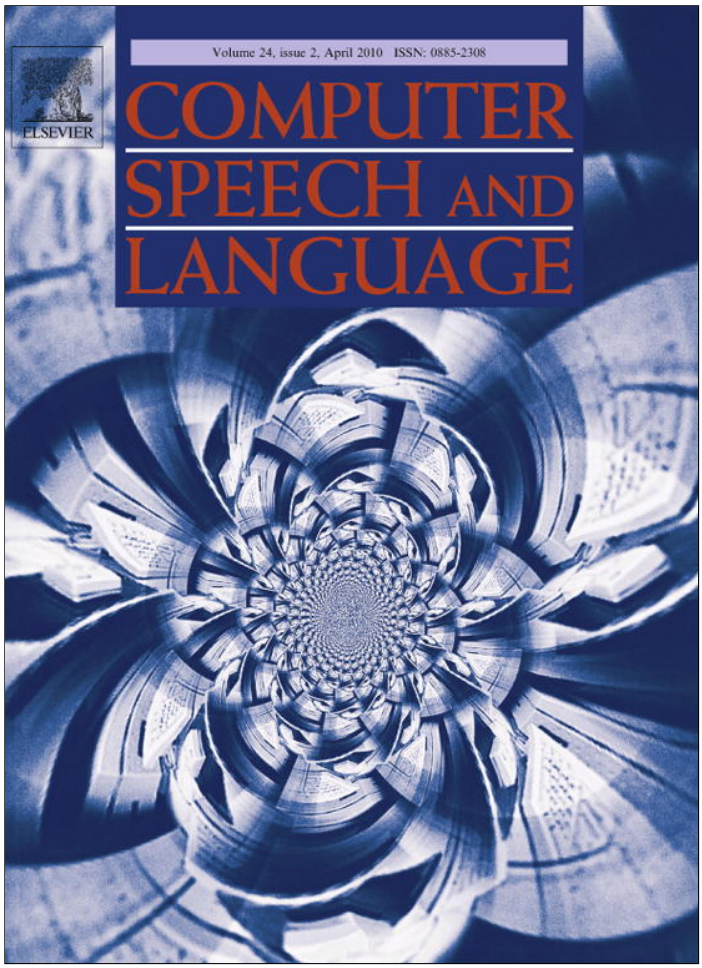

Available online at www.sciencedirect.com
ScienceDirect
Computer Speech and Language 24 (2010) 289–306
COMPUTER SPEECH AND LANGUAGE
www.elsevier.com/locate/csl

Cascaded model adaptation for dialog act segmentation and tagging

Umit Guz [a,c], Gokhan Tur [b,*], Dilek Hakkani-Tür [a], Sébastien Cuendet [a,1]

[a] International Computer Science Institute (ICSI), Speech Group, Berkeley, CA 94704, USA
[b] SRI International, Speech Technology and Research (STAR) Laboratory, Menlo Park, CA 94025, USA
[c] Isik University, Engineering Faculty, Department of Electronics Engineering, Istanbul, Turkey

Received 15 September 2008; received in revised form 3 March 2009; accepted 29 April 2009
Available online 15 May 2009

Abstract

There are many speech and language processing problems which require cascaded classification tasks. While model adaptation has been shown to be useful in isolated speech and language processing tasks, it is not clear what constitutes system adaptation for such complex systems. This paper studies the following questions: In cases where a sequence of classification tasks is employed, how important is to adapt the earlier or latter systems? Is the performance improvement obtained in the earlier stages via adaptation carried on to later stages in cases where the later stages perform adaptation using similar data and/or methods? In this study, as part of a larger scale multiparty meeting understanding system, we analyze various methods for adapting dialog act segmentation and tagging models trained on conversational telephone speech (CTS) to meeting style conversations. We investigate the effect of using adapted and unadapted models for dialog act segmentation with those of tagging, showing the effect of model adaptation for cascaded classification tasks. Our results indicate that we can achieve significantly better dialog act segmentation and tagging by adapting the out-of-domain models, especially when the amount of in-domain data is limited. Experimental results show that it is more effective to adapt the models in the latter classification tasks, in our case dialog act tagging, when dealing with a sequence of cascaded classification tasks.

Keywords: Model adaptation; Dialog act segmentation; Dialog act tagging; Meetings processing

1. Introduction

Recent advances in data-driven speech and language processing techniques combined with discriminative machine learning algorithms, such as support vector machines or Boosting, enable us to build

* Corresponding author. Tel.: +1 973 359 9939.
E-mail addresses: guz@icsi.berkeley.edu (U. Guz), gokhan@speech.sri.com (G. Tur), dilek@icsi.berkeley.edu (D. Hakkani-Tür), cuendet@icsi.berkeley.edu (S. Cuendet).
[1] Present address: Optaros, Zurich, Switzerland. This work has been done while the author has been at ICSI on leave from the Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland.

0885-2308/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.csl.2009.04.006

high-performance, robust, portable, statistical models given enough in-domain annotated data. However, such classifiers typically require large amounts of in-domain task data that is usually transcribed and then labeled by humans, an expensive and laborious process. To this end, in the literature, model adaptation techniques have been proposed. The aim of adaptation is to exploit the existing labeled data from previous corpora (*out-of-domain*), to improve the performance of the model for a new domain (*in-domain*). The amount of in-domain labeled data is typically much smaller than the amount of out-of-domain labeled data. The idea is to build a new adapted model by using the existing out-of-domain model together with the small amount of labeled in-domain data from the new application. However, the in-domain data and the out-of-domain data do not usually share the same distribution, and the out-of-domain classification model must be adapted before it can be employed. There are two main advantages of using adapted models for new in-domain data:

- Reduction of the amount of human-labeling effort necessary to come up with decent statistical systems.
- Improvement of the classification performance for a given amount of in-domain data.

The state of the art speech and language processing systems consist of a series of non-trivial processing stages. For example, in a speech-to-speech translation system, the utterances are first recognized, then chopped into sentences, translated individually, and then converted to speech using a synthesizer. One can arbitrarily build even more complex systems, for example, by incorporating a syntactic or semantic parser, a named entity extractor, or a dialog act tagger in this chain of subtasks. Another example would be call classification in which the input utterances are first recognized and then named entities are extracted and intents are determined (Gupta et al., 2006).

While model adaptation has been shown to be useful in isolated speech and language processing tasks, such as acoustic modeling (AM) and language modeling (LM), and probabilistic context-free grammars (Bellegarda, 2004), it is not clear what constitutes system adaptation for such complex systems. One can, of course, adapt each of the models individually, and wish that the performance of the whole system improves.

This paper then studies the following questions: In cases where a sequence of classification tasks is employed, how important is to adapt the earlier or latter systems? Is the performance improvement obtained in the earlier stages via adaptation carried on to later stages in cases where the later stages perform adaptation using similar data and/or methods?

Furthermore, it is known that the effect of adaptation disappears as more in-domain annotated data is provided for training models. For cascaded speech and language processing tasks, it is not clear whether the effect of cascaded adaptation disappears with similar pace. This is because, even though working with manual annotations for previous steps (e.g. manual transcriptions of speech), the following step (e.g. machine translation) is not error-free and the relative ratio of performance improvement in previous steps due to adaptation may not hold for the final system.

There are many speech and language processing problems which require such cascaded classification tasks. In our case, dialog act segmentation and tagging tasks provide a good framework to perform research on the effect of adaptation on cascaded tasks. Dialog acts (DAs) are basic building blocks for spoken language understanding in human/human conversations or multiparty meetings. DA segmentation aims to chop an input utterance into separate dialog act and sentential units. Then DA tagging is used to classify each of these units into a predefined DA category, such as question, floor grabber, or backchannel. We first analyze the effect of various model adaptation techniques for dialog act segmentation and tagging tasks. Then we perform controlled experiments to see the effect of cascaded adaptation with various portions of annotated data for each task.

This paper is organized as follows: The next section briefly explains the dialog act segmentation and tagging tasks and previous work on this area. Then we present the related model adaptation work in speech and language processing. Section 4 presents our modeling method and the adaptation approaches that we used in this work. In Section 5, we give more details on the data sets and present the results. The conclusions, as well as the outlook for future work, are discussed in Section 6.

2. Dialog act segmentation and tagging

Dialog acts are basic building blocks for spoken language understanding in human/human conversations or multiparty meetings. A dialog act is an approximate representation of the illocutionary force of an utterance, such as questions or backchannels (Stolcke et al., 2000). Dialog acts are designed to be task independent, their main goal being to provide a basis for further discourse analysis and understanding. For example, dialog acts can be used to extract the question/answer pairs in a meeting. There are a number of predefined dialog act sets in the literature, such as dialog act markup in several layers (DAMSL; Core and Allen, 1997) and meeting recorder dialog act (MRDA; Shriberg et al., 2004).

In this study we used the ICSI meeting corpus with high-level MRDA tags: *question*, *statement*, *backchannel*, *disruptions*, and *floor grabbers/holders*. Backchannels are short phrases such as *yeah* or *uh huh* to indicate that the listener is actually following the speaker. Floor grabbers indicate that the person wants to start talking; similarly, floor holders indicate that the speaker has not yet finished. Disruptions stand for statements uncompleted for some reason. Note that dialog acts can be organized in a hierarchical fashion. For instance, statements can be further categorized as *explanation* or *suggestion*. Fig. 1 shows an example of a dialog along with dialog acts.

Typically, dialog act tagging is performed on the automatic speech recognition (ASR) output. Since most ASR outputs lack typographic cues such as sentence and paragraph boundaries, an intermediate segmentation step is necessary. This task, known as *sentence unit segmentation* or *dialog act segmentation*, aims at deciding whether a particular word boundary marks the end of a dialog act unit. Typically, the speech is first automatically transcribed by the ASR and then segmented into dialog acts, and finally the categories or dialog act tags are assigned to these segments during the dialog act tagging (DAT) phase.

Dialog act segmentation is a crucial first step in processing conversational speech such as meetings (as in CALO Tur et al., 2008) or broadcast conversations (as in GALE Zimmerman et al., 2006). Dialog act segmentation is generally framed as a word boundary classification problem. For DA segmentation, Gotoh and Renals (2000) and Shriberg et al. (2000) used a method that combines hidden Markov models (HMMs) with N-gram language models (LMs) containing words and dialog act boundaries associated with them, i.e., tags (Stolcke and Shriberg, 1996). This method was extended with confusion networks in Hillard et al. (2004). Zimmerman et al. (2006) provides an overview of different classification algorithms (Boosting, hidden-event language model, maximum entropy and decision trees) applied to the dialog act segmentation for multilingual broadcast news. Besides the type of classifier, the features have widely been studied; Shriberg et al. (2000), Liu et al. (2005) showed how prosodic features can benefit the dialog act segmentation task. Investigations on prosodic and lexical features in the context of phone conversation and broadcast news speech were presented in Liu et al. (2005). More recently, Roark et al. (2006) studied syntactic features for this task.

Generally, lexical features, such as word N-grams as well as dialog contextual features (such as the previous dialog act tag) or other types of features (such as prosodic) are employed. Recently, there has been interest in training a joint classifier that segments and tags the utterances simultaneously (Zimmermann et al., 2005).

Even though the dialog acts are designed to be task independent and we consider only five top-level dialog acts, there are significant differences between different corpora because of different dialog act distributions and

- *John Smith*: so we need to arrange an office for joe browning (statement)
- *Kathy Brown*: are there special requirements (question)
- *Cindy Green*: when is he co- (disruption)
- *John Smith*: yes (affirmation) // there are (statement)
- *John Smith*: we want him to be close to the image processing guys
- *Kathy Brown*: okay (agreement) // I'll talk to the secretary (commitment)
- *Cindy Green*: hold on (floor grabber) // wh- when is he coming (question)
- *John Smith*: next monday (statement)
- *Cindy Green*: uh-huh (backchannel)

Fig. 1. Sample excerpt of a dialog along with dialog acts.

labeling inconsistencies. The differences between the corpora are particularly emphasized in speech since speech has a high variability depending on the environment in which it is uttered (e.g. number of speakers, formal/informal context), which we designate as *speech style* (Shriberg, 2005). Adaptation is thus particularly needed for speech data.

Previously, Venkataraman et al. tried employing active learning and lightly supervised learning for reducing the amount of labeled data needed for dialog act tagging with HMMs. They concluded that while active learning does not help significantly for this task, exploiting unlabeled data by using minimal supervision is effective when the DA tag sequence is also modeled (Venkataraman et al., 2002; Venkataraman et al., 2005). Note that in this study, we consider only supervised model adaptation and analyze the effect of adaptation in a controlled setting where the in-domain data is from the ICSI meeting corpus (referred to as MRDA), and out-of-domain data is the Switchboard (SWBD) corpus with the SWBD–DAMSL tag set (Jurafsky et al., 1997). Besides the difference in speech style, another challenge is that in the SWBD corpus, floor grabbers/holders are not considered as a separate class; hence, there are only four top-level dialog acts instead of the five MRDA tags.

Although there is some recent work on joint modeling of dialog act segmentation and tagging, the benefit of using a joint modeling approach instead of a cascaded approach has not been clear. The cascaded approach is currently the established method since it provides a modular architecture and the output of sentence segmentation can be optimized for other tasks such as parsing or machine translation independently. However, as we mention in the last section, it would be a further study to compare our results with adaptation performed on joint modeling.

3. Related work on model adaptation

In speech processing literature, two very popular adaptation approaches are maximum likelihood linear regression (MLLR; Gales, 1998) and maximum *a posteriori* (MAP) adaptation (Gauvain and Lee, 1994). For LM adaptation, MAP adaptation leads to a solution of the form Bellegarda (2004):

$$P_{\hat{\theta}}(w_i|h_i) = \frac{\alpha C^{(o)}(h_i w_i) + \beta C^{(i)}(h_i w_i)}{\alpha C^{(o)}(h_i) + \beta C^{(i)}(h_i)}, \tag{1}$$

where (i) stands for in-domain, (o) stands for out-of-domain, and $C_D(S)$ is the frequency count of the string S in D. α and β are the weights controlling the influence of the data sets on the final model and is usually optimized on a development set or via an expectation maximization (EM) algorithm.

Bacchiani and Roark (2003), Bacchiani et al. (2006) have shown that MAP adaptation for LM is equivalent to model interpolation or count mixing with a different parameterization of the prior distribution:

$$P_{\hat{\theta}}(w_i|h_i) = \gamma P_{\theta^{(o)}}(w_i|h_i) + (1-\gamma) P_{\theta^{(i)}}(w_i|h_i) \tag{2}$$

where $P_\theta(w_i|h_i)$ is the probability of the current word w_i given the history h_i of $n-1$ words, in an N-gram LM θ. $\gamma[0,1]$ is a weight controlling the influence of the out-of-domain data on the final model.

In the above-mentioned work, Bacchiani et al. (2004) reported positive results using unsupervised LM adaptation in a voicemail recognition system. They noted that there is no significant difference in performance between count merging and interpolation. Later, they also employed discriminative methods for language model adaptation using the perceptron algorithm. Kneser et al. (1997) have proposed using dynamic marginals for model adaptation. The idea is to adjust the N-gram weights so that the unigram marginals of the adapted N-grams match the unigram distribution of the adaptation data. Gretter and Riccardi (2001) have exploited word confidences obtained from word confusion networks during unsupervised LM adaptation. Hakkani-Tür et al. (2004) have employed unsupervised LM adaptation for new call center spoken dialog applications. One difference in their approach is that they effectively set $\gamma = 0$ in order to restrict the vocabulary size and hence make the new model small enough for a sub-real-time ASR system. Previous work on conversational telephone speech recognition showed small gains with unsupervised LM adaptation to Switchboard recognition output even at fairly high error rates (Stolcke, 2001).

A research area related to unsupervised LM adaptation deals with strategies for selecting adaptation data. Some notable studies addressing this issue include (Chen et al., 2003; Nanjo and Kawahara, 2003). LM adaptation using linear interpolation and training data filtering was presented in Fang et al. (2003). An extensive survey of LM adaptation research can be found in Bellegarda (2004).

Language model adaptation for generative models focuses on the estimation of $P(w|h)$. The main difference between MAP and model interpolation is then performing the adaptation at the model level or count level. While the exact same formulation applies to generative classification models such as Naive Bayes or HMMs, model adaptation is a relatively less studied area for discriminative classifiers. In the machine learning literature, model adaptation techniques have been studied under the area of transfer learning. A more detailed explanation of inductive and transductive transfer learning algorithms can be found in Arnold et al. (2007). For discriminative classifiers, researchers have also suggested methods for learning new models starting from existing ones. For maximum entropy models, Chelba and Acero (2006) have proposed a variant of MAP adaptation, where the goal is maximizing the regularized log-likelihood of the adaptation data during training. For Boosting, Tur (2005) explored model adaptation via changing the loss function during training. This is explained in more detail below.

While this work considered one distribution for the in-domain data and one for the out-of-domain data, a recent study introduced the idea of learning one general distribution, and then using this in conjunction with the in-domain and out-of-domain data (Daumé and Marcu, 2006).

4. Approach

In this work, both dialog act segmentation and dialog act tagging are framed as classification problems following the earlier work (Zimmerman et al., 2006, 2000; Zimmermann et al., 2006 among others) and AdaBoost.MH algorithm[2] is used to this end. Boosting has already been shown to be among the best classifiers for the sentence segmentation task (Zimmerman et al., 2006).

For dialog act segmentation, a sample (that is, a word boundary b_i, between words w_i and w_{i+1}) is represented by features containing lexical information (combination of word unigrams, bigrams, and trigrams) and the pause duration between two words:

$$w_{i-1}, w_i, w_{i+1}, w_{i-1}w_i, w_iw_{i+1}, w_{i-1}w_iw_{i+1}, w_ip_i, w_ip_iw_{i+1}$$

where p_i represents the binned pause duration between words w_i and w_{i+1}.

Once the dialog act boundaries are hypothesized, the dialog act tags are assigned. The goal of DA tagging is classifying the given sentence into one of the five top-level dialog acts (*question*, *statement*, *backchannel*, *disruptions*, and *floor grabbers/holders*). In order to see the impact of DA segmentation adaptation for DA tagging better, we deliberately kept the DA tagger simpler and did not use any prosodic or dialog contextual features and just relied on the word *n*-grams as extracted from the current sentence.

Next, we briefly describe the Boosting algorithm, then we present the model adaptation methods employed in this study, and then present the cascaded adaptation scheme.

4.1. Boosting

In this study, we employed a discriminative classifier, namely, Boosting. Boosting is an iterative learning algorithm that aims to combine weak base classifiers to come up with a strong classifier. At each iteration, a weak classifier is learned so as to minimize the training error, and a different distribution or weighting over the training examples is used to give more emphasis to examples that are often misclassified by preceding weak classifiers. In Boosting, weighted sampling is used instead of random sampling to focus learning on most difficult examples. Furthermore, weak classifiers are combined using weighted voting instead of equal voting.

[2] In this paper, we abusively use the term "Boosting" to designate the AdaBoost.MH algorithm.

Initialization:
1. Given training data from the instance space
$S = \{(x_1, Y_1), ..., (x_m, Y_m)\}$ where $x_i \in \mathcal{X}$ and $Y_i \subseteq \mathcal{Y}$.
2. Initialize the distribution $D_1(i,l) = \frac{1}{mk}$.
Algorithm:
for $t = 1, ..., T$: do
Train a weak learner $h_t : \mathcal{X} \to \mathbb{R}$ using distribution D_t.
Determine weight α_t of h_t.
Update the distribution over the training set:

$$D_{t+1}(i,l) = \frac{D_t(i,l)e^{-\alpha_t Y_i[l]h_t(x_i,l)}}{Z_t}$$

where Z_t is a normalization factor chosen so that D_{t+1} will be a distribution.
end for
Final score:
$f(x,l) = \sum_{t=1}^{T} \alpha_t h_t(x,l)$

Fig. 2. The algorithm AdaBoost.MH.

The algorithm generalized for multi-class and multi-label classification is shown in Fig. 2. Let $\mathcal{X}$ denote the domain of possible training examples and let $\mathcal{Y}$ be a finite set of classes of size $|\mathcal{Y}| = k$. For $Y \subseteq \mathcal{Y}$, let $Y[l]$ for $l \in \mathcal{Y}$ be

$$Y[l] = \begin{cases} +1, & \text{if } l \in Y \\ -1, & \text{otherwise} \end{cases}$$

Boosting assumes that the data set D consists of the training instances or examples x_i. Each example x_i in D is represented by a set of features (e.g. lexical features). Each example has also the classes (labels) $l_i \in Y$, which are assigned by human labelers. For example, the sentence segmentation task can be considered as a binary classification problem, in which every word boundary must be labeled as a sentence boundary or as a nonsentence boundary. In this manner, the set of possible classes $Y = \{+1, -1\}$ is also referred to true or reference classes, where $+1$ and -1 represent the sentence and the nonsentence boundaries, respectively.

In Boosting, every example of the training data set is assigned a weight. These weights are initialized uniformly and updated on each iteration so that the algorithm focuses on the examples that were wrongly classified on the previous iteration. At the end of the learning process, the weak learners used on each iteration t are linearly combined to form the classification function:

$$f(x,l) = \sum_{t=1}^{T} \alpha_t h_t(x,l) \tag{3}$$

with α_t the weight of the weak learner h_t and T the number of iterations of the algorithm. This algorithm can be seen as a procedure for finding a linear combination of base classifiers that attempts to minimize a loss function (Schapire and Singer, 1999), such as the logistic loss:

$$\sum_i \sum_l \ln\left(1 + e^{-Y_i[l]f(x_i,l)}\right) \tag{4}$$

In that case, the confidence score of a class, l, for an example x_i can be computed as

$$P(Y_i[l] = +1 | x_i) = \frac{1}{1 + e^{-f(x_i,l)}} \tag{5}$$

More details on Boosting can be found in Schapire (2001). In this study, we used decision stumps (single node decision trees) as the weak learners. Since our features are word or word/pause n-grams for DA tagging and segmentation, each weak learner checks the existence of a specific n-gram.

4.2. Adaptation methods

The goal of this work is to use the existing labeled data or models to improve the classification performance in a new domain. The combination of the in-domain and the out-of-domain data sets can be implemented at different levels, such as the data level (e.g. concatenation), the feature or classifier level (e.g. Boosting adaptation) and the classifier output level (e.g. linear or logistic interpolation). It should be noted that, these different model adaptation approaches are equivalent under certain conditions. For example while one can interpolate the outputs obtained by two models, the same effect can be represented in a single interpolated model, as typically done in language models (Stolcke, 2002). Similarly, data concatenation can be seen as an unweighted linear interpolation in certain cases.

In a typical classification problem, given a set of training data $D = \{(x_n, l_n) \in \mathscr{X} \times \mathscr{L} : 1 \leqslant n \leqslant N\}$, the goal is to find a function $f : \mathscr{X} \rightarrow \mathscr{L}$, where $\mathscr{X}$ is the feature space, $\mathscr{L}$ is the finite set of possible labels, N is the number of training examples x_n and their associated label l_n. The underlying assumption is that a distribution $p(x_i, l_i)$ exists for each $(x_i, l_i) \in \mathscr{X} \times \mathscr{L}$, but is unknown. In the adaptation problem, we assume two data sets, the in-domain (or task specific domain) $D^{(i)}$ and the out-of-domain $D^{(o)}$ data sets, with $|D^{(i)}| \ll |D^{(o)}|$. The goal is to find a function $f(x)$ that can predict the classification label l for each example in $D^{(i)}$ by using the rest of $D^{(i)}$ and $D^{(o)}$. This makes clear that we assume the distributions $P^{(i)}(x_i, l_i)$ and $P^{(o)}(x_i, l_i)$ of the in-domain data and the out-of-domain data, respectively, not to be dependent, in which case $D^{(o)}$ would be suboptimal for classifying $D^{(i)}$.

Adaptation methods that we used for dialog act segmentation and tagging are briefly explained below. These methods are independent of the classifier, except for adaptation with Boosting.

4.2.1. Data concatenation

The simplest way of combination is to train the classifier on the concatenation of out-of-domain and in-domain data. Due to the relative size of the out-of-domain data overwhelms the in-domain data, it is not fair to expect the best performance from this kind of elementary approach. But, basically, it can be considered as a baseline for the other adaptation approach that we used. This is equivalent to unweighted count mixing for LM adaptation. One can also upsample the labeled data with different ratios to approximate the effect of interpolation. The schematic representation of this method is shown in Fig. 3.

More formally, with this method, Boosting chooses a weak learner (i.e., an n-gram, $h_i w_i$) based on its frequency in $D^{(i)} \bigcup D^{(o)}$:

$$\frac{C^{(o)}(h_i w_i) + C^{(i)}(h_i w_i)}{C^{(o)}(h_i) + C^{(i)}(h_i)} \tag{6}$$

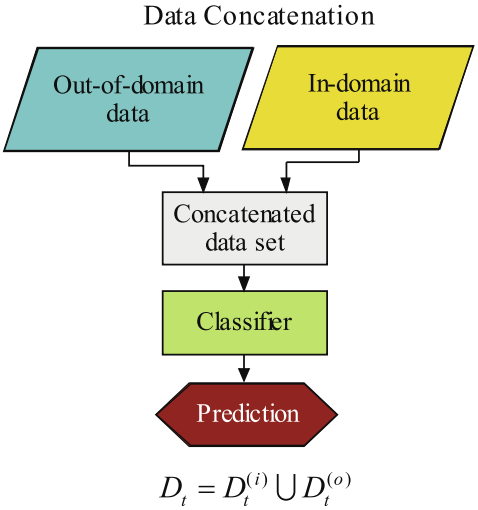

Fig. 3. Schematic representation of the data concatenation method.

4.2.2. Out-of-domain as feature

For this method, the existing classifier, $C^{(o)}$, is run first; the probability that it outputs is then used as an extra feature while training a model with the in-domain data. The final decision is made by $C^{(i)}$ trained on this enriched set of features. The schematic representation of this method is shown in Fig. 4. In other words, the feature set, F, is extended by one important feature, that is the posterior probability distribution for each of the classes, $P^{(o)}(Y_i[l] = +1|x_i)$.

4.2.3. Model interpolation

More formally, in *model interpolation*, each sample of the held-out set is evaluated by the classifier $C^{(i)}$ trained on the in-domain data and the classifier $C^{(o)}$ trained on the out-of-domain data. These evaluations yield probabilities $P^{(i)}(Y_i[l] = +1|x_i)$ and $P^{(o)}(Y_i[l] = +1|x_i)$ that the event associated with the sample x_i is belongs to class l, according to the classifier $C^{(i)}$ and $C^{(o)}$, respectively. The final decision is made from the combination of these two probabilities using either a linear function:

$$P(Y_i[l] = +1|x_i) = -b_1 - b_2 P^{(i)}(Y_i[l] = +1|x_i) - b_3 P^{(o)}(Y_i[l] = +1|x_i) \tag{7}$$

or a logistic function:

$$P(Y_i[l] = +1|x_i) = \frac{1}{1 + e^{\left(-b_1 - b_2 P^{(i)}(Y_i[l]=+1|x_i) - b_3 P^{(o)}(Y_i[l]=+1|x_i)\right)}} \tag{8}$$

where b_1, b_2, b_3 are called as interpolation weights.

The schematic representation of this method is shown in Fig. 5. One important problem for model interpolation is the estimation of these interpolation weights. While two popular approaches in the literature are employing an EM algorithm or using a held-out set and empirically optimizing it, in this study, we propose using regression techniques with a held-out set.

For implementing model adaptation via interpolation, we have employed the classifier output confidences. In Boosting, the classifier score can be converted into a confidence using the logistic function (Schapire and Singer, 2000):

$$P(c = l|x) = \frac{1}{1 + e^{-f(x,l)}} \tag{9}$$

4.2.4. Boosting adaptation

Using the same approach as in Tur (2005) a model is first built with the out-of domain data and then using Boosting adapted to small amount of in-domain labeled data. This is the same as minimizing a weighted sum of logistic loss function and the binary relative entropy of the prior probabilities of both models. The weights are optimized using a held-out set.

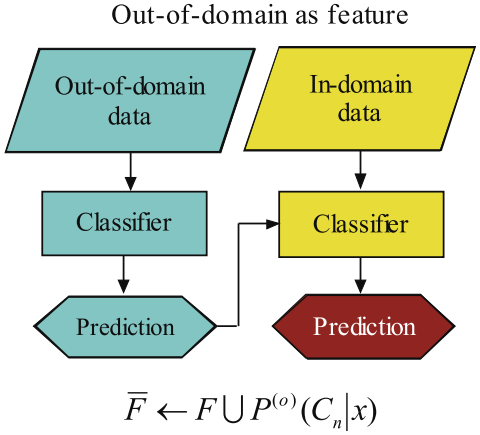

Fig. 4. Schematic representation of the "out-of-domain as feature" adaptation method.

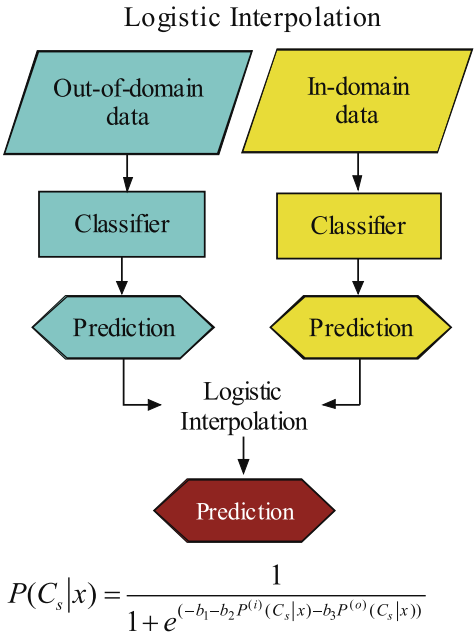

Fig. 5. Schematic representation of the logistic interpolation method.

For Boosting adaptation, we begin with an existing out-of-domain model. Then we build a new model using the small amount of labeled in-domain data based on the existing out-of-domain model. The schematic representation of this method is shown in Fig. 6. This method is similar to incorporating prior knowledge or exploiting unlabeled utterances for Boosting Schapire et al. (2005) and Tur and Hakkani-Tür (2003). In those works, a model which fits both the training data and the task knowledge or machine labeled data is trained. In our case, the aim is to train a model that fits both a small amount of application-specific labeled (in-domain) data and the existing out-of-domain model from a similar application. More formally the Boosting algorithm tries to fit both the newly labeled data and the prior model using the following loss function:

$$\sum_i \sum_l \left[\ln\left(1+\mathrm{e}^{-Y_i[l]f(x_i,l)}\right) + \eta KL(P(Y_i[l]=1|x_i)\|\rho(f(x_i,l)))\right] \tag{10}$$

where

$$KL(p\|q) = p\ln\left(\frac{p}{q}\right) + (1-p)\ln\left(\frac{1-p}{1-q}\right) \tag{11}$$

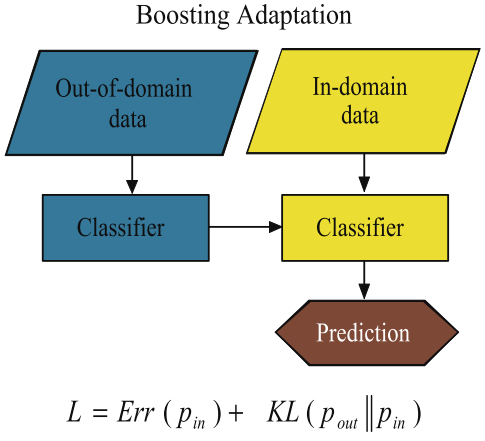

Fig. 6. Schematic representation of the Boosting adaptation method.

is the Kullback–Leibler divergence (or binary relative entropy) between two probability distributions p and q. In our case, the distributions correspond to the distribution from the prior model $KL(P(Y_i[l] = 1|x_i))$ and to the distribution from the constructed model $\rho(f(x_i, l))$, where $\rho(x)$ is the logistic function $1/(1 + e^{-x})$. This term is basically the distance from the existing out-of-domain model to the new model built with newly labeled data. In the marginal case, if these distributions are always the same, then the KL term will be zero and the loss function will be exactly the same as the first term, which is nothing but the logistic loss. here η is used to control the relative importance of these two terms. This weight may be determined empirically on a held-out set.

4.3. Cascaded model adaptation

In our earlier work we have presented the results of these model adaptation methods for dialog act segmentation (Cuendet et al., 2006) and tagging (Tur et al., 2006) independently. In this paper, we investigate their combination to check whether the improvements observed on the sentence segmentation task when using an adapted model reflect on dialog act tagging.

In particular, we aim at analyzing the effect of the adaptation methods for dialog act segmentation and tagging and testing whether the improvements observed on the dialog act segmentation task when using an adapted model reflect on dialog act tagging.

Our approach is as follows: An adapted model for sentence unit segmentation, $\hat{C}_{seg}$, is first built using one of the adaptation methods presented and the output of the model is used to get the hypothesized sentence boundaries on the *test* and *held-out* sets for dialog act tagging. An adapted dialog act tagging classifier $\hat{C}_{tag}$ is then trained and adapted using the training set of the in-domain and out-of-domain data with the true sentence boundaries. The classifier $\hat{C}_{tag}$ is used to classify the estimated sentence boundaries of the held-out set into the five dialog act classes. For the linear and logistic interpolation, the weights are optimized on the held-out set using regression as explained above and then used to classify the samples of the test set.

The expectation is that the dialog act tagging on the sentence boundaries hypothesized by the adapted model is more accurate than that performed on the sentence boundaries hypothesized by the in-domain model. Furthermore, the improvement is not washed away with adapted dialog act tagging models.

5. Experiments and results

The results are presented in three parts. We first show the results of sentence unit segmentation adaptation on the MRDA corpus. The second set of experiments presents dialog act tagging adaptation on the MRDA corpus. In the third part, we show dialog act tagging adaptation along with sentence unit segmentation, i.e., the dialog act tagging is done on the sentence boundaries hypothesized by the sentence unit segmentation classifier. For each experimental setting, we present learning curves to show the effect of the size of the in-domain data on the adaptation methods employed. Each experiment is run three times with three different orderings of the training data, and the three results are averaged. We performed our tests using the *BoostTexter* classification tool (Schapire and Singer, 2000) for Boosting. For all experiments, we used word n-grams and the duration of the pause between two words as features and iterated 1000 times.

5.1. Data sets

The meeting data is from the ICSI meeting corpus (MRDA) (Janin et al., 2004). This corpus contains 75 meetings grouped in three main types (according to the speakers, the conversation types, and so on). The data is split according to the training, test (11 meetings) and held-out (11 meetings), sets as specified in (Ang et al., 2005). The phone conversations are the subset of the Switchboard (SWBD) corpus provided by the linguistic data consortium (LDC) (RT04). The main characteristics of both data sets are shown in Table 1. In both corpora, *statements* are the most frequent dialog act category. In the SWBD corpus, *backchannels* are more than twice as frequent as in the meetings data, possibly because of the lack of eye contact in telephone conversations.

Table 1
Characteristics of the data used in the experiments.

	MRDA	SWBD
Training data size (utterances)	80577	64874
Test data size (utterances)	16211	N/A
Development data size (utterances)	16501	N/A
Average sentence length (words)	7.58	10.45
Vocabulary size	11,034	13,109
Questions	6%	4%
Disruptions	12%	6%
Floor grabbers/holders	10%	0%
Statements	58%	58%
Backchannels	12%	29%

All models were trained using reference transcriptions and tested on both the reference and the automatic transcriptions obtained using the ASR output (STT; Stolcke et al., 2005). The ASR word error rate on this set is 33.7% (18.1% substitutions, 13.7% deletions, and 1.9% insertions) and the classifier performance is thus expected to be worse on them than on the reference transcriptions.

5.2. Evaluation metrics

To measure the performance of dialog act segmentation, we used *F*-measure. *F*-measure is the harmonic mean of the recall and precision measures of the sentence boundaries hypothesized by the classifier to the ones assigned by human labelers.

$precision = \frac{t_p}{t_p+f_p}$ and $recall = \frac{t_p}{t_p+f_n}$ where f_n, f_p, and t_p are false negative, false positive, and true positive, respectively.

$$F\text{-measure} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{12}$$

In our experiments we consider the event associated with an example as a sentence boundary if the posterior probability $P(s\prime\prime|x)$ emitted by the classifier for the sample x is bigger than 0.5 (as optimized on the held-out set), and as a nonsentence boundary otherwise.

The performance of the dialog act tagging is evaluated by different metrics depending on whether or not the sentence boundaries are correct. If reference sentence boundaries are used, we simply compute the classification error rate (CER), which is defined as the ratio of the erroneous tags divided by the number of sentences. In the case of sentence boundaries output by the sentence segmentation system, we employ the previously defined metrics (Ang et al., 2005; Zimmermann et al., 2005), namely, *lenient*, *strict*, *dialog act error rate* (DER). DER is computed as the ratio of sentences with correct boundaries and correct tags divided by the number of sentences. This is the same as CER when the sentence boundaries are correct. Lenient and strict are word-based metrics. The lenient metric does not take into account the segmentation boundaries but only compares the DA types assigned to corresponding words. For the strict metric, a word is considered to be correctly classified if and only if it has been assigned the correct DA type and it lies in exactly the same DA segment as the corresponding word of the reference. Fig. 7, as presented in Zimmermann et al. (2005), demonstrates the computation of these metrics.

5.3. Sentence unit segmentation adaptation

The learning curves in Figs. 8 and 9 show the *F*-Measure for the SWBD adaptation on the MRDA corpus, in reference and STT conditions, respectively. As expected, the naive approach of data concatenation performs the worst. Logistic interpolation and "out-of-domain as feature" methods perform the best when very little meeting data is available. The logistic interpolation method becomes less effective as more in-domain data is available. The "out-of-domain as feature" is the method that performs the best independently from the size of the in-domain training data. There is a constant improvement of about 0.5% across the learning curve using this method. This is a statistically significant improvement according to a *Z*-test with 95% confidence range.

Reference	S|Q.Q.Q.Q|S.S.S|B|S.S|
System	S|Q|S|Q.Q|D.D.D|S.S|S|
Lenient	C C E C C E E E E C C
Strict	C E E E E E E E E E E
DER	C| E | E |E| E |

Metric	Errors	Reference	Error Rate
Lenient	5 match errors	11 words	45%
Strict	10 match errors	11 words	91%
DER	4 match errors	5 DAs	80%

Fig. 7. Illustration of metrics used in this study for dialog act segmentation and tagging. In this figure, S, Q, D and B are used to denote words in statements, questions, disruptions and back-channels, respectively. C and E are used to denote words of sentences that have correctly and incorrectly assigned dialog act tags, respectively.

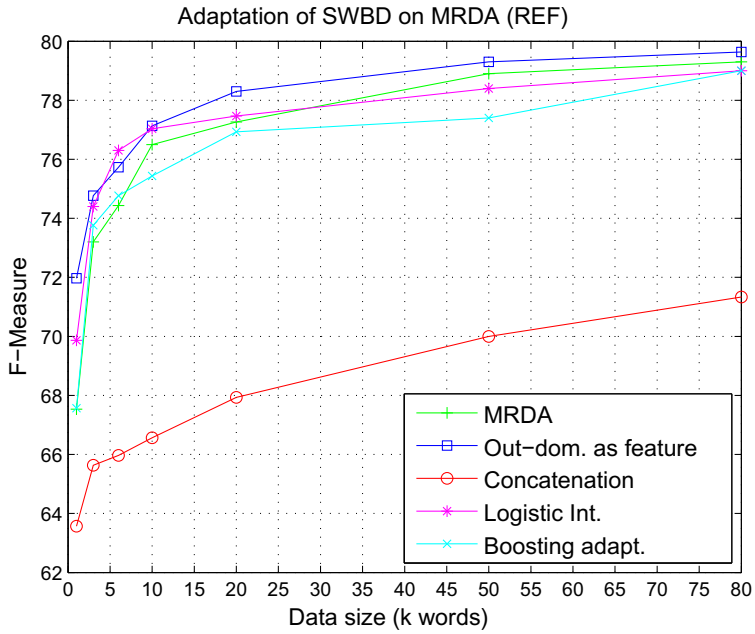

Fig. 8. *F*-measure for all the methods presented in Section 3, in reference conditions.

Both figures show that the more meeting data, the smaller the difference between the classifier trained on meetings (in-domain) only and using the adapted models. However, it should be noticed that even with the full meeting training data, all adaptation methods but the data concatenation perform better than the classifier built only on the meeting data.

The performance on the STT conditions shows the same pattern as on the reference conditions, although it is per se lower by 10–15%. The addition of the ASR error to the classification error can explain this difference. However, in both STT and reference conditions, using model adaptation one would need to label around 30–35% of the words of the meeting data to reach the same performance.

5.4. Dialog act tagging adaptation

The experiments presented in this section are done using manually transcribed and segmented data in order not to deal with automatic speech recognition and sentence unit segmentation noise. We did not use any dialog

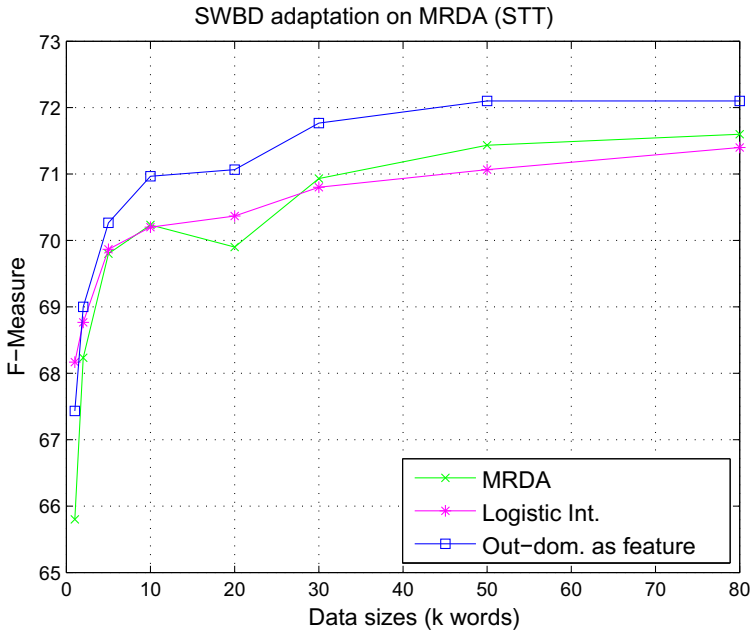

Fig. 9. *F*-measure for all the methods presented in Section 3, in STT conditions. Since the concatenation and Boosting adaptation underperformed in the reference condition experiments, we only plotted the other adaptation methods for STT conditions.

contextual information, such as the previous or the following dialog act tag, which may improve the performance further.

In this experiment, the goal is to adapt the dialog act tagging model for MRDA data using the SWBD corpus, so that the resulting adapted model can classify the dialog acts more accurately on the MRDA corpus. Although the two corpora are very similar, when the training set does not match the test set, performance drops drastically. Since we expect the proposed adaptation methods to work better with less application- specific training data, we report results using 5,000 in-domain examples from MRDA. With this small amount of in-domain data, concatenation actually decreases the classification accuracy even more, since now the out-of-domain data becomes dominant.

Similar to dialog act segmentation experiments, we performed linear and logistic interpolation methods along with the "out-of-domain as feature" method. Using logistic interpolation performed the best for this experiment. Note that an improvement of around 0.6% is significant according to the Z-test for a 95% confidence interval (see Table 2).

We have also drawn the learning curves for CER as presented in Fig. 10. The top-most curve is obtained using random selection of only MRDA training data. The lower curves are obtained using linear and logistic interpolation. Using scores obtained from the out-of-domain model as features did not help for data sizes of more than 5,000 in-domain utterances. When we employ adaptation with only 5,000 utterances from MRDA, we have seen more than 1% absolute improvement. Nevertheless, the classification with 10,000 meeting dialog act units and the SWBD model reaches the same performance as the one topped by the meeting model trained on 20,000 utterances; this is a factor of 2 reduction in the amount of labeled meeting data needed. We can improve the performance significantly by exploiting the SWBD data when the in-domain data size is less than 25,000. After about 50,000 in-domain utterances the gain disappears completely, as expected.

5.5. Dialog act tagging and sentence segmentation adaptation

After analyzing the effect of adaptation methods on dialog act segmentation and tagging separately, we investigate how they interact with each other. To do this, we repeat the experiments done in the previous section, however using the automatic sentence boundaries.

First we analyze the effect of noise introduced by sentence segmentation when only 2,000 in-domain (MRDA) utterances are available. Table 3 presents the error rates with different metrics as explained above.

Table 2
Baseline dialog act tagging error rates using in-domain (MRDA) and out-of-domain (SWBD) corpora, and adaptation results with various methods. These results are obtained using reference segmentations.

Adaptation	Error rate (%)
MRDA	25.87
SWBD	42.63
Data concatenation	31.24
Out-of-domain as feature	25.27
Logistic interpolation	24.81
Linear interpolation	25.39

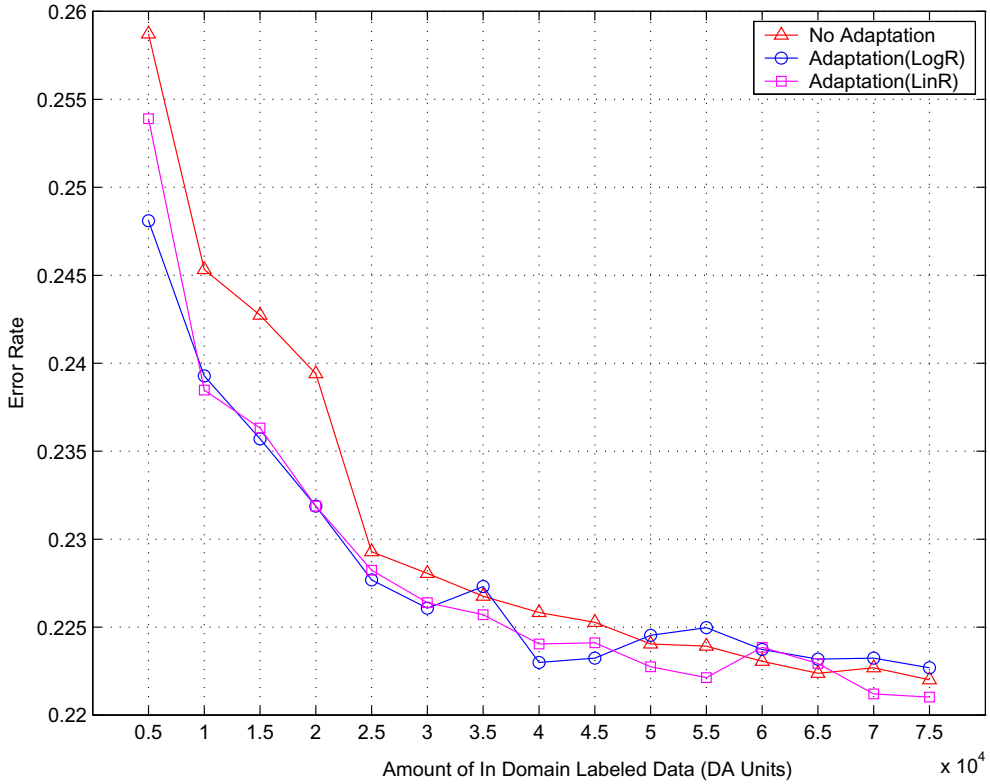

Fig. 10. CER results using dialog act classification model adaptation. The top learning curve is obtained using just ICSI MRDA data as a baseline. The lower learning curves are obtained using the adaptation with the Switchboard corpus where the weights are trained using linear and logistic regression.

Table 3
Interaction of sentence segmentation (SS) and dialog act tagging (DAT) adapted (A) vs. unadapted (U) models with various metrics when only 2000 in-domain sentences are available.

	SS:U DAT:U (%)	SS:U DAT:A (%)	SS:A DAT:U (%)	SS:A DAT:A (%)
Strict	73.91	72.69	73.24	71.95
Lenient	26.22	25.56	24.16	23.51
DER	61.79	60.72	61.36	60.25

The results show the performance for the cases when there is no adaptation (SS:U DAT:U), both systems are adapted (SS:A DAT:A), only the sentence segmentation model is adapted (SS:A DAT:U), and only the dialog act tagging model is adapted (SS:U DAT:A). Logistic interpolation is used as the adaptation method for both segmentation and tagging, since it performed consistently good for both tasks. As seen, adaptation helps sig-

nificantly whether it is done for segmentation or tagging. According to the *lenient* metric, adaptation for segmentation gives a better result than adaptation in tagging. It is the other way around for *strict* and *DER* metrics. This is because segmentation performance is also reflected in both *strict* and *DER* metrics. However, with all the metrics, adaptation performed for both segmentation and tagging gives the best results as expected. The biggest improvement (more than 10% relative) is seen for the lenient error rate metric. This is probably because this is less effected by the segmentation noise hence the DA tagging improvement can show better. This can also be seen comparing the performance of no adaptation (SS:U DAT:U), with only DA segmentation is adapted (SS:A DAT:U), where lenient error rate is reduced by 8% relative while the error rate by other metrics are reduced by less than 1% relative. To further analyze this effect, we checked the best possible DER given the segmentation noise, and when no adaptation is employed for segmentation, the lowest tagging error rate (DER) possible is found to be 45.7%. Note that this number is 0% by definition when all of the estimated dialog act unit boundaries are correct. With adaptation of dialog act segmentation models, this reduces to only 44.0%, which is significantly lower, but still far from perfect.

The next step of our analyses is checking the effect of cascaded adaptation when variable amounts of in-domain annotated data is available. Fig. 11 presents the learning curves according to the *strict* metric for all the four cases described above. When there are fewer than 10,000 sentences, adaptation applied for both segmentation and tagging helps as expected, since individual models have better performances as shown in the previous figures. However after this point, adaptation for sentence segmentation becomes less important and the curves are drawn according to whether the tagging model is adapted or not. This is in contradiction to what we would have expected. This is probably due to the fact that about 1% better segmentation performances at those points become irrelevant for DA tagging with large amount of in-domain data and the strict metric is heavily influenced by this metric. Fig. 12 depicts this influence. When there are only 2,000 in-domain annotated samples, adapting the earlier task, that is segmentation, results in better performance. At around 5,000 in-domain annotated samples, the effect of adaptation for segmentation and tagging result in similar performance. Only after then adapting the tagging results in better performance, consistent with Fig. 11.

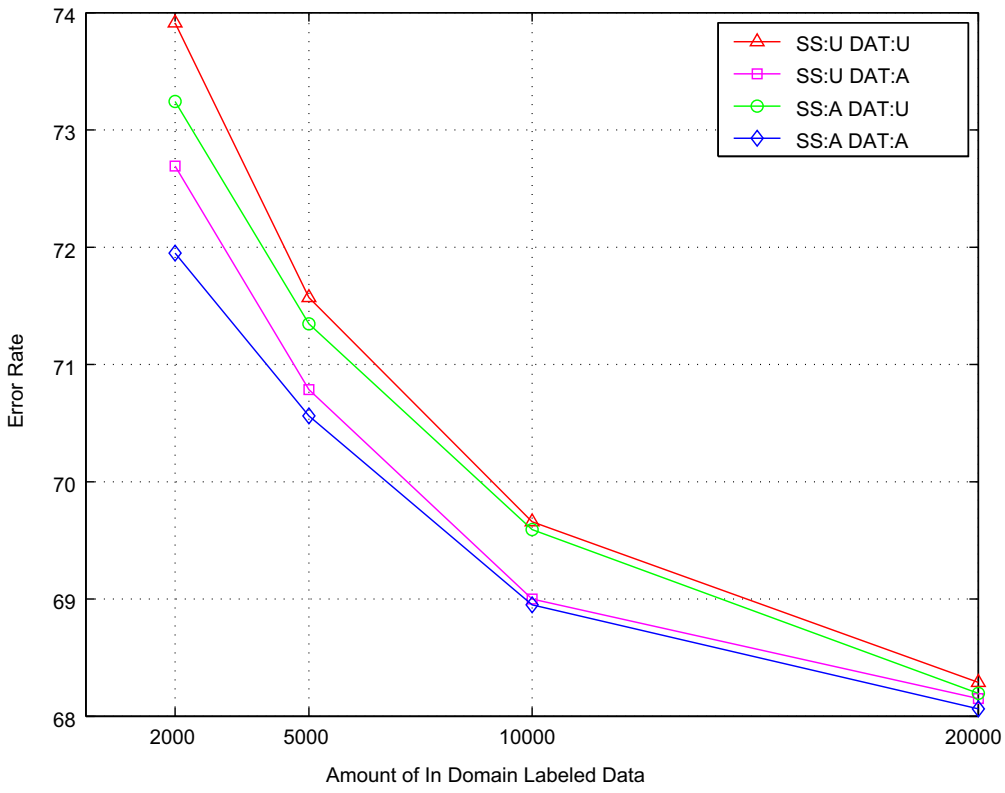

Fig. 11. DA tagging results using adaptation for both dialog act tagging and sentence segmentation with *strict* error metric.

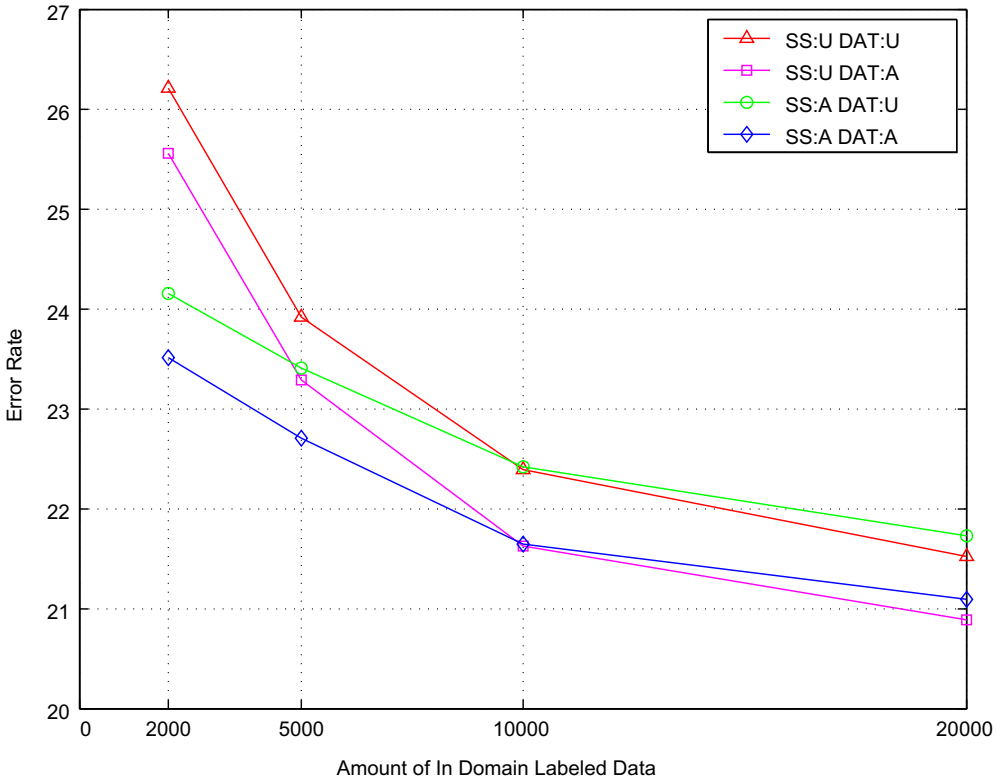

Fig. 12. DA tagging results using adaptation for both dialog act tagging and sentence segmentation with *lenient* error metric.

For both figures, the general picture is the same though: When the amount of in-domain data is limited, there is a consistent gain in performance as a result of the adaptation for each and both of the tasks. But it becomes less significant in line with the findings described in the previous subsections. The effect of adaptation vanishes after 20,000 in-domain annotated examples when each task is considered individually or jointly consistent with Figs. 10 and 8. The important results is that the adaptation of the latter task, that is tagging in our case, is more effective in general.

6. Conclusions

We have presented supervised adaptation methods for dialog act segmentation and tagging. We have shown that, for these tasks, it is possible to boost the performance of both systems when a small amount of training data is available. Our results indicate that we can achieve significantly better dialog act segmentation and tagging by adapting the out-of-domain models, especially when the in-domain data is limited. While the "out-of-domain as feature" method performed the best for dialog act segmentation, we noticed that for tagging, the "logistic regression" method outperformed it. Since logistic regression performed reasonably well for the segmentation task, and the final performance of the combined segmentation and tagging adaptation is dominated by tagging as explained above, we believe that logistic regression results in a more effective and robust adaptation.

We also perform experiments adapting either one or both of segmentation and tagging models. We have presented results on the effect of cascaded adaptation on these two tasks. We show the that it is more effective to adapt the models in the latter classification tasks, in our case tagging, when dealing with a sequence of cascaded classification tasks. The adaptation of the earlier stages are washed away and learning curves are dominated by the performance of the latter stages.

Our future work includes unsupervised adaptation of dialog act segmentation and classification models. This will enable us to bootstrap new dialog act models without labeling any application-specific data. Another venue is combining dialog act tagging with dialog act segmentation and performing a joint adaptation.

Acknowledgements

This material is based upon work supported by Defense Advanced Research Projects Agency (DARPA) CALO (Contract No. FA8750-07-D-0185, Delivery Order 0004), the Scientific and Technological Research Council of Turkey (TUBITAK) fundings at SRI, Isik University Research Fund (Contract No. 05B304), J. William Fulbright Post-Doctoral Research Fellowship, and the Swiss National Science Foundation through the research network, IM2 fundings at ICSI. 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 funding agencies. We thank Elizabeth Shriberg, Andreas Stolcke, Matthias Zimmerman, and Matthew Magimai Doss for many helpful discussions.

References

Ang, J., Liu, Y., Shriberg, E., 2005. Automatic dialog act segmentation and classification in multiparty meetings. In: Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP), Philadelphia, PA.

Arnold, A., Nallapati, R., Cohen, W., 2007. A comparative study of methods for transductive transfer learning. In: Proceedings of the International Conference on Data Mining (ICDM) Workshop on Mining and Management of Biological Data, Omaha, NE.

Bacchiani, M., Riley, M., Roark, B., Sproat, R., 2006. Map adaptation of stochastic grammars. Computer Speech and Language 20 (1), 41–68.

Bacchiani, M., Roark, B., 2003. Unsupervised language model adaptation. In: Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP), Hong Kong.

Bacchiani, M., Roark, B., Saraclar, M., 2004. Language model adaptation with MAP estimation and the perceptron algorithm. In: Proceedings of the Human Language Technology Conference (HLT) –Conference of the North American Chapter of the Association for Computational Linguistics (NAACL), Boston, MA.

Bellegarda, J.R., 2004. Statistical language model adaptation: review and perspectives. Speech Communication Special Issue on Adaptation Methods for Speech Recognition 42, 93–108.

Chelba, C., Acero, A., 2006. Adaptation of maximum entropy capitalizer: little data can help a lot. Computer Speech and Language 20 (4), 382–399.

Chen, L., Gauvain, J.-L., Lamel, L., Adda, G., 2003. Unsupervised language model adaptation for broadcast news. In: Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP), Hong Kong.

Core, M., Allen, J., 1997. Coding dialogs with the DAMSL annotation scheme. In: Proceedings of the Working Notes of the Conference of the American Association for Artificial Intelligence (AAAI) Fall Symposium on Communicative Action in Humans and Machines, Cambridge, MA.

Cuendet, S., Hakkani-Tür, D., Tur, G., 2006. Model adaptation for sentence segmentation from speech. In: Proceedings of the IEEE/ACL Spoken Language Technologies (SLT) Workshop, Aruba.

Daumé, H., Marcu, D., 2006. Practical structured learning techniques for natural language processing. Ph.D. Thesis. University of Southern California, Los Angeles, CA.

Fang, J.L.X., Gao, J., Seng, H., 2003. Training data optimization for language model adaptation. In: Proceedings of the European Conference on Speech Communication and Technology (EUROSPEECH), Geneva, Switzerland.

Gales, M.J.F., 1998. Maximum likelihood linear transformations for HMM-based speech recognition. Computer Speech and Language 12, 75–98.

Gauvain, J.-L., Lee, C.-H., 1994. Maximum a posteriori estimation for multivariate Gaussian mixture observation of Markov chains. IEEE Transactions on Speech and Audio Processing 2 (2), 291–298.

Gotoh, Y., Renals, S., 2000. Sentence boundary detection in broadcast speech transcripts. In: Proceedings of the ISCA ITRW Workshop, Paris.

Gretter, R., Riccardi, G., 2001. On-line learning of language models with word error probability distributions. In: Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP), Salt Lake City, Utah.

Gupta, N., Tur, G., Hakkani-Tür, D., Bangalore, S., Riccardi, G., Rahim, M., 2006. The AT&T spoken language understanding system. IEEE Transactions on Audio, Speech, and Language Processing 14 (1), 213–222.

Hakkani-Tür, D., Tur, G., Rahim, M., Riccardi, G., 2004. Unsupervised and active learning in automatic speech recognition for call classification. In: Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP), Montreal, Canada.

Hillard, D., Ostendorf, M., Stolcke, A., Liu, Y., Shriberg, E., 2004. Improving automatic sentence boundary detection with confusion networks. In: Proceedings of the Human Language Technology Conference (HLT) – Conference of the North American Chapter of the Association for Computational Linguistics (NAACL), Boston, MA.

Janin, A., Ang, J., Bhagat, S., Dhillon, R., Edwards, J., Macias-Guarasa, J., Morgan, N., Peskin, B., Shriberg, E., Stolcke, A., Wooters, C., Wrede, B., 2004. The ICSI meeting project: resources and research. In: Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP), Montreal.

Jurafsky, D., Shriberg, E., Biasca, D., 1997. Switchboard SWBD—DAMSL Labeling Project Coder's Manual. Technical Report 97-02. University of Colorado Institute of Cognitive Science.

Kneser, R., Peters, J., Klakow, D., 1997. Language model adaptation using dynamic marginals. In: Proceedings of the European Conference on Speech Communication and Technology (EUROSPEECH), Rhodes, Greece.

Liu, Y., Shriberg, E., Stolcke, A., Peskin, B., Ang, J., Hillard, D., Ostendorf, M., Tomalin, M., Woodland, P., Harper, M., 2005. Structural metadata research in the EARS program. In: Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP).

Nanjo, H., Kawahara, T., 2003. Unsupervised language model adaptation for lecture speech recognition. In: Proceedings of the ISCA and IEEE Workshop on Spontaneous Speech Processing and Recognition, Tokyo, Japan.

Roark, B., Liu, Y., Harper, M., Stewart, R., Lease, M., Snover, M., Shafran, I., Dorr, B., Hale, J., Krasnyanskaya, A., Yung, L., 2006. Reranking for sentence boundary detection in conversational speech. In: Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toulouse, France.

Schapire, R.E., 2001. The boosting approach to machine learning: an overview. In: Proceedings of the MSRI Workshop on Nonlinear Estimation and Classification, Berkeley, CA.

Schapire, R.E., Rochery, M., Rahim, M., Gupta, N., 2005. Boosting with prior knowledge for call classification. IEEE Transactions on Speech and Audio Processing 13 (2).

Schapire, R.E., Singer, Y., 1999. Improved boosting algorithms using confidence-rated predictions. Machine Learning 37 (3), 297–336.

Schapire, R.E., Singer, Y., 2000. Boostexter: a boosting-based system for text categorization. Machine Learning 39 (2/3), 135–168.

Shriberg, E., 2005. Spontaneous speech: how people really talk and why engineers should care (keynote address). In: Proceedings of the European Conference on Speech Communication and Technology (EUROSPEECH), Lisbon, Portugal.

Shriberg, E., Dhillon, R., Bhagat, S., Ang, J., Carvey, H., 2004. The ICSI meeting recorder dialog act (MRDA) corpus. In: Proceedings of the SigDial Workshop, Boston, MA.

Shriberg, E., Stolcke, A., Hakkani-Tür, D., Tur, G., 2000. Prosody-based automatic segmentation of speech into sentences and topics. Speech Communication 32 (1-2), 127–154.

Stolcke, A., 2001. Error modeling and unsupervised language modeling. In: Proceedings of the NIST LVCSR Workshop, Linthicum, MD.

Stolcke, A., 2002. SRILM – an extensible language modeling toolkit. In: Proceedings of the International Conference on Spoken Language Processing (ICSLP), Denver, CO.

Stolcke, A., Anguera, X., Boakye, K., Cetin, O., Grezl, F., Janin, A., Mandal, A., Peskin, B., Wooters, C., Zheng, J., 2005. Further progress in meeting recognition: the ICSI–SRI spring 2005 speech-to-text evaluation system. In: Proceedings of the NIST Meeting Recognition Workshop, Edinburgh, UK.

Stolcke, A., Ries, K., Coccaro, N., Shriberg, E., Bates, R., Jurafsky, D., Taylor, P., Martin, R., van Ess-Dykema, C., Meteer, M., 2000. Dialogue act modeling for automatic tagging and recognition of conversational speech. Computational Linguistics 26 (3), 339–373.

Stolcke, A., Shriberg, E., 1996. Automatic linguistic segmentation of conversational speech. In: Proceedings of International Conference on Spoken Language Processing (ICSLP), Philadelphia, PA.

Tur, G., 2005. Model adaptation for spoken language understanding. In: Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP), Philadelphia, PA.

Tur, G., Guz, U., Hakkani-Tür, D., 2006. Model adaptation for dialog act tagging. In: Proceedings of the IEEE/ACL Spoken Language Technologies (SLT) Workshop.

Tur, G., Hakkani-Tür, D., 2003. Exploiting unlabeled utterances for spoken language understanding. In: Proceedings of the European Conference on Speech Communication and Technology (EUROSPEECH), Geneva, Switzerland.

Tur, G., Stolcke, A., Voss, L., Dowding, J., Favre, B., Fernandez, R., Frampton, M., Frandsen, M., Frederickson, C., Graciarena, M., Hakkani-Tür, D., Kintzing, D., Leveque, K., Mason, S., Niekrasz, J., Peters, S., Purver, M., Riedhammer, K., Shriberg, E., Tien, J., Vergyri, D., Yang, F., 2008. The CALO meeting speech recognition and understanding system. In: Proceedings of the IEEE/ACL Spoken Language Technologies (SLT) Workshop, Goa, India.

Venkataraman, A., Liu, Y., Shriberg, E., Stolcke, A., 2005. Does active learning help automatic dialog act tagging in meeting data? In: Proceedings of the Interspeech, Lisbon, Portugal.

Venkataraman, A., Stolcke, A., Shriberg, E.E., 2002. Automatic dialog act tagging with minimal supervision. In: Proceedings of the Australian International Conference on Speech Science and Technology, Melbourne, Australia.

Zimmerman, M., Hakkani-Tür, D., Fung, J., Mirghafori, N., Gottlieb, L., Shriberg, E., Liu, Y., 2006. The ICSI + multilingual sentence segmentation system. In: Proceedings of the International Conference on Spoken Language Processing (ICSLP), Pittsburg, PA.

Zimmermann, M., Hakkani-Tür, D., Shriberg, E., Stolcke, A., 2006. Text based dialog act classification for multiparty meetings. In: Proceedings of the Workshop on Multimodal Interaction and Related Machine Learning Algorithms (MLMI), Washington D.C.

Zimmermann, M., Liu, Y., Shriberg, E., Stolcke, A., 2005. Toward joint segmentation and classification of dialog acts in multiparty meetings. In: Proceedings of the Workshop on Multimodal Interaction and Related Machine Learning Algorithms (MLMI), Edinburgh, UK.