (inverse document frequency) related features (e.g., max, mean, sum). The cosine similarity between the sentence and the entire meeting transcript is also included in the feature set. We compute some topic-related features to capture the characteristics of different topics within a meeting. Furthermore, because the meeting corpus has multiple participants, we create some features to indicate speaker information, such as whether the sentence is said by main speakers (measured by the words they speak in the meeting), whether there is a speaker change compared to the previous sentence, and how term usage varies across speakers in a given meeting. In total, there are 57 features in this category.

3.2. Prosodic/Acoustic Features

With the availability of meeting recordings, we also extract prosodic/acoustic features for each sentence sample. We use Praat [16] to compute the raw pitch and energy values, and derive various features from these. There are 13 original features including five F0 related features, five energy related features, the sentence duration, and two features representing the speaking rate. In addition to these raw features, we have different normalized features based on various information, such as the speaker, the topic segmentation, and contextual information. Finally, we include the prosodic delta features — the difference between the current instance's feature values to its previous M and next M instances. The total number of features for this category is 189. In our previous research, the experimental results showed that using these prosodic features alone we can get better performance than that of using the textual features. More details can be found in [12].

4. Co-Training Algorithm for Meeting Summarization

Co-training algorithm was introduced to increase the classification accuracy by exploiting the information from a large amount of unlabeled data, together with a small set of labeled data [11]. Co-training assumes that the features can be split into two independent sets, and each set is sufficient to train a good classifier. Initially two separate classifiers are trained with the labeled data, on the two sub-feature sets respectively. Each classifier then classifies the unlabeled data, selects the samples that they feel most confident with, and uses these automatically labeled samples along with the original labeled data to "teach" the other classifier. This process iterates until the classification performance stabilizes, or all the unlabeled data is used, or after certain number of iterations. There are several possible ways to apply co-training algorithm to extractive speech summarization. We investigate two methods in this study.

4.1. Sentence-based selection

Algorithm 1 Co-Training for Extractive Speech Summarization

```
Let $L$ be the set of labeled training sentences.
Let $U$ be the set of unlabeled training sentences.
Each sentence is represented by two feature sets $\{F_1, F_2\}$,
representing textual and prosodic features respectively.
Initialize training set for the two classifiers: $L_1 = L_2 = L$
while $U \neq \emptyset$ do
    Train the first classifier $\mathcal{C}_1$ on $L_2$ using $F_1$.
    Train the second classifier $\mathcal{C}_2$ on $L_1$ using $F_2$.
    for each classifier $\mathcal{C}_i (i = 1, 2)$ do
        (a) For each sentence in $U$ (represented by $F_i$), $\mathcal{C}_i$ predicts its posterior probabilities of being labeled as positive;
        (b) $\mathcal{C}_i$ chooses $p$ sentences ($P$) that it most confidently labels as positive and $n$ sentences ($N$) that it most confidently labels as negative from $U$;
        (c) $\mathcal{C}_i$ removes $P$ and $N$ from $U$;
        (d) $\mathcal{C}_i$ adds $P$ to $L_i$ with positive labels, and $N$ to $L_i$ with negative labels.
    end for
end while
```

In the classification setup for extractive speech summarization, each training or testing instance is a sentence from the document to be summarized, and positive or negative labels are used to indicate whether or not this sentence is in the summary. We therefore use sentence as the basic selection unit in each co-training iteration. Precisely, p unlabeled sentences are labeled as positive (summary sentences) and n unlabeled sentences are labeled as negative in each iteration based on the confidence scores of current classifier's prediction. These $p + n$ sentences are then added into the original training set to form a new train set. The detail of the algorithm is described in Algorithm 1, which basically follows the standard procedure of co-training.

Note that for extractive summarization task, the positive samples are only a small percent of all the instances in the document (i.e. summary is always compact). For example, in the ICSI meeting corpus the average percentage of the positive samples is 6.62%. In order to be consistent with the original training data distribution, we select more negative samples in each iteration than positive ones. We use $n = \alpha p$, where α is the ratio of the number of negative samples to the number of positive samples in the corpus ($\alpha = 15$ in our case).

4.2. Document-based selection

In sentence-based selection, the classifier labels sentences independently, regardless which document a sentence belongs to. In summary annotation, however, the decision for each sentence is not made independently, rather it is made by considering the *entire* document. This is a key difference between the classification setup for summarization vs. other classical classification tasks. In order to be consistent with human labeling process, we could use document as the basic selection unit such that the entire document can be included into the training set. Similar to the sentence-based selection method above, we select the documents that the classifiers are most confident with, and adding all the sentences in these documents into the training set for next iteration. To assign a confidence score to each document, we take the classifier's average confidence scores for all the sentences in the document. For each sentence in the document, we estimate the classifier's confidence by taking the negative entropy of the posterior distribution. The confidence score of a document D is then measured as the average negative entropy of all the sentences in D. At each iteration, the documents with high confidence scores are selected into the labeled training set. For each of the selected documents, we select the top $\frac{100}{1+\alpha}\%$ of its sentences that the classifier most confidently labels as positive. These sentences are labeled as positive, and the rest are labeled as negative.

5. Experimental Results

For co-training, we first train two classifiers using the labeled data, on the textual and prosodic feature sets respectively. Then the additional training samples are iteratively selected according to the predictions from each classifier. After co-training, we have two classifiers trained from the textual and prosodic features respectively. We then extract the summaries for each document in the development set using these two classifiers. In order to evaluate the effect of the size of the initial labeled data, we use different size, starting from 10% of the training data to all of it. When a subset of the training set is used, the rest is treated as unlabeled data (since we do not have additional meeting data with similar style and topics). The subset of labeled data is randomly selected from the training set. For each setup, we run 10 independent trials, and report the average score. Figure 1 shows the co-training results using textual features (upper figure) and prosodic features (lower graph) on the development set. For a comparison, we also include the baseline results of supervised learning (dotted line in the figure) that just uses the initial labeled data without any unlabeled data.

From Figure 1, we can see that for the supervised baseline, the results of using prosodic features are consistently better than those using textual features. In general, performance improves when using more data for training, which is as expected. The worst results are always observed when the labeled training data is 10% of the original data set. The summarization results using these two feature sets are very competitive comparing to those reported in previous work.

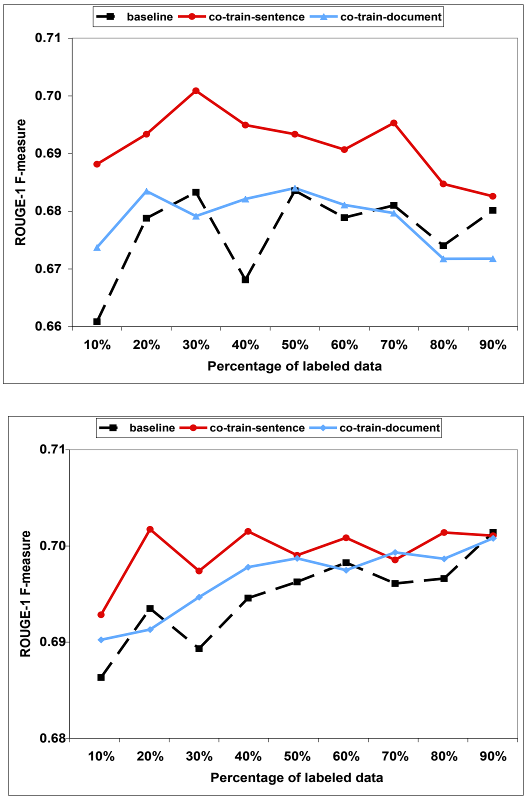

Figure 1: Co-training results on the development using the models trained from textual (upper) and prosodic features (lower).

Using sentence-based selection, co-training achieves significantly better performances than that of only using the labeled data, on both textual and prosodic features. Consistent improvements are achieved on different sizes of labeled data. The improvement is more obvious when the labeled training data set is small. When increasing the labeled data set, the differences between the co-training and baseline results decreases, since the unlabeled data that co-training algorithm exploits is getting smaller. Another interesting observation is that comparing to the baseline results, where the performance using prosodic features are consistently better than those using textural features, the results on textual features are remarkably improved by co-training, which are now competitive to the results of using prosodic features. This may be partly because of the original better performance of the prosodic classifier that adds more correct samples for textual classifier training.

For the document-based sample selection, there is no consistent improvement over the baseline results, and for some of the setups it is even worse than the baseline. One possible reason for that is that during co-training, although we select the documents with the highest confidence scores at each iteration, the classifier is not necessarily confident to *all* of the sentences in the document. In other words, the chance of of adding misclassified samples increases, which could have potential impact on the learning and labeling process in the following iterations and thus hurt the co-training performance. Nevertheless, we believe the selection criteria can be improved by making it more directly optimized for the summarization task. Possible future directions include making a partial document selection, i.e., only adding the most confident samples in the selected docu-

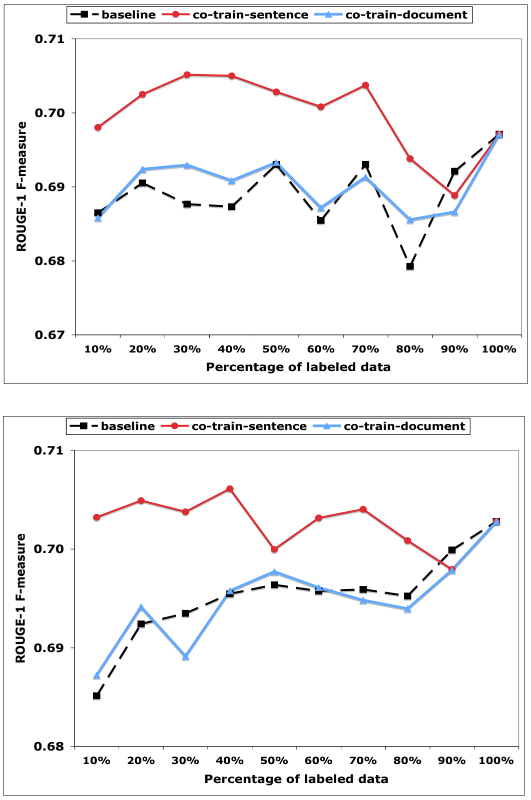

Figure 2: Co-training results on the test using the models trained from textual (upper) and prosodic features (lower).

ment such that the selected samples can be both coherent and confident. From another point of view, since the supervised summarization classification approach itself is only dependent on individual sentences, and does not use any document level decision, it does not need document information for its iterative training process (features based on document information have been prepared offline already). This is different from active learning where the selection units have to be documents for humans to create reference labels.

The test set results are shown in Figure 2 using textual features (upper figure) and prosodic features (lower). Similar to the results on the development set, co-training algorithm with sentence-based selection can effectively use the information of unlabeled data to improve the summarization performance. The improvement is more significant when the labeled data set is small. Again, no promising results are obtained using documents as the selection units.

6. Conclusion and Future Work

Summary annotation for speech recordings is time-consuming and expensive. Semi-supervised learning is a very useful technique that can increase the classification accuracy by leveraging the information from a large amount of unlabeled data. Co-training is known to be very effective if the features can be divided into two conditionally independent sets. For speech summarization, two different sets of features, textual features and acoustic/prosodic features, are naturally available, making co-training a good choice of semi-supervised speech summarization. Our experimental results verify this claim. Results on the ICSI meeting corpus showed that co-training algorithm with sentences as the selection units can effectively improve the summarization performance evaluated by ROUGE scores. Both classifiers trained using textual and prosodic features work better than only using the labeled data, with more gain for the relatively weak classifier (using textual features). In the future work we will investigate other sample selection criteria that are more geared towards the summarization task. We will also investigate other semi-supervised learning algorithms, such as transductive SVM or graph-based semi-supervised learning approaches.

7. Acknowledgments

The authors thank Dilek Hakkani-Tür for the help with prosodic features. This work is supported by NSF award IIS-0845484. Any opinions expressed in this work are those of the authors and do not necessarily reflect the views of NSF.

8. References

[1] G. Murray, S. Renals, and J. Carletta, “Extractive summarization of meeting recordings,” in *Proc. of Interspeech*, 2005.

[2] M. Galley, “A skip-chain conditional random field for ranking meeting utterances by importance,” in *Proc. of EMNLP*, 2006.

[3] A. H. Buist, W. Kraaij, and S. Raaijmakers, “Automatic summarization of meeting data: A feasibility study,” in *Proc. of CLIN*, 2005.

[4] S. Xie, Y. Liu, and H. Lin, “Evaluating the effectiveness of features and sampling in extractive meeting summarization,” in *Proc. of IEEE Spoken Language Technology (SLT)*, 2008.

[5] Z.-Y. Niu, D.-H. Ji, and C.-L. Tan, “Word sense disambiguation using labeled propagation based semi-supervised learning,” in *Proc. of ACL*, 2005.

[6] N. T. Tri, N. M. Le, and A. Shimazu, “Using semi-supervised learning for question classification,” in *Proc. of ICCPOL*, 2006.

[7] R. Liu, J. Zhou, and M. Liu, “A graph-based semi-supervised learning algorithm for web page classification,” in *Proc. of sixth International conference on Intelligent Systems Design and application*, 2006.

[8] J. H. Jeon and Y. Liu, “Semi-supervised learning for automatic prosodic event detection using co-training algorithm,” in *Proc. of ACL-IJCNLP*, 2009.

[9] K.-F. Wong, M. Wu, and W. Li, “Extractive summarization using supervised and semi-supervised learning,” in *Proc. of ACL*, 2008.

[10] J. J. Zhang and P. Fung, “Active learning of extractive reference summaries for lecture speech summarization,” in *Proc. of ACL-IJCNLP*, 2009.

[11] A. Blum and T. Mitchelli, “Combining labeled and unlabeled data with co-training,” in *Proc. of the Workshop on Computational Learning Theory*, 1998.

[12] S. Xie, D. Hakkani-Tur, B. Favre, and Y. Liu, “Integrating prosodic features in extractive meeting summarization,” in *Proceedings of ASRU*, 2009.

[13] A. Janin, D. Baron, J. Edwards, D. Ellis, D. Gelbart, N. Morgan, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke, and C. Wooters, “The ICSI meeting corpus,” in *Proc. of ICASSP*, 2003.

[14] C.-Y. Lin, “ROUGE: A package for automatic evaluation of summaries,” in *Proc. of The Workshop on Text Summarization Branches Out*, 2004.

[15] C. Chang and C. Lin, “LIBSVM: a library for support vector machines,” 2001, software available at http://www.csie.ntu.edu.tw/˜cjlin/libsvm.

[16] P. Boersma and D. Weenink, “Praat: doing phonetics by computer,” http://www.praat.org/, 2006.