Virtual Evidence for Training Speech Recognizers using Partially Labeled data

Amarnag Subramanya
Department of Electrical Engineering
University of Washington
Seattle, WA 98195-2500
asubram@u.washington.edu

Jeff Bilmes
Department of Electrical Engineering
University of Washington
Seattle, WA 98195-2500
bilmes@ee.washington.edu

Abstract

Collecting supervised training data for automatic speech recognition (ASR) systems is both time consuming and expensive. In this paper we use the notion of virtual evidence in a graphical-model based system to reduce the amount of supervisory training data required for sequence learning tasks. We apply this approach to a TIMIT phone recognition system, and show that our VE-based training scheme can, relative to a baseline trained with the full segmentation, yield similar results with only 15.3% of the frames labeled (keeping the number of utterances fixed).

1 Introduction

Current state-of-the-art speech recognizers use thousands of hours of training data, collected from a large number of speakers with various backgrounds in order to make the models more robust. It is well known that one of the simplest ways of improving the accuracy of a recognizer is to increase the amount of training data. Moreover, speech recognition systems can benefit from being trained on hand-transcribed data where all the appropriate word level segmentations (i.e., the exact time of the word boundaries) are known. However, with increasing amounts of raw speech data being made available, it is both time consuming and expensive to accurately segment every word for every given sentence. Moreover, for languages for which only a small amount of training data is available, it can be expensive and challenging to annotate with precise word transcriptions – the researcher may have no choice but to use partially erroneous training data.

There are a number of different ways to label data used to train a speech recognizer. First, the most expensive case (from an annotation perspective) is fully supervised training, where both word sequences and time segmentations are completely specified[1]. A second case is most commonly used in speech recognition systems, where only the word sequences of utterances are given, but their precise segmentations are unknown. A third case falls under the realm of semi-supervised approaches. As one possible example, a previously trained recognizer is used to generate transcripts for unlabeled data, which are then used to re-train the recognizer based on some measure of recognizer confidence (Lamel et al., 2002).

The above cases do not exhaust the set of possible training scenarios. In this paper, we show how the notion of virtual evidence (VE) (Pearl, 1988) may be used to obtain the benefits of data with time segmentations but using only partially labeled data. Our method lies somewhere between the first and second cases above. This general framework has been successfully applied in the past to the activity recognition domain (Subramanya et al., 2006). Here we make use of the TIMIT phone recognition task as an example to show how VE may be used to deal with partially labeled speech training data. To the best of our knowledge, this paper presents the first system to express training uncertainty using VE in the speech domain.

2 Baseline System

Figure 1 shows two consecutive time slices of a dynamic Bayesian network (DBN) designed for con-

[1]This does not imply that all variables are observed during training. While the inter-word segmentations are known, the model is not given information about intra-word segmentations.

Proceedings of NAACL HLT 2007, Companion Volume, pages 165–168,
Rochester, NY, April 2007.

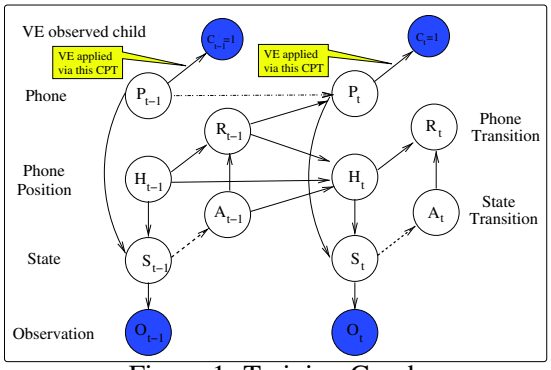

Figure 1: Training Graph.

text independent (CI) phone recognition. All observed variables are shaded, deterministic dependences are depicted using solid black lines, value specific dependences are shown using a dot-dash lines, and random dependencies are represented using dashed lines. In this paper, given any random variable (rv) X, x denotes a particular value of that rv, D_X is the domain of X ($x \in D_X$), and $|D_X|$ represents its cardinality.

In the above model, P_t is the rv representing the phone variable, H_t models the current position within a phone, S_t is the state, O_t the acoustic observations, A_t and R_t indicate state and phone transitions respectively. Here, $D_{X_t} = D_{X_{t-1}}$, $\forall t, \forall X$. In our implementation here, $D_{H_t}, D_{A_t} \in \{0, 1, 2\}$, $D_{R_t} \in \{0, 1\}$. Also $\delta\{c_1, \ldots, c_n\}$ is an indicator function that turns on when all the conditions $\{c_1, \ldots, c_n\}$ are true (i.e. a conjunction over all the conditions). The distribution for H_t is given by $p(h_t|h_{t-1}, r_{t-1}, a_{t-1}) = \delta_{\{h_t=0, r_{t-1}=1\}} + \delta_{\{h_t=a_{t-1}+h_{t-1}, r_{t-1}=0\}}$, which implies that we always start a phone with $H_t = 0$. We allow skips in each phone model, and A_t=0, indicates no transition, A_t=1 implies you transition to the next state, A_t=2 causes a state to skip ($H_{t+1} = H_t + 2$). As the TIMIT corpus provides phone level segmentations, P_t is observed during training. However, for reasons that will become clear in the next section, we treat P_t as hidden but make it the parent of a rv C_t, with, $p(c_t = 1|p_t) = \delta_{l_t=p_t}$ where l_t is obtained from the transcriptions ($l_t \in D_{P_t}$). The above formulation has exactly the same effect as making P_t observed and setting it equal to l_t (Bilmes, 2004). Additional details on other CPTs in this model may be found in (Bilmes and Bartels, 2005). We provide more details on the baseline system in section 4.1.

Our main reason for choosing the TIMIT phone recognition task is that TIMIT includes both sequence and segment transcriptions (something rare for LVCSR corpora such as Switchboard and Fisher). This means that we can compare against a model that has been trained fully supervised. It is also well known that context-dependent (CD) models outperform CI models for the TIMIT phone recognition task (Glass et al., 1996). We used CI models primarily for the rapid experimental turnaround time and since it still provides a reasonable test-bed for evaluating new ideas. We do note, however, that our baseline CI system is competitive with recently published CD systems (Wang and Fosler-Lussier, 2006), albeit which uses many fewer components per mixture (see Section 4.1).

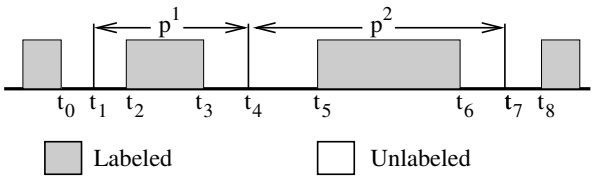

Figure 2: Illustration showing our rendition of Virtual Evidence.

3 Soft-supervised Learning With VE

Given a joint distribution over n variables $p(x_1, \ldots, x_n)$, "evidence" simply means that one of the variables (w.l.o.g. x_1) is known. We denote this by $\bar{x}_1$, so the probability distribution becomes $p(\bar{x}_1, \ldots, x_n)$ (no longer a function of x_1). Any configuration of the variables where $x_1 \neq \bar{x}_1$ is never considered. We can mimic this behavior by introducing a new virtual child variable c into the joint distribution that is always observed to be one (so $c = 1$), and have c interact only with x_1 via the CPT $p(c = 1|x_1) = \delta_{x_1=\bar{x}_1}$. Therefore, $\sum_{x_1} p(c = 1, x_1, \ldots, x_n) = p(\bar{x}_1, \ldots, x_n)$. Now consider setting $p(c = 1|x_1) = f(x_1)$, where $f()$ is an arbitrary non-negative function. With this, different treatment can be given to different assignments to x_1, but unlike hard evidence, we are not insisting on only one particular value. This represents the general notion of VE. In a certain sense, the notion of VE is similar to the prior distribution in Bayesian inference, but it is different in that VE expresses preferences over combinations of values of random variables whereas a Bayesian prior expresses preferences over combinations of model parameter values. For a more information on VE, see (Bilmes, 2004; Pearl, 1988).

VE can in fact be used when accurate phone level segmentations are not available. Consider the illustration in Figure 2. As shown, t_1 and t_4 are the

start and end times respectively for phone p^1, while t_4 and t_7 are the start and end times for phone p^2. When the start and end times for each phone are given, we have information about the identity of the phone that produced each and every observation. The general training scenario in most large vocabulary speech recognition systems, however, does not have access to these starting/ending times, and they are trained knowing only the sequence of phone labels (e.g., that p^2 follows p^1).

Consider a new transcription based on Figure 2, where we know that p^1 ended at some time $t_3 \leq t_4$ and that p^2 started at sometime $t_5 > t_4$. In the region between t_3 and t_5 we have no information on the identity of the phone variable for each acoustic frame, except that it is either p^1 or p^2. A similar case occurs at the start of phone p^1 and the end of phone p^2. The above information can be used in our model (Figure 1) in the following way (here given only for $t_2 \leq t \leq t_6$): $p(C_t = 1|p_t) = \delta_{\{p_t=p^1, t_2 \leq t \leq t_3\}} + \delta_{\{p_t=p^2, t_5 \leq t \leq t_6\}} + f_t(p^1)\delta_{\{p_t=p^1, t_3 \leq t \leq t_5\}} + g_t(p^2)\delta_{\{p_t=p^2, t_3 \leq t \leq t_5\}}$. Here $f_t(p^1)$ and $g_t(p^2)$ represent our relative beliefs at time t in whether the value of P_t is either p^1 or p^2. It is important to highlight that rather than the absolute values of these functions, it is their relative values that have an effect on inference (Bilmes, 2004). There are number of different ways of choosing these functions. First, we can set $f_t(p^1) = g_t(p^2) = \alpha, \alpha > 0$. This encodes our uncertainty regarding the identity of the phone in this region while still forcing it to be either p^1 or p^2, and equal preference is given for both (referred to as "uniform over two phones"). Alternatively, other functions could take into account the fact that, in the frames 'close' to t_3, it is more likely to be p^1, whereas in the frames 'close' to t_5, it is more likely to be p^2. This can be represented by using a decreasing function for $f_t(p^1)$ and an increasing function for $g_t(p^2)$ (for example linearly increasing or decreasing with time).

As more frames are dropped around transitions (e.g., as $t_3 - t_2$ decreases), we use lesser amounts of labeled data. In an extreme situation, we can drop all the labels ($t_3 < t_2$) to recover the case where only sequence and not segment information is available. Alternatively, we can have $t_3 = t_2 + 1$, which means that only one frame is labeled for every phone in an utterance — all other frames of a phone are left untranscribed. From the perspective of a transcriber, this simulates the task of going through an utterance and identifying only one frame that belongs to each particular phone without having to identify the phone boundary. In contrast to the task of determining the phone boundary, identifying one frame per word unit is much simpler, less prone to error or disagreement, and less costly (Greenberg, 1995).

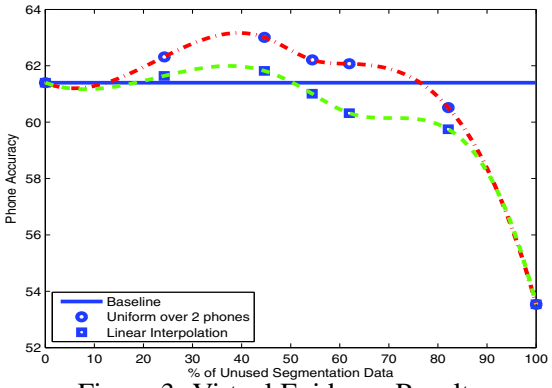

Figure 3: Virtual Evidence Results

4 Experimental Results

4.1 Baseline System

We trained a baseline TIMIT phone recognition system that made full use of all phone level segmentations (the fully supervised case). To obtain the acoustic observations, the signal was first preemphasized ($\alpha = 0.97$) and then windowed using a Hamming window of size 25ms at 100Hz. We then extracted MFCC's from these windowed features. Deltas and double deltas were appended to the above observation vector. Each phone is modeled using 3 states, and 64 Gaussians per state. We follow the standard practice of building models for 48 different phones and then mapping them down to 39 phones for scoring purposes (Halberstadt and Glass, 1997). The decoding DBN graph is similar to the training graph (Figure 1) except that the variable C_t is removed when decoding. We test on the NIST Core test set (Glass et al., 1996). All results reported in this paper were obtained by computing the string edit (Levenshtein) distance between the hypothesis and the reference. All models in this paper were implemented using the Graphical Models Toolkit (GMTK) (Bilmes and Bartels, 2005).

4.2 VE Based Training and Results

We tested various cases of VE-based training by varying the amount of "dropped" frame labels on either side of the transition (the dropped labels became the unlabeled frames of Figure 2). We did this until there was only one frame left labeled for every phone. Moreover, in each of the above cases, we tested a number of different functions to gener-

ate the VE scores (see section 3). The results of our VE experiments are shown in Figure 3. The curves were obtained by fitting a cubic spline to the points shown in the figure. The phone accuracy (PA) of our baseline system (trained in a fully supervised manner) is 61.4%. If the total number of frames in the training set is N_T, and we drop labels on N frames, the amount of unused data is given by $U = \frac{N}{N_T} * 100$ (the x-axis in the figure). Thus $U = 0\%$ is the fully supervised case, whereas $U = 100\%$ corresponds to using only the sequence information. Dropping the label for one frame on either side of every phone transition yielded $U = 24.5\%$.

It can be seen that in the case of both "uniform over 2 phones" and linear interpolation, the PA actually improves when we drop a small number ($\leq$ 5 frames) of frames on either side of the transition. This seems to suggest that there might be some inherent errors in the frame level labels near the phone transitions. The points on the plot at U=84.7% correspond to using a single labeled frame per phone in every utterance in the training set (average phone length in TIMIT is about 7 frames). The PA of the system using a single label per phone is 60.52%. In this case, we also used a trapezoidal function defined as follows: if $t = t_i$ were the labeled frames for phone p^1, then $f_t(p^1) = 1, t_i - 1 \leq t \leq t_i + 1$, and a linear interpolation function for the other values t during the transition to generate the VE weights. This system yielded a PA of 61.29% (baseline accuracy 61.4%). We should highlight that even though this system used only 15.3% of the labels used by the baseline, the results were similar! The figure also shows the PA of the system that used only the sequence information was about 53% (compare against baseline accuracy of 61.4%). This lends evidence to the claim that training recognizers using data with time segmentation information can lead to improved performance.

Given the procedure we used to drop the frames around transitions, the single labeled frame for every phone is usually located on or around the midpoint of the phone. This however cannot be guaranteed if a transcriber is asked to randomly label one frame per phone. To simulate such a situation, we randomly choose one frame to be labeled for every phone in the utterance. We then trained this system using the "uniform over 2 phones" technique and tested it on the NIST core test set. This experiment was repeated 10 times, and the PA averaged over the 10 trails was found to be 60.5% (standard deviation 0.402), thus showing the robustness of our technique even for less carefully labeled data.

5 Discussion

In this paper we have shown how VE can be used to train a TIMIT phone recognition system using partially labeled data. The performance of this system is not significantly worse than the baseline that makes use of all the labels. Further, though this method of data transcription is only slightly more time consuming that sequence labeling, it yeilds significant gains in performance (53% v/s 60.5%). The results also show that even in the presence of fully labaled data, allowing for uncertainity at the transitions during training can be beneficial for ASR performance. It should however be pointed out that while phone recognition accuracy is not always a good predictor of word accuracy, we still expect that our method will ultimately generalize to word accuracy as well, assuming we have access to a corpus where at least one frame of each word has been labeled with the word identity. This work was supported by an ONR MURI grant, No. N000140510388.

References

[Bilmes and Bartels2005] J. Bilmes and C. Bartels. 2005. Graphical model architectures for speech recognition. *IEEE Signal Processing Magazine*, 22(5):89–100, September.

[Bilmes2004] J. Bilmes. 2004. On soft evidence in Bayesian networks. Technical Report UWEETR-2004-0016, University of Washington, Dept. of EE.

[Glass et al.1996] J. Glass, J. Chang, and M. McCandless. 1996. A probabilistic framework for feature-based speech recognition. In *Proc. ICSLP '96*, volume 4, Philadelphia, PA.

[Greenberg1995] S Greenberg. 1995. The Switchboard transcription project. Technical report, The Johns Hopkins University (CLSP) Summer Research Workshop.

[Halberstadt and Glass1997] A. K. Halberstadt and J. R. Glass. 1997. Heterogeneous acoustic measurements for phonetic classification. In *Proc. Eurospeech '97*, pages 401–404, Rhodes, Greece.

[Lamel et al.2002] L. Lamel, J. Gauvian, and G. Adda. 2002. Lightly supervised and unsupervised acoustic model training. *Computer Speech and Language*.

[Pearl1988] J. Pearl. 1988. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, Inc.

[Subramanya et al.2006] A. Subramanya, A. Raj, J. Bilmes, and D. Fox. 2006. Recognizing activities and spatial context using wearable sensors. In *Proc. of the Conference on Uncertainty in Artificial Intelligence (UAI)*.

[Wang and Fosler-Lussier2006] Y. Wang and E. Fosler-Lussier. 2006. Integrating phonetic boundary discrimination explicity into HMM systems. In *Proc. of the Interspeech*.