TRAINING A PROSODY-BASED DIALOG ACT TAGGER FROM UNLABELED DATA

Anand Venkataraman   Luciana Ferrer   Andreas Stolcke   Elizabeth Shriberg

Speech Technology and Research Laboratory
SRI International, Menlo Park, CA

ABSTRACT

Dialog act tagging is an important step toward speech understanding, yet training such taggers usually requires large amounts of data labeled by linguistic experts. Here we investigate the use of *unlabeled* data for training HMM-based dialog act taggers. Three techniques are shown to be effective for bootstrapping a tagger from very small amounts of labeled data: iterative relabeling and retraining on unlabeled data; a dialog grammar to model dialog act context, and a model of the prosodic correlates of dialog acts. On the SPINE dialog corpus, the combined use of prosodic information and unlabeled data reduces the tagging error between 12% and 16%, compared to baseline systems using word information and various amounts of labeled data only.

1. INTRODUCTION

Knowing the discourse function (dialog act, or DA) of an utterance is useful in a number of applications, including speech recognition and understanding. Intuitively, we aim to classify utterances in dialogs as "questions", "requests", "acknowledgments", and the like, based upon which further semantic and pragmatic processing can take place. Furthermore, DA modeling can constrain the language models used in word recognition, improving word-level accuracy. We call a system that classifies utterances into a finite set of DA categories a *dialog act tagger*. Many methods for DA tagging have been proposed, but the most commonly used ones rely on HMM modeling [1, 2]. A survey of various approaches and how DA tagging integrates with speech recognition models can be found in [3].

One pervasive problem for data-driven DA taggers is that they rely heavily on lexical features, usually in the form of DA-specific statistical language models (LMs). Such models require considerable amounts of training data to be effective, yet hand-labeling dialogs by DA annotation experts is time consuming, and highly domain-specific (meaning that models are not easy generalized across tasks) [4]. To alleviate this problem we have begun to study the use of *unlabeled* training data to improve DA taggers, bootstrapping from only a very small amount of hand-labeled data. In previous work [5] we found that a purely word-based DA tagger could be improved considerably by an iterative automatic labeling and retraining on unlabeled data. We also found that statistical dialog grammars, which model the contextual constraints in DA sequences and which have a smaller vocabulary than word-based DA LMs, can effectively boost accuracy even with small amounts of hand-labeled training data.

Here we focus on the use of another knowledge source to improve unsupervised training: prosody. Prosodic cues have been shown effective for DA tagging in a supervised setting [6, 7], and may require less training data than corresponding lexical models

Table 1. DAs and their distributions in the full bootstrap data. Vocabulary size is the total number of distinct words within a class in the data. RFA stands for a request for acknowledgment and ACK stands for an acknowledgment.

DA	Description	Proportion	Vocab.
SS	Action statements	23.55%	211
SF	Target location	15.86%	183
SH	Status reports	10.74%	74
SO	All other statements	7.12%	304
Q	Questions/RFA	7.98%	190
A	Answers and ACKs	34.75%	86

for a given task [8]. This suggests that prosodic information could be especially effective with little hand-labeled data, and when combined with unlabeled training data.

The paper is organized as follows. Section 2 describes the data we used and how it was partitioned in weakly supervised training. Section 3 describes the prosodic features and model used. Section 4 summarizes the HMM-based tagging paradigm. Section 5 gives and discusses our results. Section 6 concludes and suggests future work.

2. DATA

Our experimental data derives from the SPINE (Speech in Noisy Environments) corpus, which was created for developing speech recognition in military noise environments [9]. The data set is identical to the one used in [5]. However, the partitioning of the data used here is different from that in the earlier study. The reason for repartitioning was mainly to enable the construction of a new partition of labeled data disjoint from both the bootstrap and evaluation sets to be used for tuning of variable parameters in our proposed method.

The SPINE corpus consists of a total of 44,412 DA units in 324 dialogs. Randomly selected contiguous dialog blocks averaging about 32 dialog units each, from 89 randomly selected dialogs, were hand-labeled for DA classes by one of the authors. Because of resource constraints, multiple annotations and interlabeler agreements were not computed in this preliminary explorative study. Nevertheless, we took care to ensure that the inventory of discourse functions used for labeling was much more detailed than was necessary here, in the hope that a continuation of this work will find use for a larger set of dialog acts. For the experiments reported in this paper, we mapped the detailed dialog acts into a smaller set of six broad DA classes, which are listed in Table 1. The 89 annotated dialog fragments were subsequently split into a 20-dialog set (632 utterances) for parameter tuning, a 32-

Table 2. Various amounts of bootstrap data tested. The name of the experiment in column 1 also indicates the number of dialogs used for bootstrapping. Column 2 shows the total number of utterances in all these dialogs, and column 3 shows what percentage of the training data this amount would constitute.

Expt.	Utterances.	% of Training
Boot-32	970	3.27%
Boot-16	474	1.60%
Boot-08	245	0.83%
Boot-04	116	0.39%
Boot-02	57	0.19%

dialog set (970 utterances) to use for bootstrapping the model, and a 39-dialog set (1192 utterances) that was held out for evaluation of the model. The remaining 235 dialogs which were unannotated were split into a 209-dialog training set (29,471 utterances) and a 26-dialog validation set (3274 utterances). Although the (unlabeled) validation set was used here, it was set aside to make sure that we used the same training set as in [5]. Decisions regarding the values of tunable parameters were made via evaluation for accuracy on the 20-dialog held-out tuning set that was hand-annotated for DA classes. Five different subsets of the 32-dialog bootstrap set were subsequently used for bootstrapping our model in five sets of experiments. The experiments investigated the effect of diminishing bootstrap data size on the performance of the classifier. These five bootstrap data sets are shown in Table 2.

3. PROSODIC MODELING

We used three types of prosodic features in our model—duration, energy, and pitch. Duration features were extracted from the alignment information obtained from a speech recognizer. Energy features were estimated from frame-level energy information in the signal. Pitch features were obtained from a postprocessed version of the pitch track obtained using an improved version of the approach in [10]. Pitch contours were *stylized*, octave errors were estimated, and, most importantly, a set of speaker-specific pitch range parameters was computed. These parameters include a value that allows us to estimate a speaker's *floor* or lowest typical F0 value. We used some additional features to represent the duration between the end of the previous speaker's utterance and the onset of the current speaker's utterance as well as a flag to represent speaker change between utterances.

As in prior work [6, 8], CART-style decision trees were used as the prosodic model. During each iteration, a tree was trained on the prosodic feature vectors by using the classifications obtained in the preceding iteration. An automatic feature selection algorithm was employed to choose the best subset of the full set of features. This algorithm used a held-out subset of the training data for estimating this subset, subsequently rebuilding the tree with the full training set after the subset was identified. As the process of training the trees was computationally very expensive, we calculated a core subset of *essential features* that formed the set of features used in all our experiments. This was done by using the largest bootstrap set to obtain a set of trees (each of them using a different subset of the full set of features) and subsequently keeping only those features that had a highest average (subject to a threshold) feature usage (as implied by the number of tree nodes at which that feature was used to make a decision) among all the trees that performed

better than the baseline. The subset of features we finally used was as follows:

1. Difference between average stylized pitch in the first and the last 20-ms windows.

2. Log-ratio of the last stylized pitch and the speaker's floor pitch.

3. Log-ratio of the first stylized pitch and the speaker's floor pitch.

4. Log-ratio of the maximum stylized pitch in the last word and the speaker's floor pitch.

5. Location of the maximum stylized pitch normalized by total duration.

6. Pause duration from the last unit and the current unit of the current speaker.

7. Flag indicating speaker change.

4. MODEL

Our DA tagger model employs the widely used HMM tagging paradigm to integrate lexical, prosodic, and contextual knowledge [3]. In this framework, HMM states represent individual DAs. The individual utterances (words and/or prosody) correspond to the observations generated by the states. The theory behind our implementation can be summarized as follows. Let $\mathbf{D} = D_1 \cdots D_n$ be the sequence of dialog acts associated with the n utterances in a given dialog. Let $\mathbf{W} = W_1 \cdots W_n$ be the sequence of utterances in the same dialog and $\mathbf{F} = F_1 \cdots F_n$ be the sequence of prosodic feature vectors characterizing each of the n utterances. If we approximate the observations by $\mathbf{F}$ and $\mathbf{W}$, then we seek $\widehat{\mathbf{D}} = \mathrm{argmax}_{\mathbf{D}} \, \mathrm{P}(\mathbf{D}, \mathbf{W}, \mathbf{F})$. Making the standard Markov assumptions, we can write

$$\mathrm{P}(\mathbf{D}, \mathbf{W}, \mathbf{F}) = \prod_{i=1}^{n} \mathrm{P}(D_i | \mathbf{D}_{ih}) \mathrm{P}(W_i, F_i | D_i)$$

where $\mathbf{D}_{ih}$ is the h-length, possibly truncated, history of the ith DA in the sequence of DAs. We estimate $\mathrm{P}(D_i | \mathbf{D}_{ih})$ in the standard way from counts of DA N-grams. To calculate $\mathrm{P}(W_i, F_i | D_i)$, we observe that $\mathrm{P}(W_i, F_i | D_i)$ can be decomposed as follows:

$$\mathrm{P}(W_i, F_i | D_i) = \mathrm{P}(W_i | D_i, F_i) \mathrm{P}(F_i | D_i) \tag{1}$$

By assuming, not unreasonably, that the observed words are independent of the prosodic features given the discourse category of the sentence, we may simplify (1) to

$$\mathrm{P}(W_i, F_i | D_i) = \mathrm{P}(W_i | D_i) \mathrm{P}(F_i | D_i)$$

Now the first term on the right hand side is straightforward to estimate from a DA-specific language model trained only on sentences belonging to the particular DA. But we still face the problem of calculating $\mathrm{P}(F_i | D_i)$. Here, suppose that we had some means of obtaining a posterior distribution over dialog acts given the prosodic features. Then the problem becomes one of estimating $\mathrm{P}(F_i | D_i)$ from $\mathrm{P}(D_i | F_i)$. By Bayes' rule,

$$\mathrm{P}(D_i | F_i) = \frac{\mathrm{P}(F_i | D_i) \mathrm{P}(D_i)}{\mathrm{P}(F_i)} \tag{2}$$

If we arranged things so that $P(D)$, the prior probability of observing any particular DA, is independent of the DA itself, then the posterior estimated by the above equation will reduce to a constant multiple of $P(F_i|D_i)$ since $P(F_i)$ is itself independent of any particular DA. We thus obtain $P(D_i|F_i) = \kappa_i P(F_i|D_i)$ for some non-negative constant $\kappa_i = 1/\alpha_i$ or equivalently $P(F_i|D_i) = \alpha_i P(D_i|F_i)$. Substituting, we get

$$\widehat{\mathbf{D}} = \arg\max_{\mathbf{D}} \prod_{i=1}^{n} P(D_i|\mathbf{D}_{ih})P(W_i|D_i)P(D_i|F_i) \qquad (3)$$

which is considerably easier to compute. $P(W_i|D_i)$ can be obtained from DA-specific language models and $P(D_i|F_i)$ can be obtained from a decision tree. Still, in order for this scheme to work we must ensure that we have a uniform prior distribution over DAs when using a classifier to generate the posterior. Fortunately, this condition is easily met. We do this by resampling the tree training data to represent all DA classes equally well. A final subtlety regarding the model is that since we estimate language model likelihoods and DA posteriors conditioned on the feature vectors using two different paradigms, we ought to determine the best way to combine the two values. To do this, we rewrite (3) as

$$\widehat{\mathbf{D}} = \arg\max_{\mathbf{D}} \prod_{i=1}^{n} P(D_i|\mathbf{D}_h)P(W_i|D_i)P(D_i|F_i)^{\gamma} \qquad (4)$$

where γ is an exponent representing our intuitive notion of the degree of importance we might attach to the DA posterior in relation to the language model likelihoods. If $\gamma = 0$, then exactly the model of [5] obtains since the exponentiated posterior is 1 and thus the contribution of the prosodic feature vector to any discrimination between DAs is nil. With $0 < \gamma \leq 1$, the contribution of the prosodic feature vector to the discriminative power of the model is non-nil, but still attenuated, since by raising the posterior to a fractional power, we effectively make the resultant value closer to 1. Conversely, with $\gamma > 1$, the posterior's contribution to the discrimination is amplified. We estimated a near-optimal value of γ by a simple grid search, minimizing the DA classification error on the tuning set.

In all our experiments, we restricted ourselves to using trigrams both for the DA sequence LM (DA grammar) and for DA-specific LMs, as these were found to give the best results in [5]. The rest of the procedure we followed is the same as described in the earlier work: an initial tagger is trained from the available labeled data (the bootstrap data); it is then used to tag the unlabeled training data, after which the model is retrained using all training data. The resulting tagger is presumably improved over the initial one that was trained on the bootstrap data alone; we should therefore iterate the training and relabeling. Any improvement obtained after the first iteration is a result of incorporation of the unlabeled data into the training procedure. As discussed in [5], the repeated retraining procedure may be seen as an approximate form of expectation-maximization [11] aimed at maximizing the training data likelihood.

Evaluation metrics

In [5], the best iteration from which to obtain models was determined by calculating the language model perplexity on some held-out data, referred to as the *validation set*. In fact, it was found that
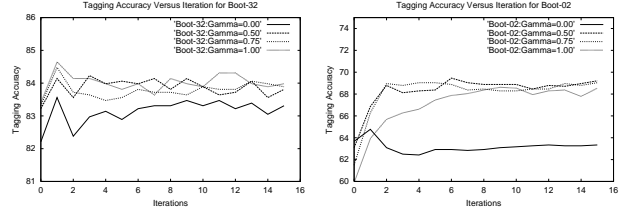

Fig. 1. Accuracy versus iteration for the Boot-32 and Boot-02 experiments, respectively. Iteration 0 corresponds to the case where training was performed with only the bootstrap data. For Boot-32 and $\gamma = 0$, the best iteration was 2 and accuracy at that iteration was 82.38%. By best iteration, we mean the iteration at which the validation set accuracy was highest. For $\gamma = 0.5$, 0.75 and 1.0, the best iteration was 1 and the evaluation accuracies at this iteration were 84.14%, 84.48% and 84.65% respectively. For Boot-02, corresponding numbers were 63.09% at iteration 2, 68.124 and 67.87% at iteration 6 respectively.

Table 3. Summary of results. *Baseline* in column 2 shows the accuracy of the baseline model ($\gamma = 0$). Columns 3 and 4 show the value of γ and the iteration number at which validation set accuracy was the highest, respectively. Columns 5 and 6, show the accuracy of the experiment after iteration 0, which represents the performance of the system after training only on the bootstrap set and at the best iteration.

EXPT	Baseline	γ	Iter	Iter-0	Best
Boot-32	82.38	1.00	1	84.65	84.65
Boot-16	79.70	0.80	4	79.03	80.96
Boot-08	78.78	0.50	3	76.09	*77.94*
Boot-04	61.83	0.35	6	61.24	64.60
Boot-02	63.09	0.45	3	67.11	68.37

the validation set perplexity was a good criterion for picking models with near optimal classification accuracies. Unfortunately, here we are unable to use the same criterion for determining the values of the tunable parameters — γ and the iteration number. The reason for this is that by substituting $P(F_i|DA_i)$ with $P(DA_i|F_i)^{\gamma}$, we lose the ability to obtain real probability estimates and thus the ability to calculate the perplexity of a given corpus that can be meaningfully compared to the perplexity obtained with a different value of γ. Therefore, we use the 20-dialog held-out portion from the bootstrap set as our validation set on which we calculate the classification accuracy at each iteration. The evaluation accuracies we report in this paper are those from the iteration at which this validation set accuracy was maximum.

5. RESULTS AND DISCUSSION

We evaluated our framework for different amounts of bootstrap data as shown in Table 2. The two extreme conditions for bootstrap data were Boot-32 and Boot-02, with 970 and 57 utterances, respectively. Because of space constraints we present graphical results in Figure 1 on only these two sets for $\gamma = 0, 0.25, 0.50$ and 1.0, while, however, discussing the other results in text.

Table 3 summarizes the results for all the bootstrap sets. The

results can be summarized as follows. First, we confirmed that prosodic cues are effective in improving the baseline (iteration 0) accuracies of the DA tagger. Second, we are always able to do better than the accuracy at the end of iteration 0, even if by just one more iteration. This shows that learning from unlabeled training examples is clearly of help, both with and without prosodic modeling. Third, as one would expect, this addition of the unlabeled training data is most helpful for the smallest bootstrap sets. However, as in our previous work, we observed that it is better to use all available labeled data for training whenever possible. That is, the small-bootstrap-size learners never catch up with the initial accuracy of the large-bootstrap-size learners, in spite of the fact that substantially more unlabeled was used in the former than labeled data in the latter ("There is no data like labeled data").

Finally, and most importantly from the perspective of this study, automatic labeling seems to combine well with the use of prosodic information. In all but one case (further discussed below), the prosody-based tagger obtained with unlabeled data improves upon the tagger using unlabeled data, but no prosodic information. Combining unlabeled data and prosodic cues gives and overall tagging error reduction between 12% and 16% relative, compared to the baseline of using only the lexical, labeled training data.

A further point that is not obvious from the results shown, but which we observed in the results, was that although the incorporation of prosodic information into the model usually improved the performance of the baseline model, it was hard to predict any general trend in the amounts of such improvements between the various bootstrap set sizes. In particular, as Table 3 shows, there is no improvement over the baseline in Boot-08 and Boot-02 is better than Boot-04 even though it uses less boot data.[1] Upon closer inspection of the data, we found many dialogs which had extremely skewed distributions of dialog classes *within* them. For instance, in some dialogs the participants would be talking not within the task assigned to them, but rather *about it*, and all the utterances within this dialog might be labeled with SO (for statement-other). Similarly many dialogs were totally lacking in the question class. While the large bootstrap sets seemed to be immune to such problems, the small bootstrap sets were particularly prone to be affected by such issues. Preliminary investigation showed that although the language model is able to generalize gracefully in the case of little training data, the decision tree is not. This causes the induced trees to be overly sensitive to the particular choice of a bootstrap set. For this reason, accuracy comparisons are only meaningful when compared against a baseline ($\gamma = 0$) model that uses the same bootstrap set and not a different bootstrap set albeit of comparable size.

6. CONCLUSIONS AND FUTURE WORK

Our experiments on the SPINE dialog data have shown that unlabeled data can be used effectively for training HMM-based dialog act taggers, through an iterative relabeling and retraining procedure. Furthermore, prosodic features modeled by decision tree classifier can be integrated into this framework, and help tagging accuracy both with labeled and unlabeled training data.

Several questions remain for future work. One is whether the results carry over to other and potentially much larger corpora and

---

[1] Boot-08 did produce models that had better accuracy than the baseline at several iterations, but unfortunately, these were not the models which had the best validation set accuracy

DA labeling systems, such as the Switchboard-DAMSL system developed for conversational telephone speech [4]. This then raises the question of whether unlabeled data can boost the effectiveness of DA taggers for improving speech recognition LMs beyond previous results, which showed only modest benefits [3]. Furthermore, it will be interesting to see if completely unsupervised DA taggers (using DA categories defined only in terms of similarity in some suitable feature space) might be effective for speech processing tasks.

7. REFERENCES

[1] M. Nagata, "Using pragmatics to rule out recognition errors in cooperative task-oriented dialogues", in J. J. Ohala, T. M. Nearey, B. L. Derwing, M. M. Hodge, and G. E. Wiebe, editors, *Proc. ICSLP*, vol. 1, pp. 647–650, Banff, Canada, Oct. 1992.

[2] M. Nagata and T. Morimoto, "An experimental statistical dialogue model to predict the speech act type of the next utterance", in K. Shirai, T. Kobayashi, and Y. Harada, editors, *Proceedings of the International Symposium on Spoken Dialogue*, pp. 83–86, Tokyo, Nov. 1993.

[3] A. Stolcke, K. Ries, N. Coccaro, E. Shriberg, D. Jurafsky, P. Taylor, R. Martin, C. Van Ess-Dykema, and M. Meteer, "Dialogue act modeling for automatic tagging and recognition of conversational speech", *Computational Linguistics*, vol. 26, pp. 339–373, 2000.

[4] D. Jurafsky, E. E. Shriberg, B. Fox, and T. Curl, "Lexical, prosodic, and syntactic cues for dialog acts", *in Proceedings of ACL/COLING-98 Workshop on Discourse Relations and Discourse Markers*, pp. 114–120. Association for Computational Linguistics, 1998.

[5] A. Venkataraman, A. Stolcke, and L. Shriberg, "Automatic dialog act tagging with minimal supervision", *in Proc. 9th Australian International Conference on Speech Science and Technology*, Melbourne, Australia, Dec. 2002. Australian Speech Science and Technology Association.

[6] E. Shriberg, R. Bates, A. Stolcke, P. Taylor, D. Jurafsky, K. Ries, N. Coccaro, R. Martin, M. Meteer, and C. Van Ess-Dykema, "Can prosody aid the automatic classification of dialog acts in conversational speech?", *Language and Speech*, vol. 41, pp. 439–487, 1998.

[7] P. Taylor, S. King, S. Isard, and H. Wright, "Intonation and dialog context as constraints for speech recognition", *Language and Speech*, vol. 41, pp. 489–508, 1998.

[8] E. Shriberg, A. Stolcke, D. Hakkani-Tür, and G. Tür, "Prosody-based automatic segmentation of speech into sentences and topics", *Speech Communication*, vol. 32, pp. 127–154, Sep. 2000, Special Issue on Accessing Information in Spoken Audio.

[9] Navy Research Laboratory, "Speech in Noisy Environments", http://elazar.itd.nrl.navy.mil/spine/, 2001.

[10] K. Sönmez, E. Shriberg, L. Heck, and M. Weintraub, "Modeling dynamic prosodic variation for speaker verification", in R. H. Mannell and J. Robert-Ribes, editors, *Proc. ICSLP*, vol. 7, pp. 3189–3192, Sydney, Dec. 1998. Australian Speech Science and Technology Association.

[11] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the *EM* algorithm", *Journal of the Royal Statistical Society, Series B*, vol. 34, pp. 1–38, 1977.