UNSUPERVISED TRAINING ON A LARGE AMOUNT OF ARABIC BROADCAST NEWS DATA

Jeff Ma, Spyros Matsoukas

BBN Technologies, 10 Moulton Street, Cambridge, MA, USA

ABSTRACT

The unsupervised training we carried out on the 1,858-hour untranscribed Arabic Broadcast News (BN) data yields a sizable gain. However, this gain is only about half of that achieved on the 1,900-hour English BN data. This paper presents our efforts that aim at enlarging the gain on the Arabic data. These efforts include a design of an explicit hypothesis-confidence-estimating method for the data selection, use of new features and neural networks (NN) to improve hypothesis-confidence estimation, and alleviation of the over-fitting problem existing in the estimation. Our experiments show that both the explicit hypothesis-confidence-estimating method and the use of new features improve the estimation and render the unsupervised training extra gains; the use of neural networks doesn't significantly improve the confidence estimation; the alleviation of the over-fitting problem is not significant enough to decrease the word error rate (WER). This paper also presents improvements of unsupervised training we conducted on a morpheme-based Arabic system and on models trained with maximum mutual information (MMI) criterion.

Index Terms—speech recognition, unsupervised training, confidence estimation, Arabic broadcast news

1. INTRODUCTION

In paper [1], we reported the results of the unsupervised training we carried out on both the 1,900-hour English BN data and the 1,858-hour Arabic BN data. Although the gain produced by the unsupervised training on the Arabic data is substantial, it is much smaller than the gain observed on the English data. As part of our efforts to find out the reason, we manually inspected 6 episodes randomly selected from the Arabic data, and the inspection revealed that only less than half (48%) of the 6-episode data is pure news speech, and the remaining is either pure music or speech with music background or drama dialogues. The high percentage of non-news and noisy news data was out of our expectation, and we thought that it is the major factor resulting in the relatively poor performance on the Arabic data. We then started to focus on improving the data-selecting algorithm, hoping to exclude as much noisy data as possible from the training. As our initial efforts in this direction, in paper [1] we implemented the incremental training method and the language model perplexity-based episode-removing method. Both methods helped the unsupervised training to yield extra gains, but the total gain on the Arabic data was still only about half of that observed on the English data. Since then, we have worked more in this direction. We also ran the MMI training on the unsupervised data to see the effect of unsupervised training on MMI models.

We organize this paper as follows: Section 2 presents an explicit hypothesis-confidence-estimating method we designed to improve the data selection; Section 3 addresses the other efforts we took to improve the confidence estimation, including use of new features, use of NNs, and alleviation of the over-fitting problem; Section 4 reports MMI training experiments we carried out on the unsupervised data; Section 5 concludes this paper.

2. AN EXPLICIT ESTIMATION METHOD FOR HYPOTHESIS CONFIDENCE

In our previous work [1], we showed the benefit of the use of confidence scores in the data selection. Therein, the confidence score of a hypothesis was estimated implicitly as a weighted sum of its word confidence scores,

$$cnf(h) = \sum_{i-1}^{N} cnf(W_i) \bullet len(W_i) / \sum_{i=1}^{N} len(W_i) \qquad (1)$$

where N is the number of words in hypothesis h, and $cnf(W_i)$ and $len(W_i)$ are the confidence and duration (length) of word W_i, respectively. The score computed in Eqn.1 is an approximation to the correctness of a hypothesis. The higher the average word confidence score, the more likely the hypothesis is correct. However, it is possible that two hypotheses have the same average word confidence scores but differ significantly in terms of accuracies.

It would be preferable to explicitly estimate confidence scores for hypotheses. We have been using the generalized linear model (GLM) [3] to estimate confidence in various applications [4]. To train the GLM for the explicit estimation of hypothesis confidence, the correctness of each hypothesis must be defined, and it is usually defined as a binary value – either 1 if correct or 0 if wrong. Normally, we would consider one hypothesis as correct if its accuracy is 100% (or 1). If we define the hypothesis correctness in this way, few hypotheses, however, in the whole test set will be correct, because the sentence accuracies are significantly low even for recognition systems that have WERs below 10%. The GLM, hence, will be trained to select little data, and the unsupervised training won't work properly. In fact, for the unsupervised training it is not crucial that the automatically transcribed data that is selected for the acoustic training must have no errors. Our previous experiments on the English BN data show that the unsupervised training without data filtering (blindly selects all data) performs only slightly worse than the lightly supervised training [1]. Therefore, we choose an *accuracy threshold, x,* and consider one hypothesis safe to add into the training if its accuracy is above the threshold x ($0.5<x<1$). Then, we define one hypothesis as correct if its accuracy is above the threshold and as wrong if not. In this way, there will be more correct samples in the GLM model

training if $x < 1$, and the estimated scores reflect the confidence that the accuracies of hypotheses are in the region of [*x,* 1]. Similarly to the implicit method (Eqn.1), we use a *confidence threshold* to select data. One hypothesis is selected if its confidence estimated in the new explicit way is higher than the confidence threshold. So, there are two thresholds used in the explicit method, the accuracy threshold for the model training and the confidence threshold for the data selection. The best setting of the two thresholds needs to be tuned.

Besides defining the correctness, we also need to generate hypothesis-related features for the GLM training. We generated 21 hypothesis-related features for each hypothesis, such as posterior probability, normalized acoustic and language model (LM) scores, and number of words. The GLM training is then to learn the mapping between the hypotheses' features and their correctness values (1 or 0).

Following the work done in [1], we carried out all unsupervised training experiments on the 1,858-hour Arabic data, out of which 1,570 hours remain after the automatic audio segmentation. The baseline model was trained on the 150-hour seed data. Results reported in all the following experiments were obtained using ML speaker-independent models (MI-SI), unless otherwise specified. The test set "h4ad04" used in some experiments denotes the Arabic development set released by LDC for the EARS RT04 evaluation.

Table 1 lists the comparison between the explicit and the implicit methods. The column "Train data" indicates the seed data plus the data selected by the methods. For the implicit method, the confidence threshold of 0.85 was found in [1] to produce the best WER reduction. For the explicit method, we set the accuracy threshold to 0.7 and the confidence threshold to 0.5. In this comparison, we didn't carefully tune the two thresholds for the explicit method.

Estimation method	Conf. thres.	Train data (#hours)	Mode size (#Gauss)	Un-adapted WER (%)
Implicit	0.85	150+488	388K	16.3
Explicit	0.5	150+817	551K	16.0

Table 1. Comparison of the explicit and implicit estimation of hypothesis confidence used in the unsupervised training (WERs measured on h4ad04 test set).

Compared to the best result we obtained by using the implicit method, the explicit method produces 0.3% absolute gain. So, the explicit method outperforms the implicit method. A more thorough comparison of the two methods will be given in the next section.

3. IMPROVING THE CONFIDENCE ESTIMATION

Since the new explicit estimation method outperforms the old implicit method, we next tried to further improve the confidence estimation for the explicit method.

3.1. Use of new features

We noticed that some features, which should be beneficial to the confidence estimation, had not been used in our current confidence estimation. Historically, we had been using features derived from *n*-best lists for the GLM training. No features derived from lattices, which accommodate more information than *n*-best lists, had been used, though our modern decoding sequences output lattices. The size of a lattice is a good indication of the decoding uncertainty, which has a strong correlation with any kind of confidences. So, we added two simple and straight-forward lattice features to the GLM training, number of nodes per word and number of arcs per word. To get the two features for one lattice, we divided the number of nodes and the number of arcs in the lattice by the number of words in the corresponding top-1 hypothesis. Moreover, our earlier experiments using the LM perplexities (PPL) of episodes to exclude data show positive results [1], so we also added the episode LM perplexity as another feature to the GLM training. We trained the GLMs on the "h4ad05" data set, which includes 6 hours of Arabic BN data put together by BBN.

The normalized cross entropy (NCE) is a commonly used metric to measure confidence estimation quality [5]. Table 2 shows effect of these new features on the NCE metric. The first half of the table (2^{nd}-4^{th} rows) shows the effect on the explicit hypothesis-confidence estimation, and the second half (5^{th}-7^{th} rows) shows the effect on the word confidence estimation, which is used in the implicit hypothesis-confidence estimation. The accuracy threshold in the explicit estimation was set to 0.7. The results show that the new features increase the NCE significantly for the hypothesis confidence estimation and slightly for the word confidence estimation. This is reasonable since the new features are more closely correlated with hypotheses than with words. The results also show that the lattice features is more beneficial to the estimation than the PPL feature does.

Confidence type	Use Latt. Feat.	Use PPL feat.	NCE score
Hypothesis	no	no	0.394
Hypothesis	yes	no	0.444
Hypothesis	yes	yes	0.457
Word	no	no	0.326
Word	yes	yes	0.332
Word	yes	yes	0.333

Table 2. Effect of adding the lattice and perplexity features to the GLM training for the confidence measures (the NCE scores measured on the h4ad05 test set)

Since the new features improved the performance of the GLMs, we then re-ran the unsupervised training experiment using the explicit hypothesis-confidence-estimating method as well as the one using the implicit method, but we replaced the GLMs used in the data selection with their corresponding new ones trained with the new features. These new experiments are shown in Table 3.

Method	Accu. thres.	Conf. thres.	Lat. & PPL	Train data (#hours)	Un-adaptd WER
-	-	-	-	150	18.1
Explici	0.7	0.5	no	150+817	16.0
Explici	0.65	0.5	yes	150+770	15.7
Implici	-	0.85	no	150+488	16.3
Implici	-	0.85	yes	150+496	16.1

Table 3. Effect of the new lattice and perplexity features on the performance of the unsupervised training (WER on h4ad04)

To have fair comparisons, we tuned the thresholds for all the cases given in Table 3. For clarity, only the best setting for each

case is listed in the table. The results show that the new features reduce the WER by 0.3% absolute (the 3rd vs. the 4th row) the training that uses the explicit method and by 0.2% absolute (the 5th vs. the 6th row) for the training that uses the implicit method. With the new features in both methods, the explicit method outperforms the implicit method by 0.4% absolute WER reduction (the 4th vs. the 6th row). Compared to our previous work (the 5th row), the development of the explicit estimation method and the use of new features renders the unsupervised training 0.6% extra gain (the 4th row). Compared with the baseline given in the 2nd row, the unsupervised training with the improved data selection (the 4th row) achieves a 2.4% absolute (13% relative) WER reduction.

3.2. Investigation on using neural networks

In Section 3.1, we generated 24 features for each hypothesis. The GLM was trained to learn the mapping between the features and the hypotheses' correctness values. However, we believe that the mapping is highly nonlinear, so we tried NNs to learn the mapping, hoping improving the modeling accuracy.

We trained NNs on the same data – the "h4ad05" data set – as used to train the GLM in the explicit method. This data set has 2059 segments after the automatic audio segmentation. We used the top-1 decoding hypothesis of each segment to train the GLM models in the explicit method, so there were 2059 training samples. We noticed that the trained GLM has a greatly higher NCE score on the development set (or Dev set) than on a validation set (or Val set), so these samples are not enough and cause the trained model to over-fit the training data. We will address this over-fitting issue later. The Val set that we used is called "h4av05" – a 3-hour test set that was set up by BBN as well. We used the Neural Network Toolbox in Matlab to carry out this investigation. We found that among many NN training algorithms available in the Toolbox, the resilient back-propagation (RBP) algorithm and the Bayesian regularization (BR) algorithm perform the best for our case. The BR algorithm, though assumed to be able to generalize well, performed similarly to the RBP, so we list only the performance of the RBP algorithms in Table 4. The pattern, "*m-n-l*", in the "Structure" column of the table indicates that the NN has *m, n* and *l* nodes in its input, hidden and output layers, respectively. The NN has no hidden layer if n=0, and it becomes similar to the GLM.

model	structure	Training Iterations	NCE	
			H4ad05	H4av05
GLM	22-0-1	80	0.475	0.318
NN	22-0-1	100	0.469	0.313
	22-2-1	100	0.491	0.314
	22-3-1	100	0.498	0.300
	22-3-1	50	0.487	0.322
	22-4-1	50	0.500	0.296

Table 4. Comparison of the GLM and the NNs trained with the resilient back-propagation algorithm

From the results shown in Table 4, we first see that the NN performs similarly to the GLM model when it lacks hidden layers. Second, we observe that the NCE score on the dev set – "h4ad05"— can be easily improved by increasing the NN model size, but the NCE score quickly degrades on the Val set – "h4av05". So, the NN training is fragile to the over-fitting problem. With careful tuning, such as the one with the structure "22-3-1" and with 50 iterations, the NN is able to outperform the GLM. But the gain on the NCE metric is minor. So we didn't pursue further in this direction.

3.3. Alleviation of the over-fitting problem

We had realized that the GLM training in the explicit hypothesis-confidence estimation suffers from over-fitting due to lack of training data. We then tried two things to alleviate this problem.

We had been using only the top-1 hypothesis of each segment in the GLM training. To having more training samples, our first effort is to include top-n (n>1) hypotheses in the GLM training.

At this time, BBN had set up a new 6-hour development set – "bnat05" – for the DARPA GALE project. For the purpose of using this unsupervised training in the coming GALE evaluation, we switched to this new dev set in all later experiments. We also switched to a morpheme-based Arabic system [3], which reduces the out-of-vocabulary (OOV) rate substantially. We conjectured the high OOV rate in the word-based Arabic systems might be one factor that deteriorates the unsupervised training. Later, we found that unsupervised training on the morpheme-based produces the similar relative gains on the "h4ad04" test set as the word-based system does, so the high OOV rate doesn't affect the unsupervised training performance. The validation set is still the "h4av05" set.

Top-n	NCE scores		Train data (#hours)	Un-adapted WER
	bnat05	h4av05		
1	0.434	0.384	150+800	19.5
2	0.431	0.397	150+771	19.5
5	0.423	0.400	150+782	-

Table 5. Effect of using top-n hypotheses in the GLM training (The WER measured on the bnat05 data set)

Table 5 lists the experiments using top-n hypotheses. In these experiments the accuracy and confidence thresholds were set to 0.8 and 0.5, respectively. With top-n (2 and 5) hypotheses included in the training, the gap on the NCE between the Dev and the Val sets is reduced, and the NCE score on the Val set is also improved. So, the over-fitting problem is alleviated. However, the unsupervised training using the top-2 hypotheses in the GLM training produces the same WER as the baseline, which uses the top-1 hypothesis. Since the NCE improvement from the top-2 to the top-5 is insignificant, we didn't run unsupervised training for the top-5 case. We also tried to include more than 5 hypotheses, but we didn't see further improvements on the NCE score. This is reasonable, because the similarity among n-best hypotheses for each segment is normally high, and the addition of more such hypotheses doesn't reduce the data sparseness much. This is the reason that we used only the top-1 hypothesis in the first place.

Our second effort was to train the GLM on a larger Dev set. BBN had also set up a larger development set – "bncad05" – by adding two hours of broadcast conversational data to the "bnat05" set. Table 6 shows the effect of using this larger data set. In these experiments, the settings for the accuracy and confidence thresholds are 0.65 and 0.7, respectively. We found that these settings are slightly better than those used in the experiments listed in Table 5. Also, since the use of top-n (n>1) hypotheses doesn't affect the WER performance, we still used only top-1 hypothesis in these experiments. As expected, the larger data set reduces the

gap of the NCE score between the Dev and the Val sets and improves the NCE score on the Val set. So the over-fitting problem is alleviated. But, again, the WER of the unsupervised training is not improved.

Dev set	NCE scores		Train data (#hours)	Un-adapted WER
	Dev set	h4av05		
bnat05	0.510	0.395	150+880	19.4
bncad05	0.486	0.408	150+859	19.4

Table 6. Effect of training the GLM on a larger data set (The WER measured on the bnat05 set)

Both of the two efforts alleviate the over-fitting problem. However, the NCE improvements on the Val set are of insignificance (3~5%). It might be the reason that no improvement has been achieved on the WER performance.

The baseline model trained on the 150-hour data produces a 21.7% WER on the "bnat05" test set at the un-adapted decoding pass. So the relative gain on the "bnat05" test set from the unsupervised training listed in Table 6 is 10.6%. It is smaller than that the 13% gain observed on the "h4ad04" test set at the same decoding pass. We have investigated it. As described in [1], the 1,858-hour un-transcribed Arabic audio data was collected from 6 different sources. All the episodes in the "h4ad04" set are from the 6 sources, but only 8 out of the 12 half-hour episodes in the "bnat05" test set are from the 6 sources. The decomposition of the WER according to episodes reveals that the unsupervised training doesn't decrease the WER at all for the 4 episodes in the "bnat05" that are not from the 6 sources. That is the reason that the unsupervised training produces a smaller gain on the "bnat05" set. In general, it is true that a speech recognition model trained on data from one source performs poorly on data from another source. However, after a similar decomposition of the WER produced by the unsupervised training we carried on the English data, we saw that the dissimilarity among Arabic data sources is larger. It could be another factor that the unsupervised training produces a smaller gain on the Arabic data than on the English data.

4. MMI TRAINING

Recall that the incremental training method in [1] yields a 0.3% extra gain. It could also yield a similar gain if combined with the new explicit confidence-estimating method. We have not done it. Instead, to see how much of the gain remains after the MMI training, we picked the best unsupervised training experiment – the last row of Table 6 – and proceeded to train the "MMI-SI" and "MMI-SAT" models. The "MMI-SI" denotes the speaker-independent (SI) model trained under the MMI criterion, and the "MMI-SAT" denotes the model trained by the speaker-adaptive training (SAT) method under the MMI criterion.

The performance of the different types of models is listed in Table 7. The 2nd and 3rd rows list the performance of the baseline, which was trained on the 150-hour seed data, and the 4th and 5th rows list the performance of the unsupervised training that added 859 hours of unsupervised data to the seed data. First, Comparing the 2nd and 4th rows, one can see that the relative gain (or WER reduction) from the unsupervised training is 10.6% (21.7 →19.4) at the un-adapted decoding pass and shrinks to 8.0% (17.6 → 16.2) after the adaptation. Second, comparing the 3rd and 5th rows, one can see that after the MMI training the relative gain from the unsupervised training is 7.2% (16.7 → 15.5) after the adaptation. This gain is only slightly smaller than the gain (8.0%) observed on the ML models at the same stage. So, the errors existing in the selected data don't hurt the MMI training significantly.

Train data (#hours)	WER on bnat05	
	Unadapted (model type)	Adapted (model type)
150	21.7 (ML-SI)	17.6 (ML-SAT)
150	20.4 (MML-SI)	16.7 (MMI-SAT)
150+859	19.4 (ML-SI)	16.2 (ML-SAT)
150+859	18.6 (MMI-SI)	15.5 (MMI-SAT)

Table 7. Improvements from the unsupervised training after the MMI training and the adapted decoding

5. CONCLUSIONS

We have presented our efforts aiming at enlarging the gain from unsupervised training on the 1,858-hour Arabic BN data. Due to the high percentage of non-news and noisy news data, we focused mainly on improving the data selection. The explicit method we developed for hypothesis confidence estimation outperforms the old implicit method and renders the unsupervised training 0.4% extra WER reduction. The use of the lattice and perplexity features improves the confidence estimation and produces 0.2-0.3% WER reduction. Our investigation shows that there is no significant benefit from the use of neural networks in the confidence estimation because of the over-fitting problem caused by the lack of training data. To alleviate the over-fitting problem in the confidence estimation, both the use of top-n (n>1) hypotheses and the training on a larger Dev set achieve limited successes (3-5% improvements on the NCE metric) but have not contributed any WER reduction.

After these efforts, the unsupervised training on the 1,858-hour Arabic data corpus yields a 13% relative gain on the h4ad04 test set. This gain is still smaller than the 21.6% gain achieved on the 1,900-hour English data. We believe the major reason is the high percentage of non-news and noisy news data existing in the Arabic corpus. Other reasons could be the larger dissimilarity among Arabic sources, relatively poorer performance of the baseline model used to decode the data, etc.

6. ACKNOWLEDGEMENTS

This work was supported under the GALE program of the Defense Advanced Research Projects Agency, Contract No. HR0011-06-C-0022.

7. REFERENCES

[1] J. Ma, S. Matsoukas, etc, "Unsupervised Training on Large Amounts of Broadcast News Data", ICASSP 2006, Vol. 3, pp. 1056-1059, May 2006.

[2] B. Xiang, K. Nugyuen, etc. "Morphological Decomposition for Arabic Broadcast News Transcription", ICASSP 2006, Vol. 1, pp. 1089-1092, May 2006.

[3] T. Hastie and R. Tibshirani. *Generalized Additive Models.* Chapman and Hall, London, 1990.

[4] M. Siu, H. Gish, F. Richardson, "Improved estimation, evaluation and applications of confidence measures for speech recognition", Eurospeech 97.

[5]http://www.nist.gov/speech/tests/ctr/h5_2000/h5-2000v1.3.htm.