Co-training Using Prosodic and Lexical Information for Sentence Segmentation

Umit Guz[1], *Sébastien Cuendet*[1], *Dilek Hakkani-Tür*[1], *Gokhan Tur*[2]

[1]International Computer Science Institute, Berkeley, CA, USA
[2]SRI International, Speech Technology and Research Laboratory, Menlo Park, CA, USA
{guz,cuendet,dilek}@icsi.berkeley.edu, gokhan@speech.sri.com

10.21437/Interspeech.2007-585

Abstract

We investigate the application of the co-training learning algorithm on the sentence boundary classification problem by using lexical and prosodic information. Co-training is a semi-supervised machine learning algorithm that uses multiple weak classifiers with a relatively small amount of labeled data and incrementally uses unlabeled data. The assumption in co-training is that the classifiers can co-train each other, as one can label samples that are difficult for the other. The sentence segmentation problem is very appropriate for the co-training method since it satisfies the main requirements of the co-training algorithm: the dataset can be described by two disjoint and natural views that are redundantly sufficient. In our case, the feature sets are capturing lexical and prosodic information. The experimental results on the ICSI Meeting (MRDA) corpus show the effectiveness of the co-training algorithm for this task.

Index Terms: co-training, sentence segmentation, prosody, self-training, Boosting

1. Introduction

Co-training is a very effective machine learning technique that has been used successfully in several classification tasks like web page classification, word sense disambiguation, and named-entity recognition [1-6,among others]. Co-training is a semi-supervised learning method that aims to improve performance of a supervised learning algorithm by incorporating large amounts of unlabeled data into the training data set. Co-training algorithms work by generating two or more classifiers trained on different views of the input labeled data that are then used to label the unlabeled data separately. The most confidently labeled examples of the automatically labeled data can then be added to the set of manually labeled data. The process may continue for several iterations. In this paper, we describe the application of the co-training method for sentence segmentation where we use prosodic and lexical information as two views of the data.

Sentence segmentation from speech is part of a process that aims at enriching the unstructured stream of words that are the output of standard speech recognizers. Its role is to find the sentence units in this stream of words. Sentence segmentation is a preliminary step toward speech understanding. It is of particular importance for speech-related applications, as most of the further processing steps, such as parsing, machine translation and information extraction, assume the presence of sentence boundaries [7,among others]. Once the sentence boundaries are detected, further syntactic and/or semantic analysis can be performed on these sentences.

Usually, speech recognizer output lacks the textual cues to these entities (such as headers, paragraphs, sentence punctuation, and capitalization). However, speech provides extra nonlexical cues, related to features like pitch, energy, pause and word durations, named as prosodic features. It has been shown that for segmentation of speech into sentences, prosodic and lexical cues provide complementary information. In our previous work we proposed methods to combine them to improve performance of the segmentation system [8].

Although statistical methods are widely used for sentence segmentation, the drawback is that they require significant amounts of labeled data, which is expensive, time-consuming, and laborious to prepare. In our earlier work we proposed supervised model adaptation methods for sentence segmentation using a small amount of labeled in-domain data and a large amount of labeled out-of-domain data [9]. This paper focuses on semi-supervised training of sentence segmentation models without exploiting any out-of-domain data using co-training compared with the traditional semi-supervised training approach of self-training.

In this study, we consider the speech features (lexical and prosodic) as two disjoint and natural feature sets or views and we try to improve performance of the baseline by using these feature sets with the co-training algorithm. Starting with a very small number of labeled data from which lexical and prosodic features can be extracted, the aim is to increase the amount of labeled data by using large amounts of unlabeled data.

In the next section we present related work on co-training and then describe our sentence segmentation and co-training approaches in Section 3. We provide experimental results using self-training and co-training with the ICSI Meeting Recorder Dialog Act (MRDA) corpus in Section 4.

2. Related Work

The co-training approach was first introduced and performed by Blum and Mitchell [1-2]. The main goal is using multiple views together with unlabeled data to augment a much smaller set of labeled examples. More specifically, the presence of multiple distinct views of each example can be used to train separate models for the same task, and then each classifier's predictions on the unlabeled examples are used to augment the training set of the other classifier. Figure 1 presents the basic algorithm. The task Blum and Mitchell used was identifying the web pages of academic courses from a large collection of web pages collected from several computer science departments. Their co-training implementation had two natural feature sets: the words that are present in the course web page and the words that are used in the links pointing to that web page. For this task, both views of examples are considered as sufficient for learning. Blum and Mitchell showed that co-training is probably approximately correct (PAC) learnable when the two views are individually

sufficient for classification and conditionally independent given the class. Their results showed that the error rate of the combined classifier was reduced from 11% to 5%.

There has been much effort on investigating the effectiveness of the co-training algorithm in different domains and applications. In recent work [3], it is shown that the independence assumption can be relaxed, and co-training is still effective under a weaker independence assumption. In that work, a greedy algorithm to maximize the agreement on unlabeled data is proposed. This resulted in improved results in a co-training experiment for named entity classification. It is shown that the rate of disagreement between two classifiers with weak independence is an upper bound on the co-training error rate.

In [4], co-training was applied to the e-mail classification task. In this work, it was found that performance of the co-training was sensitive to the learning algorithm used. In particular, co-training with Naïve Bayes did not result in better performance. However, this was not case with support vector machines. The authors explained this situation with the inability of the Naïve Bayes to deal with large sparse datasets. This explanation was also confirmed by significantly better results after feature selection.

Other work [5-6] was about investigation of the sensitivity of the co-training to the assumptions of conditional independence and redundant sufficiency. In the first experiment, co-training was applied to the web page database from [1]. The results showed that co-training using Naïve Bayes was not better than Expectation Maximization even when there is a natural split of features. Both Expectation Maximization and co-training with Naïve Bayes improved performance of the initial classifier by approximately 10%. The second experiment was performed on a dataset that had been created in a semi-artificial manner so that the two feature sets are truly conditionally independent. In addition, the condition of redundantly sufficient features was met, since the Naïve Bayes trained on each of the data sets separately was able to obtain a small error rate. It was found that co-training with Naïve Bayes well outperformed Expectation Maximization, and even outperformed Naïve Bayes trained with all examples labeled. Their third experiment involved performing co-training on a dataset whereby a natural split of feature sets is not used. The two feature sets were chosen by randomly assigning all the features of the dataset into two different groups. This was tried for two datasets: one with a clear redundancy of features, and one with an unknown level of redundancy and nonevident natural split in features. The results indicated that the presence of redundancy in the feature sets gave the co-training algorithm a bigger advantage over Expectation Maximization. The results of these experiments verified that the co-training has a considerable dependence on the assumptions of conditional independence and redundant sufficiency. However, even when either or both of the assumptions are violated, the performance of co-training can still be quite useful in improving a classifier's performance. We believe that the sentence segmentation task demonstrates a sufficient amount of redundancy since ends of sentences are typically marked with lexical and prosodic cues.

Some studies also consider using different classification algorithms instead of different views for co-training. For example, [12] employs maximum entropy and hidden Markov models (HMMs) for part-of-speech tagging and parsing.

3. Approach

We first briefly present our sentence segmentation approach using lexical and prosodic features. Then, we present how we employ the co-training algorithm for this task using various example selection mechanisms. We also provide a description of the self-training method commonly used for semi-supervised learning.

Obtain a small set of L of labeled examples
Obtain a large set U of unlabeled examples
Obtain two sets F_1 and F_2 of features that are redundantly sufficient

while U is not empty **do**
  Learn classifier C_1 from L based on F_1
  Learn classifier C_2 from L based on F_2
  for each classifier C_i **do**
    C_i labels examples from U based on F_i
    C_i chooses the most confidently predicted examples E from U
    E is removed from U and added (with their given labels) to L
  end for
end while

Figure 1: *Basic co-training algorithm*

3.1. Sentence Segmentation

In this study, we consider sentence segmentation as a binary classification problem. For each word boundary, a probability is emitted by a statistical classifier, namely, Boosting [10]. If the probability is higher than a given threshold, a period is inserted at the word boundary. Prosodic and lexical features are used to represent word boundaries to the classifier. The 6 lexical features are N-grams composed of the word following the boundary of interest and the two previous words. The 34 prosodic features are the pause duration between the two words at the word boundary of interest, and various measures of the pitch and the energy of the voice of the speaker. The features are designed so that they measure the value of the pitch or energy before and after the word boundary of interest and their difference or comparison [11]. The range in which the value is measured is either the word or the window before/after the word boundary, and the measure considers the maximum, the minimum or the average value in this range. Some features are also normalized by speaker.

3.2. Co-Training

In this study we use an extended version of the basic co-training algorithm [12]. We use prosodic and lexical information as two separate views for the sentence segmentation task. Our co-training approach consists of multiple stages. In the first stage, we train two separate models using only prosodic and only lexical features. Then we estimate the sentence boundaries for the unlabeled portion of the data using these models. The examples are sorted according to their confidence scores. At this point, we tried different example selection mechanisms for co-training:

- **Agreement**: In this strategy, we consider only the examples that get high confidence scores according to both prosodic and lexical models. We add these examples to the training set of individual models and iterate.

Table 1. *Co-training performance figures with different strategies compared with self-training and baseline when only 1,000 manually labeled examples are available* (*F-Measure* (%))

1K	*Baseline*		*Agreement*		*Disagreement*		*Self-training*	
	Lex	*Pros*	*Lex*	*Pros*	*Lex*	*Pros*	*Lex*	*Pros*
M1	63.65	58.35	68.31	64.15	69.52	66.03	64.68	58.34
M2	45.96	57.24	69.28	63.07	69.07	64.95	45.85	58.03
M3	55.40	59.02	70.59	64.17	70.17	64.67	57.84	58.65
Avg.	55.00	58.20	69.39	63.79	69.58	65.21	56.12	58.34

Table 2. *Co-training performance figures with different strategies compared with self-training and baseline when only 1,000 manually labeled examples are available* (*NIST error rates* (%))

1K	*Baseline*		*Agreement*		*Disagreement*		*Self-training*	
	Lex	*Pros*	*Lex*	*Pros*	*Lex*	*Pros*	*Lex*	*Pros*
M1	62.41	70.32	64.63	62.61	61.94	59.66	63.09	70.56
M2	75.31	77.32	60.26	64.53	57.03	58.35	74.03	75.68
M3	69.49	69.72	61.04	60.09	58.71	61.84	66.01	69.90
Avg.	69.07	72.45	61.97	62.41	59.22	59.95	67.71	72.04

- **Disagreement**: In this strategy, we consider only the examples that are labeled with high confidence scores using one model and low confidence scores using the other one. We add these examples to the training set of the other model. The motivation here is to incorporate new examples that are hard to classify to the other model.

This process may be iterated until the models do not improve any more using a held-out set. After that, one can train a single model using both the lexical and prosodic features of the automatically and manually labeled examples in order to combine the models.

3.3. Self-Training

To compare performance of the co-training we also used the well-known self-training semi-supervised training for this task. For self-training, the given model estimates the sentence boundaries for the unlabeled portion of the data. Then the examples that are classified with high confidence scores are added to the training set, the model is retrained, and the whole process is iterated. To be compatible with the co-training experiments, instead of using self-training with all the features, we used it for the individual prosodic and lexical models.

4. Experiments and Results

All experiments were performed using manual transcriptions to avoid the noise introduced by the speech recognition system. The prosodic features were computed using the forced alignments of the manual transcriptions. We performed experiments using different sizes of initial manually labeled data and using different co-training methods. We compared performance using self-training and the baseline without any semi-supervised learning.

4.1. Data Sets

In our experiments we use the lexical and prosodic features of the ICSI Meeting Recorder Dialog Act (MRDA) Corpus. We use 51 meetings that have in total 538,956 examples with prosodic and lexical features, as training data. We use three different random orderings of the training set, namely M1, M2, and M3 in order to get different feature distributions and remove the biasing effect in the evaluation stage. In addition to this, both the development and test sets consist of 11 meetings and they have 110,851 and 101,510 examples, respectively. The test and development sets are kept the same for all the experiments. All the experiments are repeated for M1, M2, and M3, and their average is plotted.

4.2. Evaluation Metrics

For sentence segmentation, performance of the baseline and the co-training method is evaluated by the F-Measure and the NIST error rate. The F-Measure, which is often used in information retrieval and natural language processing, is the weighted harmonic mean of the precision and recall measures for the classes hypothesized by the classifier to the ones assigned by human labelers. The NIST error rate is the ratio of the number of insertion and deletion errors for sentence boundaries made by the classifier to the number of reference sentence boundary classes.

4.3. Experimental Results

Figure 2 presents the results using co-training with the agreement strategy. The curves show the performance improvement on the individual prosodic and lexical models when different sizes of initial manually labeled data are used. For each point in this plot, we report the average of three experiments performed with M1, M2, and M3, when the optimum number of unlabeled examples for the development set is added to the initial training set. This figure shows that the co-training process improves the results of the baseline significantly, especially when a lesser amount of labeled data is available. With only 1,000 manually labeled examples, the performance of the lexical model increases from 55% to 69%, an improvement of 25% relative.

Tables 1 and 2 provide complete results using different strategies of co-training and self-training when only 1,000 manually labeled examples are available. As seen, the disagreement strategy is slightly better than the agreement strategy. This behavior can be explained by reasoning that if both models are confident about an example it is relatively less informative. It is impressive that co-training strategies significantly outperform self-training, which provides slight improvement over the baseline.

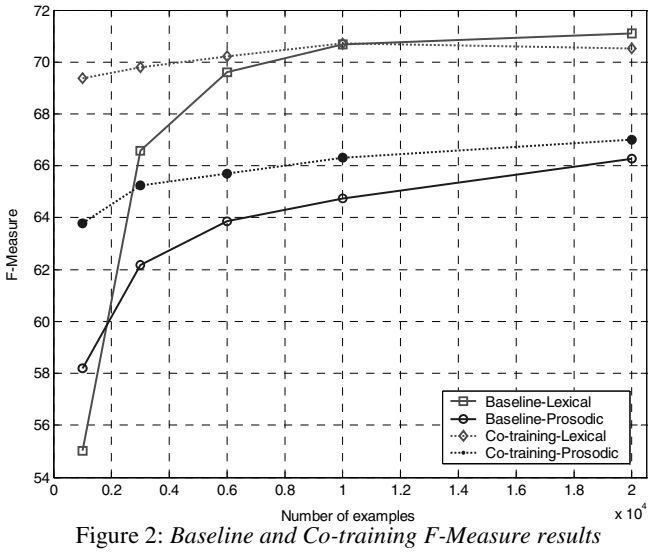

Figure 2: *Baseline and Co-training F-Measure results*

Table 3. *Co-training performance figures when both lexical and prosodic features are simultaneously used for modeling*

1K	*Baseline*		*Co-training*	
	F-Measure (%)	*NIST* (%)	*F-Measure* (%)	*NIST* (%)
M1	70.34	52.52	71.52	51.17
M2	70.15	51.45	70.30	51.45
M3	69.63	53.15	71.91	50.23
Avg.	70.04	52.37	71.24	50.95

Next we perform experiments by comparing a baseline model that uses all the features extracted from 1,000 examples and a model trained also with selected examples via co-training. To the best of our knowledge, co-training studies using different feature sets do not attempt to provide this sort of comparison with all the features simultaneously used for modeling. The typical practice is combining multiple classifiers at the score level [1,6,among others]. Table 3 presents these results. The semi-supervised co-training method results in modest but consistent improvements over the baseline for all three experiments and the average F-Measure is improved from 70.04 to 71.24 %.

5. Conclusions

We have investigated the application of the co-training learning algorithm on the sentence boundary classification problem by using lexical and prosodic information. The experimental results on the ICSI MRDA corpus show the effectiveness of the co-training algorithm for the task of sentence segmentation. Performance of the lexical and prosodic models is improved by 25% and 12% relative, respectively, when only a small set of manually labeled examples are used. When both information sources are combined, the semi-supervised co-training method results in modest but consistent improvements over the baseline.

Our future work includes employing cross-adaptation methods instead of simply concatenating the data to improve performance. The classifiers trained with lexical and prosodic features can be treated as a committee of classifiers, and can be used for committee-based active learning. We also plan to investigate the application of committee-based active learning for this task and combine with co-training. Furthermore, we plan to experiment with speech recognition output.

6. Acknowledgments

This work was partly supported by the Scientific and Technological Research Council of Turkey (TUBITAK), the Fulbright Scholar Program, and the Swiss National Science Foundation through the research network IM2 and Defense Advanced Research Projects Agency (DARPA) GALE (HR0011-06-C-0023) and CALO (NBCHD-030010) funding at ICSI and SRI, respectively. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funders.

7. References

[1] Blum, A., Mitchell, T. M., "Combining labeled and unlabeled data with co-training", in Proceedings of the Conference on Computational Learning Theory, pp.92-100, 1998.

[2] Mitchell, T. M., "The role of unlabeled data in supervised learning", in Proceedings of the Sixth International Colloquium on Cognitive Science, 1999.

[3] Abney, S., "Bootstrapping", in Proceedings of ACL, 2002.

[4] Kiritchenko, S., Matwin, S., "Email classification with co-training", in Proceedings of CASCON, 2001.

[5] Nigam, K., Ghani, R., "Understanding the behavior of co-training", in Proceedings of KDD-2000, Workshop on Text Mining, 2000.

[6] Nigam, K., Ghani, R., "Analyzing the effectiveness and applicability of co-training", in Proceedings of the ACM International Conference on Information and Knowledge Management, pp. 86-93, 2000.

[7] Mrozinski, J., Whittaker, E. W. D., Chatain, P., Furui, S., "Automatic sentence segmentation of speech for automatic summarization", in Proceedings of ICASSP, 2005.

[8] Shriberg, E., Stolcke, A., Hakkani-Tür, D., Tur, G., "Prosody-based automatic segmentation of speech into sentences and topics", Journal of Speech Communication, special issue on Accessing Information in Spoken Audio, vol. 32, no. 1-2, September, 2000.

[9] Cuendet, S., Hakkani-Tür, D., Tur, G., "Model adaptation for sentence segmentation from speech", in Proceedings of IEEE/ACL SLT Workshop, 2006.

[10] Schapire, R. E., Singer, Y., "Boostexter: A Boosting based system for text categorization", Machine Learning, vol. 39, no. 2/3, pp. 135–168, 2000.

[11] Stolcke, A., Shriberg, E., "Automatic linguistic segmentation of conversational speech", in Proceedings of ICSLP, 1996.

[12] Wang, W., Huang, Z., Harper, M., "Semi-supervised learning for part-of-speech tagging of Mandarin transcribed speech", in Proceedings of ICASSP, 2007.