SEMI-SUPERVISED DISCRIMINATIVE LANGUAGE MODELING FOR TURKISH ASR

A. Çelebi[a], H. Sak[a], E. Dikici[a], M. Saraçlar[a],
M. Lehr[b], E. Prud'hommeaux[b], P. Xu[c], N. Glenn[d], D. Karakos[c], S. Khudanpur[c], B. Roark[b], K. Sagae[e], I. Shafran[b],
D. Bikel[f], C. Callison-Burch[c], Y. Cao[c], K. Hall[f], E. Hasler[g], P. Koehn[g], A. Lopez[c], M. Post[c], D. Riley[h]

[a]Boğaziçi U., [b]OHSU, [c]JHU, [d]BYU, [e]USC, [f]Google, [g]U. of Edinburgh, [h]U. of Rochester

ABSTRACT

We present our work on semi-supervised learning of discriminative language models where the negative examples for sentences in a text corpus are generated using confusion models for Turkish at various granularities, specifically, word, sub-word, syllable and phone levels. We experiment with different language models and various sampling strategies to select competing hypotheses for training with a variant of the perceptron algorithm. We find that morph-based confusion models with a sample selection strategy aiming to match the error distribution of the baseline ASR system gives the best performance. We also observe that substituting half of the supervised training examples with those obtained in a semi-supervised manner gives similar results.

Index Terms: Discriminative Training, Semi-supervised Learning, Language Modeling, Confusion Modeling

1. INTRODUCTION

In automatic speech recognition (ASR), language models assign weights to word sequences to discriminate between acoustically similar sequences. Discriminative training of language models has been shown to improve the speech recognition accuracy by resolving acoustic confusions more effectively [1]. In discriminative language modeling (DLM), a speech recognizer is employed to generate a set of competing hypotheses for an utterance. Given the correct transcription of an utterance and the set of competing hypotheses (confusion set), discriminative learning techniques can be applied to make use of positive and negative examples to reward features in the correct transcription and penalize features in the competing hypotheses.

This research is supported in part by TUBITAK under Project number 109E142 and by the Turkish State Planning Organization (DPT) under the TAM Project number 2007K120610. Murat Saraçlar is supported by the TUBA GEBIP Award. Part of this research was done at the JHU CLSP Summer Workshop 2011. This work is also partially supported by NSF Grants #IIS-0963898 and #IIS-0964102.

However, this approach requires a large amount of transcribed speech data. Several approaches have been proposed to overcome the necessity of supervised learning for DLM. For instance, Xu et al. propose a self-supervised discriminative training method, in which an exponential language model is trained using only untranscribed speech and a large text corpus [2]. First, *cohorts* for words w are determined from the first-pass ASR output lattices for untranscribed speech utterances. The discriminative training is based on maximization of the likelihood ratio between the words w in the text corpus and their cohorts. In another work, Kurata et al. propose to generate the probable n-best lists that an ASR system may possibly output, for an input hypothetical utterance given a word sequence [3]. They call this process *Pseudo-ASR* since they use phoneme similarities estimated from an acoustic model to generate the competing hypotheses. The discriminative training of the model is based on the generalized probabilistic descent (GPD) algorithm and more recently they applied discriminative reranking using the perceptron algorithm [4]. In another study, Tan et al. propose a system for channel modeling of ASR for simulating the ASR corruption using a phrase-based machine translation system trained between the reference phoneme and output phoneme sequences from a phoneme recognizer [5]. Jyothi et al. have also modeled the phonetic confusions using a confusion matrix that takes into account word-based phone confusion log likelihoods and distances between the phonetic acoustic models [6]. The confusion matrix is used to generate confusable word graphs for training a discriminative language model using the perceptron algorithm.

In this paper, we investigate various confusion models to employ in semi-supervised learning of discriminative reranking models for Turkish. We experiment with confusion models at various granularities, namely, word, sub-word (morph), syllable and phone levels, in order to address the highly productive morphology of Turkish. However, these can also be applied to other languages including morphologically simpler languages.

input set of training examples $\{y_i : 1 \leq i \leq N\}$
input number of iterations T
$\bar{\alpha} = 0, \bar{\alpha}_{sum} = 0$
for $t = 1 \ldots T, i = 1 \ldots N$ **do**
  $z_i = \text{argmax}_{z \in \mathbf{GEN}(y_i)} \bar{\alpha} \cdot \mathbf{\Phi}(z)$
  $\bar{\alpha} = \bar{\alpha} + \Delta(y_i, z_i)(\mathbf{\Phi}(y_i) - \mathbf{\Phi}(z_i))$
  $\bar{\alpha}_{sum} = \bar{\alpha}_{sum} + \bar{\alpha}$
end for
return $\bar{\alpha}_{avg} = \bar{\alpha}_{sum}/(NT)$

Fig. 1. The WER-sensitive perceptron algorithm

The confusion models are trained over transcribed speech which is generally available for acoustic modeling. However, for discriminative training, the negative examples corresponding to sentences from a text corpus are generated using confusion and language models. We experiment with various language models to choose among possible confusions. For discriminative training of models, we use a variant of the perceptron algorithm, the *WER-sensitive perceptron* which has been shown to perform better for reranking ASR hypotheses [7]. Rather than just using the top simulated hypotheses output by the model, we also experiment with different sampling strategies. The strategy that simulates the word error distribution in the ASR output gives the best improvement.

The paper is organized as follows. In section 2, we introduce the discriminative training of reranking models. Following that, in section 3, we describe the system. In section 4, we present the experimental setup and the results, before concluding with section 5.

2. SEMI-SUPERVISED DISCRIMINATIVE RERANKING

We use a variant of the averaged perceptron algorithm, the *WER-sensitive perceptron* [7], to estimate the parameters of the reranking model discriminatively, as shown in Figure 1. The standard perceptron algorithm is trained to minimize the number of misclassifications, whereas the WER-sensitive perceptron algorithm is trained to minimize a loss function which is defined using edit distances of hypotheses to the reference transcription. Hence, discriminative training of feature parameters runs on the criteria of minimizing the word error rate (WER) rather than the number of misclassifications.

The algorithm estimates a parameter vector $\bar{\alpha} \in \mathbb{R}^d$ using a set of training examples. In the conventional DLM, the function $\mathbf{GEN}$ generates a set of hypotheses for an acoustic input x using a baseline speech recognizer. In this work, we aim to generate a confusion set of sentences for an input sentence y as similar as possible to what we would get from a recognizer if we had the acoustic utterance for that sentence. Hence, learning is semi-supervised in the sense that training does not directly require transcribed speech examples but we use transcribed speech to build confusion models to generate the training examples. The representation $\mathbf{\Phi}$ maps each y to a feature vector $\mathbf{\Phi}(y) \in \mathbb{R}^d$. As in [8], this work uses morph unigram counts as features. The function $\Delta(y_i, z_i)$ is defined as the edit distance between z_i and y_i.

The learned averaged parameter vector $\bar{\alpha}_{avg}$ can be used for mapping an unseen acoustic input x to an output y by searching for the best scoring hypothesis:

$$y = \underset{\tilde{y} \in \mathbf{GEN}(x)}{\text{argmax}} \{\lambda \log P(\tilde{y} \mid x) + \bar{\alpha}_{avg} \cdot \mathbf{\Phi}(\tilde{y})\}$$

where $\mathbf{GEN}(x)$ is a set of hypotheses output from the baseline recognizer with the recognition score $\log P(\tilde{y} \mid x)$, and λ is a scaling factor optimized on a held-out set.

The WER-sensitive perceptron gives significantly better results (0.4%) than the standard perceptron when trained on the real ASR hypotheses while omitting the baseline recognition score. We also observe that the WER-sensitive perceptron is oblivious to the omission of the baseline recognition score in training. This is important since the scores output from the confusion models do not match the ASR baseline scores, and this would result in a mismatch between training and testing.

3. SYSTEM DESCRIPTION

The flow of our system starts with the generation of a confusion graph for a given word sequence. We then extract *n-best* hypotheses ($n = 1000$) out of that graph and apply different sampling schemes in order to select instances with different properties. In the final step, we use sampled hypotheses ($n = 50$) to train the language model discriminatively.

3.1. Confusion Graph Generation

$$\mathcal{C}(\mathcal{W}) = (prune(\mathcal{W} \circ \mathcal{L}_{\mathcal{W}} \circ \mathcal{CM}) \circ \mathcal{L}_{\mathcal{M}}^{-1} \circ \mathcal{G}_{\mathcal{M}})$$

Confusion graph $\big(\mathcal{C}(\mathcal{W})\big)$ generation happens in two stages. In the first stage, the given word sequence $\mathcal{W}$ is first composed with the lexicon $\mathcal{L}_{\mathcal{W}}$ and then with the confusion model $\mathcal{CM}$. Based on the granularity of the $\mathcal{CM}$, composition with the $\mathcal{L}_{\mathcal{W}}$ transforms $\mathcal{W}$ into a phone, syllable, morph or word lattice. Later composition with the $\mathcal{CM}$ simulates the noisy channel property of ASR as each confusion adds noise over the reference. Based on the length of the word sequence and the number of possible confusions, the resulting lattice might be very large. Hence, we prune that lattice by using only those arcs that form its 1000-best paths. Further investigation showed that this filtering has no negative effect on the results.

In the second stage, if the resulting lattice from the first stage is not morph-based, it is first composed with the inverse morph lexicon $\mathcal{L}_{\mathcal{M}}^{-1}$ since the language model transducer $\mathcal{G}_{\mathcal{M}}$ is morph-based. $\mathcal{G}_{\mathcal{M}}$ scores the sequences in the lattice based

on their likelihood seen in a typical sentence in the language being modeled.

3.1.1. Confusion Models

We use five different $\mathcal{CM}$s. While two of them are phone-based, the other three are syllable, morph, and word-based models. One of the phone-based $\mathcal{CM}$s is estimated using the acoustic similarities of phones by calculating the Bhattacharrya distance (bh) between the representative Gaussians of each phone in the acoustic model of the ASR, as proposed in [3].

The other four $\mathcal{CM}$s are estimated with the edit distance (ed) method, where we align the reference sentence with its recognized transcription and extract cases of substitutions, insertions, and deletions by using one of the optimal alignments. Unlike what was done in [6], we didn't use any special cost function to get the best alignment.

3.1.2. Language Models

We use three different LM approaches for $\mathcal{G}_{\mathcal{M}}$: GEN-LM, ASR-LM and NO-LM. While GEN-LM is estimated from Turkish newswire data set collected from the Internet, ASR-LM is derived from ASR n-bests. Since our ultimate goal is to simulate the ASR output, using the ASR-based LM is more intuitive as it is supposed to give higher scores to those alternatives that resemble the ASR output most. As a third approach, called NO-LM, we choose not to apply any language model, which means that only the scores of the $\mathcal{CM}$ are used to pick the n-best out of the lattice at the end.

3.2. Sampling Schemes

The top 50 simulated hypotheses from the confusion graph have very different word error (WE) distribution than the ASR WE distribution, containing less word errors. Hence we sample 50 instances from the top 1000 in order to match the WE distribution of the ASR. We use two specific sampling schemes.

The first sampling scheme, called the Uniform Sampling (US), follows the method applied in [9]. We select instances in uniform intervals from the WER-ordered list, hoping that they contain more word errors. In this scheme, denoted US-k, the best and the worst hypotheses are always selected.

The second approach is where we specifically sample instances according to the actual WE distribution obtained from the ASR output. We learn how frequently each unique word error occurs in the ASR 50-best, and try to simulate this distribution by picking samples from the artificially generated 1000-best in proportional numbers. We call this method ASRdist-50.

Note that, both sampling approaches sort the n-best list in ascending number of word errors before doing sampling.

4. EXPERIMENTS

4.1. Experimental Setup

In this study, DLM is applied on a Turkish broadcast news transcription task [10]. The data set we use in our experiments consists of approximately 194 hours of speech recorded from news channels. We divide the data into disjoint training (188 hours), held-out (3.1 hours) and test (3.3 hours) subsets, containing about 1.3 million, 23199 and 23410 words, respectively. The training subset is further divided into 12 equi-size bins. The bins 1-6 are used to obtain the n-best lists from which the confusion models are learned. As most of the confusions happen with very low probability, to reduce the computational cost, we filtered out all confusions with probability less than 0.01 in all models. After pruning, the phone-based models contain over a hundred parameters, whereas the syllable, morph, and word-based models contain 28K, 137K, and 362K parameters, respectively. These models are then used to generate the artificial n-bests from the reference transcriptions of bins 7-12.

We use the SRILM toolkit for building language models and the OpenFST library for finite-state operations. The speech recognition system is based on the statistical morph-based ASR system of [10]. Using this setup, the generative baseline and oracle WER on the held-out set are 22.9% and 14.2% and on the test set are 22.4% and 13.9%, respectively. When we use ASR 50-best from bins 7-12 for discriminative training, WERs drop to 22.1% and 21.8% on the held-out and the test sets, respectively.

4.2. Results

In our experiments, we use five different $\mathcal{CM}$s with three different LM approaches. Table 1 shows the WER improvement on the held-out set for these 15 different configurations. The results are obtained with the ASRdist-50 sampling scheme.

Table 1. WER (%) on held-out set for different $\mathcal{CM}$s and LMs w/ ASRdist-50 (held-out baseline: 22.9%; real (ASR) 50-best: 22.1%)

CMs	GEN-LM	ASR-LM	NO-LM
phone-based (bh)	22.8	22.7	N/A[1]
phone-based (ed)	22.6	22.7	N/A[1]
syllable-based	**22.5**	**22.4**	22.6
morph-based	22.6	**22.4**	**22.5**
word-based	22.6	**22.5**	22.7

Based on the NIST MAPSSWE test, phone-based models yield significantly smaller WER improvements compared to syllable-, morph- and word-based models, whereas the difference between these three models is not significant. When compared with the baseline on the held-out set, configurations

[1]These configurations were omitted because generating confusions with phone-based $\mathcal{CM}$s without an LM is computationally expensive.

Table 2. Sampling from the 1000-best list w/ morph-based $\mathcal{CM}$ and ASR-LM (test set baseline: 22.4%; real (ASR) 50-best: 21.8%)

Sampling Strategy	WER on test set	KL-distance
NoSampling	22.1	0.38
Top-50	22.1	0.43
US-50	22.0	0.27
ASRdist-50	21.8	0.23

given in bold in Table 1 are significantly better at $p < 0.005$. For the comparison of LMs, even though there is no significant difference between them, configurations with the ASR-LM result in significantly better WER improvement over the baseline. The best configuration uses the morph-based $\mathcal{CM}$ and the ASR-LM and is significantly better than the baseline at $p < 0.001$ on the held-out set.

With this best configuration, a comparison of the different sampling schemes over the test set is shown in Table 2. While using Top-50 and US-50 sampling strategies (as presented in section 3.2) is not significantly different than using no sampling strategy, improvement by the ASRdist-50 is significant at $p = 0.006$. This supports our assumption, that is, the more simulated n-bests resemble the ASR output in terms of WE distribution, the better WER improvement we get.

Figure 2 shows how the WE distribution of the ASRdist-50 is more similar to ASR's compared to Top-50's. We also measure this similarity in terms of KL-distance given in Table 2 for each sampling strategy. To test our assumption even further, we calculate the correlation between the KL-distance and WER improvement over all experiments to see whether the drop in KL-distance correlates with the drop in WER and find out that there is a correlation of 0.84, which further supports our assumption.

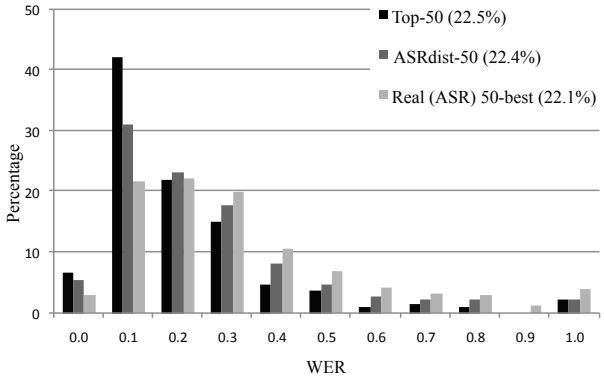

Fig. 2. Word error distribution on the held-out set

We also test if our artificially generated n-bests give similar results with the real ASR n-bests. Table 3 shows that when we combine the real n-bests from the bins 1-6 with the simulated ones from the bins 7-12, we get similar WER improvement with respect to the real n-bests from the bins 1-12.

Table 3. Test set results for combining real and simulated n-best lists (test set baseline: 22.4%)

bins 1-6	bins 7-12	WER (%)
Real (ASR)	Real (ASR)	21.5
Real (ASR)	Simulated (ASRdist-50)	21.6

5. CONCLUSION

In this study, we examined the semi-supervised learning of discriminative language models for a Turkish ASR system and experimented with different confusion and language models. We observe that phone-based confusion models are outperformed by syllable-, morph- and word-based models whereas there is no significant difference among these three. Among the different LMs, ASR-LM gives the best WER improvement over the baseline. Moreover, we get the best WER improvement by trying to match the ASR WE distribution. Finally, when we substitute half of the real n-best lists in our data set with the simulated ones, we achieve almost the same result as the whole set of real n-bests would.

6. REFERENCES

[1] B. Roark, M. Saraçlar, and M. Collins, "Discriminative n-gram language modeling," *Computer Speech and Language*, vol. 21, no. 2, pp. 373–392, April 2007.

[2] P. Xu, D. Karakos, and S. Khudanpur, "Self-supervised discriminative training of statistical language models," in *ASRU*, 2009, pp. 317–322.

[3] G. Kurata, N. Itoh, and M. Nishimura, "Acoustically discriminative training for language models," in *ICASSP*, 2009, pp. 4717–4720.

[4] G. Kurata, N. Itoh, and M. Nishimura, "Training of error-corrective model for ASR without using audio data," in *ICASSP*, 2011, pp. 5576–5579.

[5] Q.F. Tan, K. Audhkhasi, P.G. Georgiou, E. Ettelaie, and S. Narayanan, "Automatic speech recognition system channel modeling," in *Interspeech*, 2010, pp. 2442–2445.

[6] P. Jyothi and E. Fosler-Lussier, "Discriminative language modeling using simulated ASR errors," in *Interspeech*, 2010, pp. 1049–1052.

[7] H. Sak, M. Saraçlar, and T. Güngör, "Discriminative reranking of ASR hypotheses with morpholexical and n-best-list features," in *ASRU*, 2011, pp. 202–207.

[8] E. Arısoy, M. Saraçlar, B. Roark, and I. Shafran, "Discriminative language modeling with linguistic and statistically derived features," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 20, no. 2, pp. 540–550, 2012.

[9] E. Dikici, M. Semerci, M. Saraçlar, and E. Alpaydın, "Data sampling and dimensionality reduction approaches for reranking ASR outputs using discriminative language models," in *Interspeech*, 2011, pp. 1461–1464.

[10] E. Arısoy, D. Can, S. Parlak, H. Sak, and M. Saraçlar, "Turkish broadcast news transcription and retrieval," *IEEE Transactions on Audio, Speech & Language Processing*, vol. 17, no. 5, pp. 874–883, 2009.