iVector-based Acoustic Data Selection

Olivier Siohan, Michiel Bacchiani

Google Inc., New York

siohan@google.com, michiel@google.com

Abstract

This paper presents a data selection approach where spoken utterances are selected in a sequential fashion from a large out-of-domain data set to match the utterance distribution of an in-domain data set. We propose to represent each utterance by its iVector [1], a low dimensional vector indicating the coordinate of that utterance in a subspace acoustic model. We show that the distribution of iVectors can characterize a data set and enables distinguishing subsets of utterances from different domains. Last, we present experimental speech recognition results based on a system trained on a data set constructed by the proposed algorithm and a comparison with random data selection.

Index Terms: speech recognition, data selection, acoustic modeling

1. Introduction

The continuous growth of speech applications deployed on a large variety of devices and languages requires a significant effort to collect the transcribed speech corpora needed to build such systems. This has traditionally been an expensive and time consuming procedure, which led to the development of DataHound [2], a data collection application running on Android mobile devices to record spoken utterances. With such a tool, we have been able to collect speech corpora in over 50 languages, which were then used to train the acoustic models for Google Voice Search [3].

While DataHound is a very effective tool to bootstrap a system, the data it collects is not fully representative of the target application real usage because of the use of pre-specified prompts and a limited coverage in terms of speakers and environmental conditions. Therefore, it is typically beneficial to re-train the acoustic models from the anonymized spoken utterances extracted from the application logs.

However, given the large number of languages and applications that are supported at Google, manual transcription of those audio logs scales poorly. For those reasons, we rely heavily on unsupervised training techniques where the hypothesized transcripts are used as reference labels to train the systems [4, 5]. Simple heuristics such as discarding very short utterances and utterances with low confidence are used to filter out sentences presumed to have errorful transcripts. This bears some similarity with the data selection procedures used in active learning [6, 7] which rely on confidence measures [8].

It was shown however that when the confidence annotator is constructed from side information provided by the recognition model, the data selection procedure could improperly sample the joint acoustic and label space [9]. In the case of an application like Voice Search, high prior queries have good coverage in the original training corpus and are more likely to be recognized with high confidence. This could lead to a biased sampling which would then reduce the acoustic contextual diversity of the selected data set. Over time, this could reduce performance as each new training set would slowly drift towards focusing on high prior queries. In addition, modern acoustic model training techniques such as Deep Neural Network systems [10] are computationally expensive. Hence, crafting a training set to get the best possible performance with the smallest amount of training data is especially attractive.

For those reasons, we propose in this paper a data selection approach which enforces that the distribution of the selected data matches the distribution of the target domain data. In the next section, we describe the basic principle of the data selection procedure. Next, we describe a specific implementation of that procedure which uses the distribution of iVectors [1], a vector of fixed dimensionality characterizing an utterance, to represent a corpus of utterances. Last, we present and discuss experimental results.

2. Entropy-based data selection

2.1. Principle

When a new speech-enabled application is deployed, the audio data extracted from its logs is the best data to use to re-train the system since it matches exactly the application domain. Let us call $P(A, W)$ the joint probability of an utterance from the application logs, represented by its sequence of acoustic feature vectors A and its word sequence W.

A data selection procedure based on keeping high-confidence utterances will typically lead to constructing a data set having an utterance distribution $Q(A, W)$ which will differ from $P(A, W)$. Such a data set is no longer an optimal training set for the application. This paper proposes a data selection approach that enforces that the distribution Q will match P based on a relative entropy criterion.

This algorithm is described in Algorithm 1 and operates as follows. Let $D_{KL}(P \| Q)$ be the Kullback-Leibler divergence [11], also known as relative entropy, between the distribution P and Q. Let S be a data set consisting of already selected utterances and let us denote its distribution by $Q_S(A, W)$. We propose an iterative data selection algorithm where an utterance u will be added to the selected set S, if and only if it adding it to S does not increase the KL divergence $D_{KL}(P \| Q)$. Note that by design, the divergence will monotonically decrease as the size of the selected set increases. As the distributions P and Q get closer and closer, fewer and fewer utterances get selected, as will be illustrated in Section 4.3.2.

Ideally, such an approach should be applied to the joint distribution $P(A, W)$ of the acoustic and word sequences. However, practical consideration make this a difficult task: the distributions have to be updated on a per-utterance basis and the derivation of the KL divergence should be computationally tractable and efficient.

In [12], this approach was applied to text data selection using only $P(W)$. Given a small target-domain data set and a large out-of-domain data set, the authors were able to select subsets of sentences matching the n-gram distribution of the target-domain data.

In this paper we propose to apply this data selection procedure using $P(A \mid W)$. Namely, given a reference transcript (in our case provided by an ASR system), we would like to characterize the sequence of feature vectors corresponding to an utterance and derive its corresponding distribution in a computationally efficient manner. For that purpose, we suggest to characterize each utterance by its identity vector, or iVector, as proposed in [1].

Algorithm 1: Relative-entropy data selection algorithm

```
Input: A reference distribution $P$; an initial set of
       already selected utterances $S$; a set of
       out-of-domain utterances $U$
Output: The selected data set $S$
Estimate the distribution $Q_S$
$D \leftarrow D_{KL}(P \| Q_S)$
for each utterance $u$ do
    Estimate $Q_{S \cup u}$
    $D' \leftarrow D_{KL}(P \| Q_{S \cup u})$
    if $D' < D$ then
        $S \leftarrow S \cup u$
        $D \leftarrow D'$
return $S$
```

3. iVector Model

3.1. Principle

Modeling a speech signal for speech or speaker recognition typically involves large models such as Hidden Markov Models (HMM) or Gaussian mixture models (GMM) to represent the distribution of feature vectors derived from some form of short term spectral analysis. In some applications such as speaker adaptation or speaker identification, it is desirable to re-estimate those models from a small amount of data such as a few sentences. This often requires using modeling techniques which decompose at training time the acoustic space into a small number of subspaces. Once the subspace bases have been estimated, it is then possible to reliably estimate the coordinates of a short utterance along those subspaces. This is the driving principle behind fast adaptation techniques such as Cluster Adaptive Training (CAT) [13] or Eigenvoices [14]. In the speaker recognition community, utterances from a given speaker can be pooled together to estimate the coordinate of that speaker in the subspaces. It was shown that those coordinates can be used to characterize the speaker [1] and were then dubbed identity vector, or iVector for short.

Recently, iVectors were successfully used for fast speaker adaptation [15]. Similar to Eigenvoices, the means of a speaker-independent (SI) HMM-based system are stacked together to form a supervector M_0 of dimension Nd, where N refers to the number of Gaussian densities in the model and d is the feature vector dimension. Given an utterance i, the adapted mean supervector $M(i)$ is obtained according to the following equation:

$$M(i) = M_0 + Vy(i) \tag{1}$$

where V is the so called total variability matrix of size $Nd \times R$ estimated at training time from a large set of utterances following the algorithm detailed in [1], and $y(i)$ is the iVector of dimension R (the number of bases) corresponding to utterance i, i.e. the coordinates of utterance i in the subspace spanned by the columns of V. Since $R \ll Nd$, the iVector $y(i)$ can be reliably estimated from a single utterance, leading to an adapted recognition model.

3.2. iVector distribution as a data set characterization

In this paper, we propose to characterize each utterance by its iVector. Given an existing iVector factor model, i.e. the supervector M_0 and variability matrix V, it is possible to estimate the iVector of each utterance given its transcript, similar to what was done in the iVector-based speaker adaptation approach in [15]. Regardless of its duration, any utterance can then be represented by a vector of fixed dimension R.

Note that the iVector of an utterance is an indication of the match of that utterance to the reference SI model. From Eq. 1, a null iVector indicates that the adapted model means corresponds to the SI model means M_0, while an iVector with a large magnitude would indicate an acoustic mismatch between the utterance and the SI model.

An entire data set can then be described by the distribution of its iVectors, which we use as a proxy for $P(A \mid W)$. Note that iVectors are representative of the acoustic variabilities related to speakers, dialects and channels, rather than what is being said. As demonstrated in [1], the iVector model follows a Bayesian framework and operates under the assumption that the iVector prior distribution is Normal. For that reason, we represent in this work the iVector distribution using a Normal distribution with full covariance matrix. This greatly simplifies the derivation of the KL divergence between two iVector distributions since it then admits a closed form solution. In addition, since a Normal distribution admits sufficient statistics of fixed size, the distribution $Q_{S \in u}$ used in Algorithm 1 can be efficiently re-estimated for each candidate utterance u by storing the statistics $\sum_u y(u)$ and $\sum_u y(u)y(u)'$ derived from each utterance iVector $y(u)$.

4. Experiments and Results

4.1. Databases

All our experiments were conducted using a database of utterances extracted from the logs of the Voice Search application and the Voice-based Input Method running on Android phones and tablets for the Russian language. In accordance with our data retention policies, the provenance of all those recordings were anonymized.

In this paper we treat Voice Search (VS) as our target domain while the Voice Input Method (IME), which is typically used to fill out text fields in applications like SMS dictation, is our out-of-domain data source from which we seek to collect a subset of utterances matching the target domain. Depending on the experiments, we used various data sets, including a VS and an IME test sets, each of them consisting of about 20 hours of speech, as well as an IME data set used for data selection and consisting of about 600 hours of high-confidence utterances selected from the logs. A last data set consisting of about 150 hours of data, a mix of VS and IME recordings, was used to train the iVector factor model, which was used to estimate the iVector of each utterance on all the other data sets.

4.2. Clustering using iVectors

In a first set of experiments, we investigated whether iVectors could characterize our VS and IME test sets and provide sufficient information to distinguish the 2 data sets apart. We first estimated an iVector factor model (the V matrix in Eq. 1) from the 150 hours IME+VS data set. To train that model, we started from a small initial speaker-independent and context-independent HMM-based system of 1680 Gaussian mixture components and estimated a factor model consisting of 32 bases, i.e. the iVector dimension are 32. We constructed a total of 8 random subsets of data, the first 4, denoted $\mathrm{VS}_1, \ldots \mathrm{VS}_4$, consisting exclusively of VS utterances and the last 4, $\mathrm{IME}_1, \ldots \mathrm{IME}_4$, of IME utterances. We enforced all those data sets to be disjoint.

We then computed the iVector of all utterances in those subsets and estimated for each subset its corresponding iVector distribution. From those distributions, we constructed an 8×8 matrix holding the KL divergence between all subset pairs. The objective of that experiment was to validate whether the KL divergence between iVector distributions would enable clustering all VS subsets together, and all IME subsets together. This result is reported in Table 1 using subsets holding 1 hour of data each. The first 4 dimensions of the KL divergence matrix refer to the 4 IME subsets, the next 4 dimensions to the VS subsets. This matrix confirms that the within-domain distances are always smaller than the cross-domain distances, resulting in a perfect clustering of the subsets along their respective domain.

Table 1: *Matrix of KL divergences between iVector distributions estimated on 8 different subsets of 1 hour of data each. The first 4 dimensions refer to IME subsets, the next 4 to VS subsets.*

0.00	0.57	0.61	0.61	1.45	1.34	1.45	1.40
0.57	0.00	0.59	0.60	1.42	1.36	1.46	1.38
0.63	0.62	0.00	0.64	1.42	1.28	1.44	1.35
0.62	0.63	0.65	0.00	1.39	1.32	1.39	1.38
1.52	1.55	1.40	1.36	0.00	0.79	0.77	0.70
1.49	1.53	1.33	1.34	0.84	0.00	0.79	0.78
1.48	1.49	1.37	1.29	0.79	0.77	0.00	0.74
1.47	1.49	1.33	1.35	0.70	0.76	0.76	0.00

We then investigated the minimum amount of data required before no longer being able to cluster the subsets into their corresponding domains. Table 2 shows the same KL divergence matrix as before, this time constructed from subsets holding only 10 min of data each. One can observe that the clustering is no longer perfect: some of the within-domain KL divergences are larger that the cross-domain ones. This illustrates that in order to bootstrap the data selection, we will need to initialize the selected data set using at least 10 min of data.

Table 2: *Matrix of KL divergences between iVector distributions, similar to Table 1 but using 10 min long subsets*

0.00	3.62	4.34	4.42	5.71	5.60	5.99	5.75
4.16	0.00	3.91	4.54	6.63	6.17	6.45	6.55
5.11	4.09	0.00	4.78	7.30	6.29	7.05	7.61
4.90	4.63	4.61	0.00	6.61	5.77	6.24	6.96
5.32	5.74	5.58	5.53	0.00	5.04	5.92	5.62
6.03	5.83	6.25	5.87	5.97	0.00	5.60	6.38
5.23	5.35	5.54	5.02	6.10	5.03	0.00	6.24
5.86	5.92	5.86	6.04	5.58	5.45	6.10	0.00

4.3. Data selection using iVectors

4.3.1. Evaluation on artificial mini-batches

In the next set of experiments, we evaluated the proposed algorithm by running the data selection on mini-batches of utterances. That is, instead of adding a single candidate utterance to the set of already selected utterances, we consider a mini-batch of M utterances at a time. If this does not result in increasing the KL divergence, those utterances are added to the selected set. We evaluated this approach on an artificially constructed data set consisting of alternated batches of $M = 150$ utterances of VS and IME data (about 10 min of data per mini-batch). By design, the input data is then made of half IME and half VS data. The target domain is defined by the VS test. As a result, we expect that the data selection should favor selecting VS data over IME. Indeed, the experiment showed that the selected data set ended-up with 71% of VS data, illustrating the effectiveness of the selection approach.

4.3.2. Data selection rate

Next we evaluated the data selection rate, or how fast the selected data set grows as a function of the amount of input data. For this experiment, the selection operates on one utterance at a time and the target domain is set to VS. The “already selected” data set is initialized by randomly sampling 10 min of data from the VS test set. The input data consists of a sequence of 100k utterances from a random mix of IME and VS utterances. Figure 1 represents the number of selected utterances as a function of the number of input utterances. It shows that the data selection procedure saturates after processing about 40K utterances. This implies that in its current form, the data selection algorithm cannot construct a very large selected data set because ultimately the distributions P and Q of the target and the selected iVectors get very close to each other and no utterance get selected any more. For that reason, we adopted an approach where the input set of utterances is split into multiple subsets of

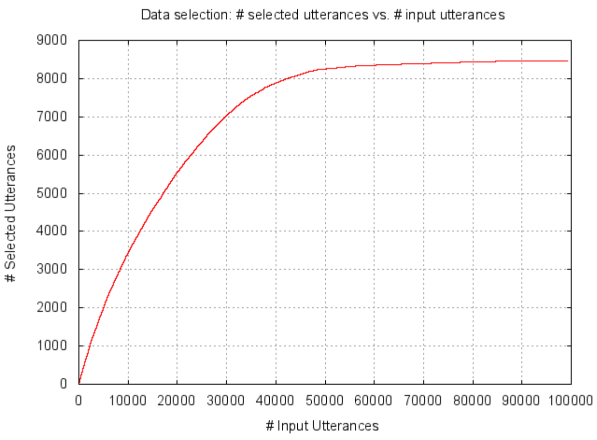

Figure 1: *Number of selected utterances as a function of the number of input utterances.*

40k utterances each and the data selection is run independently on each subset. This led to a selection procedure that retained about 1/5th of the input data.

4.3.3. Recognition experiments

In the next set of experiments, we trained an iVector factor model with 128 bases and ran the data selection procedure selecting one utterance at a time using VS as a target domain. The selection data set consisted of 600 hours of IME data and the algorithm ended-up selecting a total of 150 hours of data. From that selected data set we extracted the first 25 hours, 50 hours, 75 hours, 100 hours and 125 hours of data to construct multiple training sets. We also constructed random subsets of equivalent size from the original 600 hours set. For each amount of data, we then ended-up with 2 training sets: the first one, randomly selected from the 600 hours data set, the second one, selected using the proposed data selection approach.

We then built a speaker independent HMM-system based on a scaled-down version of our Voice Search training procedure [16] using LDA-based feature vectors and boosted-MMI training [17]. Results are available in Figure 2 and are given in terms of word error rates (WER). The system trained on the data set selected by the proposed algorithm outperforms random data selection for all training set sizes. When using a 25-hours training set, the word error rate reduction is 1% absolute; it is 0.6% on the 75-hours training set, and 0.3% on the 125-hours set.

5. Conclusions

We proposed a sequential data selection approach designed to construct a training set matching a desired in-domain utterance distribution from an out-of-domain data set. The selection algorithm is based on a relative entropy criterion and the distributions are defined as the distributions of the iVector associated to each utterance. We have shown that when using Normal distributions with full covariance matrices, it is possible to characterize a data set and cluster data subsets based on their respective domains. Recognition experiments have shown that the proposed approach outperforms random selection of a training set.

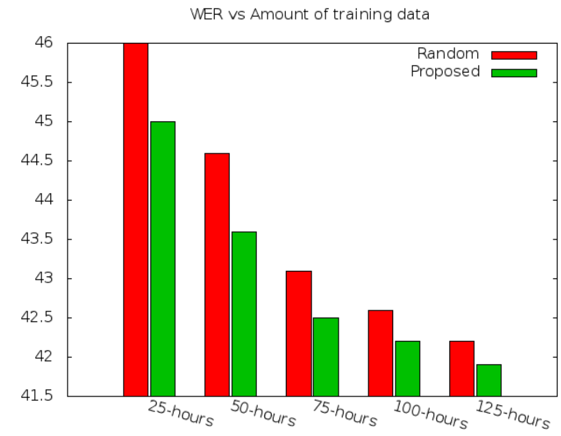

Figure 2: *WER comparing Random data selection with the Proposed approach for various amount of training data.*

6. References

[1] P. Kenny, G. Boulianne, and P. Dumouchel, “Eigenvoice modeling with sparse training data,” *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 3, pp. 345–354, 2005.

[2] T. Hughes, K. Nakajima, L. Ha, A. Vasu, P. Moreno, and M. LeBeau, “Building transcribed speech corpora quickly and cheaply for many languages,” in *Conference of the International Speech Communication Association (INTERSPEECH)*, Makuhari, Japan, 2010, pp. 1914–1917.

[3] “Google voice search,” http://www.google.com/mobile/voice-search.

[4] L. Lamel, J.-L. Gauvain, and G. Adda, “Unsupervised acoustic model training,” in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Orlando, USA, 2002.

[5] K. Yu, M. Gales, L. Wang, and P. C. Woodland, “Unsupervised training and directed manual transcription for LVCSR,” *Speech Communication*, vol. 52, no. 7–8, pp. 652–663, 2010.

[6] G. Riccardi and D. Hakkani-Tur, “Active learning: theory and applications to automatic speech recognition,” *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 4, pp. 504–511, 2005.

[7] N. Itoh, T. N. Sainath, D.-N. Jiang, J. Zhou, and B. Ramabhadran, “N-best entropy based data selection for acoustic modeling,” in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2012, pp. 4133–4136.

[8] H. Jiang, “Confidence measures for speech recognition: A survey,” *Speech Communication*, vol. 45, no. 4, pp. 455–470, 2005.

[9] R. Zhang and A. I. Rudnicky, “A new data selection approach for semi-supervised acoustic modeling,” in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2006.

[10] G. Hinton, L. Deng, D. Yu, G. Dahl, A. rahman Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, and B. K. Tara Sainath and, “Deep neural networks for acoustic modeling in speech recognition,” *IEEE Signal Processing Magazine*, vol. 28, no. 6, pp. 82–97, nov 2012.

[11] S. Kullback and R. A. Leibler, “On information and sufficiency,” *Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[12] A. Sethy, P. G. Georgiou, B. Ramabhadran, and S. S. Narayanan, “An iterative relative entropy minimization-based data selection approach for n-gram model adaptation,” *IEEE Transactions on Audio, Speech & Language Processing*, vol. 17, no. 1, pp. 13–23, 2009.

[13] M. J. F. Gales, “Cluster adaptive training of hidden markov models,” *IEEE Transactions on Speech and Audio Processing*, vol. 8, no. 4, pp. 417–428, Jul.

[14] R. Kuhn, P. Nguyen, J.-C. Junqua, L. Goldwasser, N. Niedzielski, S. Fincke, K. Field, and M. Contolini, “Eigenvoices for speaker adaptation,” in *In proceedings of International Conference on Speech and Language Processing (ICSLP)*, 1998, pp. 1771–1774.

[15] M. Bacchiani, “Rapid adaptation for mobile speech applications,” in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Vancouver, Canada, 2013.

[16] J. Schalkwyk, D. Beeferman, F. Beaufays, B. Byrne, C. Chelba, M. Cohen, M. Kamvar, and B. Strope, “Your word is my command: Google search by voice: A case study,” in *Advances in Speech Recognition*, A. Neustein, Ed. Springer US, 2010, pp. 61–90.

[17] D. Povey, D. Kanevsky, B. Kingsbury, B. Ramabhadran, G. A. Saon, and K. Visweswariah, “Boosted mmi for model and feature-space discriminative training,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2008.