Abstract

Discriminative language modeling (DLM) aims to choose the most accurate word sequence by reranking the alternatives output by the automatic speech recognizer (ASR). The conventional (supervised) way of training a DLM requires a large amount of acoustic recordings together with their manual reference transcriptions. These transcriptions are used to determine the target ranks of the ASR outputs, but may be hard to obtain. Previous studies make use of the existing transcribed data to build a confusion model which boosts the training set by generating artificial data: a process known as semi-supervised training. In this study we concentrate on the unsupervised setting where no manual transcriptions are available at all. We propose three ways to determine a sequence that could serve as the missing reference text and two approaches which use this information to (i) determine the ranks of the ASR outputs in order to train the discriminative model directly, and (ii) build a confusion model in order to generate artificial training examples. We compare our techniques with the supervised and the semi-supervised setups. Using the reranking variant of the WER-sensitive perceptron algorithm, we obtain word error rate improvements up to half of those of the supervised case.

Index Terms: discriminative language modeling, unsupervised training, ranking perceptron

1. Introduction

Discriminative language modeling aims to choose the most accurate word sequence by reranking the alternatives output by the automatic speech recognizer (ASR). The performance of modeling depends on the amount of training data. To train a discriminative language model (DLM) the standard (supervised) way, one needs the acoustic inputs (utterances) together with their manual transcriptions [1]. The acoustic input is passed through the ASR to obtain the N-best list of alternative transcriptions (called the hypotheses), and the manual transcriptions act as the ground truth to determine the accuracy of these hypotheses. However, listening to the recording and transcribing it is a costly process in terms of time and labor, and in some applications where the speaker’s identity must be kept confidential, it is not possible to obtain the ground truth.

A method used in this case is to build the DLM on artificial hypotheses instead of real (ASR) hypotheses. This is named as semi-supervised training. In this approach, one first learns a confusion model (CM) which represents the possible confusions (errors) made by the ASR system. The CM can be trained using phoneme similarities from an acoustic model [2, 3], translations alternatives [4] or competing words [5], and can be represented as a weighted-finite state transducer [6] or as a machine translation system [7]. The CM is then used to generate a number of mistranscriptions of sentences in a separate text corpus, as if those sentences were uttered and passed through the recognizer. Although this avoids the need for manual transcriptions to train the DLM, one still needs manually transcribed acoustic data to train the CM, even if this set might be much less in quantity, or out-of-domain.

By contrast, in the unsupervised case there is absolutely no transcribed text accompanying the acoustic input. Without transcribed text the accuracy of the hypotheses cannot be determined. Ways to build a DLM under such setting have been investigated in the literature for the last couple of years. In Xu et al. [5], a separate text corpus is utilized to disambiguate between the confused words in ASR outputs (so called the cohort sets) and training is done by maximum likelihood optimization using an exponential model. A more recent study learns phrasal cohorts instead of word cohorts by pairwise alignments of the hypotheses, uses them to generate artificial N-best lists, and reports a decrease in word error rate (WER) of up to 40% of that of the supervised case with the perceptron algorithm [8]. In Jyothi et al. [9], a large amount of unlabeled data is reprocessed using a weak acoustic model, and small but statistically significant improvements in WER are reported. Finally in Kuo et al. [10], the authors employ the Minimum Bayes Risk (MBR) criterion to choose the reference hypothesis for training the DLM via the perceptron.

The perceptron is a popular algorithm which learns the model parameters discriminatively. The variant of the algorithm mentioned in studies cited above is the structured perceptron, whose aim is to choose the most accurate hypothesis by minimizing an objective function related to WER [1]. In this study we propose using the ranking perceptron algorithm for unsupervised training of models. The ranking perceptron is another variant of the perceptron which aims to reorder the N-best list with respect to decreasing accuracy of the hypotheses [11]. It has been shown for supervised and semi-supervised training that the ranking perceptron outperforms the structured perceptron although training takes longer [12].

We also present our approaches to choose the word sequence (target output) for unsupervised training, which will serve as the missing reference text. We explore three methods based on the recognition score and the Minimum Bayes Risk (MBR) criterion. We use this target output to achieve DLM training conditions similar to the supervised and semi-supervised setups, by either training the DLM directly using the real hypotheses or building a CM first to generate artificial data for training. We show that we can achieve WER improvements over the baseline about 50% of the ones obtained using the supervised case. Experiments also show that unsupervised DLM training and unsupervised CM training can be combined to decrease WER further.

This paper is organized as follows: In Section 2 we give the mathematical details of DLM and CM training by presenting the linear model, the training algorithms and artificial hypothesis generation pipeline. Section 3 is devoted to the novel target
output selection scheme. We explain the setup and results of our experiments in Section 4, and conclude the paper with a summary and discussion in Section 5.

2. Methods

2.1. Linear model

The linear model sets the mathematical basis for discriminative language modeling computations [13]. The elements of the linear model are:

- **x**: The acoustic input (i.e., the utterance itself)
- **y**: The target output (ground truth). For supervised training, this is the reference transcription of x. For semi-supervised training, this is the sentence in the separate text corpus.
- **GEN()**: The function which generates a list of hypotheses for training the DLM. For supervised training, GEN represents the ASR system and takes x as its input parameter. For semi-supervised training, GEN stands for the CM and has the input parameter y.
- **Ŷ**: The set of hypotheses output by GEN
- **Φ(·)**: The representation which maps the hypotheses to a d-dimensional space (can be \(\Phi(x, \hat{y})\) or \(\Phi(y, \hat{y})\), depending on the setup)

Each element of the hypothesis vector \(\Phi\) is associated with a weight, and the objective of DLM training is to estimate the model vector \(w\) which contains these weights. In the testing phase, the hypothesis whose vector yields the greatest inner product with the optimal model is selected to be the final output: \(y^* = \arg\max_y \langle w, \Phi(x, \hat{y}) \rangle\).

2.2. Training the DLM

As denoted in the previous section, training a DLM means estimating the model (weight) vector, \(w\). In our study we use two different algorithms for this purpose, both of which utilize an iterative optimization scheme.

2.2.1. Structured WER-sensitive perceptron (WPer)

The perceptron is a popular method used in solving structured prediction problems in which the aim is to pick among \(\hat{y}\) the hypothesis with the least number of word errors (WE) with respect to \(y\). The variant of this algorithm which we use in this study, the WER-sensitive perceptron [14], tries to minimize a loss function in terms of the edit distances between \(y\) and \(\hat{y}\), and denoted by \(\Delta(y, \hat{y})\).

The pseudocode of WPer is shown in Figure 1. Here, \(y_i\) denotes the oracle which is the hypothesis with the least WE, and \(z_i\) is the current-best hypothesis which yields the highest inner product score under the current model. Passing over the data a number of times, the model updates itself by favoring the features in \(y_i\) and penalizing the ones in \(z_i\), with a sensitivity multiplier \(\Delta(y_i, z_i)\). The resulting weights are averaged for robustness.

2.2.2. Ranking WER-sensitive perceptron (WPerRank)

The WPer algorithm uses only two hypotheses of the N-best list for training. Nevertheless, the remaining hypotheses can also play a role in determining a new ranking of the list. The ranking WER-sensitive perceptron algorithm is used to make use of this potential.

The ranking approach considers the WE of the hypotheses as an indicator of their desired (target) ranking: For two hypotheses \(a\) and \(b\), if \(a\) has fewer word errors, it must have an inner product score significantly higher than that of \(b\) (The significance gap is defined by \(\tau\Delta(a, b)\), where \(\tau\) is a positive margin multiplier). The algorithm with the pseudocode given in Figure 2 updates the model vector just like the WPer, this time for each \((a, b)\) pair which violates the above criterion. Another improvement in this algorithm is the use of learning (\(\eta\)) and decay (\(\gamma\)) rates for iterative optimization.

2.3. Training the CM

In cases where there is no sufficient acoustic data to train a DLM the supervised way, or an ASR system that is in-domain (i.e., suited to the context upon which the DLM will be trained), it is possible to do the DLM training by incorporating other available text data (which is not accompanied by an acoustic recording), via confusion modeling.

A confusion model (CM) is a model which contains the acoustic confusions that could be made by the ASR system, and in some sense represents the inherent variability in the real ASR output hypotheses. Once the confusion model is learned, one can apply it on any text to generate artificial hypotheses, as if the uttering of that text were passed through an ASR system. The text and the artificial hypotheses can then be fed into discriminative modeling as training examples.

In this study, we adopt a weighted finite-state transducer (WFST) based confusion modeling technique as in [15]. The language unit representation we use to build the CM is morphs, which are statistically derived subword units [16]. The procedure starts by analyzing the real ASR hypotheses. Each hypothesis, represented as a morph sequence, is aligned to the corresponding reference transcription using the Levenshtein (edit) distance. This alignment gives a list of morph pairs subject to confusion by the ASR. The probability of confusion for any morph pair is computed using the frequency of their match-ups.
The CM is represented as a WFST, whose input-output pairs are these morph pairs, and weights are the computed confusion probabilities.

The generation of artificial hypotheses after learning the CM follows the following composition sequence: N-best($W \circ \text{LM} \circ \text{CM}$). Here $W$ represents the input text upon which artificial hypotheses will be generated. The text is represented in morphs by composing it with an appropriate lexicon, $\text{LM}$. Composing this result with $\text{CM}$ yields the alternative hypotheses in the form of a graph, along with their probabilities of occurrence. Finally, the most probable $N$ paths are selected and returned.

3. Choosing the target output

Whether it is used to train a DLM or to build a CM, the reference transcription of the utterance plays an essential role. In supervised training it is needed to determine the level of accuracy of the hypotheses, a measure of their target ranks in the N-best list. In semi-supervised learning, it acts as the ground truth from which the probabilities of confusions are derived. When the reference transcription is not available, we need to determine another target output which will take over its place. In this section we explore the ways to generate or choose this word sequence by observing the hypotheses in the N-best list.

A first choice would be to select the 1-best (i.e., the hypothesis which has the highest recognition score) as the target output. In this case, the 1-best will have no word errors, and the WE of other hypotheses will be assigned by aligning each to the 1-best. However in this setup, the WPer algorithm would not make any updates since the oracle and current best hypotheses will always be the same. A crucial observation is that with the WPerRank algorithm it is still possible train the model by using the other hypotheses of the N-best list.

A second approach is to determine the target output by minimizing an error function, or maximizing an objective function for this purpose is the Minimum Bayes Risk (MBR) \cite{17, 18, 10}. The MBR value for a target output candidate hypothesis $y$ is defined as:

$$l(y|x) = E_{\hat{y}|x}[L(\hat{y}, y)]$$

$$= \sum_{\hat{y} \in \text{GEN}(x)} L(\hat{y}, y)p(\hat{y}|x)$$

(1)

Here, $L(\hat{y}, y)$ denotes the Levenshtein distance of the other hypotheses aligned to that candidate, and $p(\hat{y}, x)$ denotes the posterior probability (recognition score) assigned to the hypotheses by the ASR system. The MBR reference is the hypothesis which yields the lowest MBR score:

$$\hat{y} = \arg\min_{y \in \text{GEN}(x)} l(y|x)$$

(2)

As a third approach, the MBR technique could be utilized on the token (word, morph, etc.) level rather than the sentence level. In this method, the hypotheses are aligned token-wise, and each chunk is processed separately. This requires creating a token confusion network (also known as the sausage), and finding the best path along this graph from the first node to the last. This method is called Segmental MBR (SegMBR, in short) \cite{19}.

4. Experiments

4.1. Dataset and experimental setup

In this study we apply discriminative language modeling for Turkish ASR. Our dataset consists of manually transcribed acoustic recordings of broadcast news collected from TV/radio channels and divided into three subsets as shown in Table 1.

<table>
<thead>
<tr>
<th>Subset</th>
<th>Duration (h)</th>
<th>Number of utterances</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>188</td>
<td>105355</td>
</tr>
<tr>
<td>Held-out</td>
<td>3.1</td>
<td>1947</td>
</tr>
<tr>
<td>Test</td>
<td>3.3</td>
<td>1784</td>
</tr>
</tbody>
</table>

The hypotheses are organized in 50-best lists and are represented as morph sequences. There are a total of 45889 unique morphs in the dataset. The feature vector $\Phi$ consists of morph unigram counts. The first element of the feature vector, $\Phi_0$, contains the recognition score given by the ASR system to the hypothesis, where applicable.

The baseline ASR system is prepared using the AT&T \cite{20} and the SRILM \cite{21} toolkits. Morfessor algorithm \cite{22} is employed to determine Turkish morphs and MBR computations are done using the SRILM toolkit. The WFST-based confusion modeling system is implemented by the OpenFST library \cite{23}. WPer is iterated at most 50 times whereas WPerRank is iterated at most 40 times, and algorithmic parameters are optimized on the held-out set. The generative baseline and oracle rates are 22.9% and 14.2% for the held-out set, and 22.4% and 13.9% for the test set, respectively.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Explanation</th>
<th>CM</th>
<th>DLM</th>
</tr>
</thead>
<tbody>
<tr>
<td>i</td>
<td>Supervised</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>ii</td>
<td>Semi-Supervised</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>iii</td>
<td>Unsupervised DLM</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>iv</td>
<td>Unsupervised CM</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

Table 2 shows the availability of data in terms of acoustic input (A) and manual reference transcriptions (T) for different training methods presented so far. For instance for Scenario (iv), we have some acoustic recordings with which we build the CM, and some text data (actually, the manual reference transcriptions of a different portion of the dataset) on which the DLM will be trained. As denoted in Section 1, the main focus will be on scenarios (iii) and (iv) which can be viewed as a substitute for scenarios (i) and (ii), respectively, for the case where manual transcriptions of the acoustic data are not available.

4.2. Experimental results

In our experiments we divide our 188 hours of training data into two equal subsets, one used to train the DLM and the other reserved for building the CM where applicable, as shown in Table 2. We apply two-fold cross validation by using these sets interchangeably and average the results when reporting.

4.2.1. Unsupervised training of the DLM

In the first experimental set we create a case similar to supervised training where the DLM is trained using real ASR hypotheses. This time, however, we do not have the manual reference transcriptions. We apply the three approaches presented
in Section 3 to choose the target output to replace the missing manual transcription and use this as a reference to determine the target ranks of the hypotheses.

The first three rows of Table 3 show unsupervised DLM training (Scenario iii) performance in terms of WER with respect to the three approaches and two algorithms on the held-out (hld) and test (tst) subsets. The last row contains the supervised case (Scenario i) where the target output is the manual reference itself, and is included for comparison.

Table 3: Unsupervised DLM training WER (%) (Baseline: hld 22.9%, tst 22.4%).

<table>
<thead>
<tr>
<th>Target output</th>
<th>WPer</th>
<th>WPerRank</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>hld</td>
<td>tst</td>
</tr>
<tr>
<td>1-best</td>
<td>22.9</td>
<td>22.4</td>
</tr>
<tr>
<td>MBR</td>
<td>22.7</td>
<td>22.3</td>
</tr>
<tr>
<td>SegMBR</td>
<td>22.7</td>
<td>22.3</td>
</tr>
<tr>
<td>Reference</td>
<td>22.2</td>
<td>22.0</td>
</tr>
</tbody>
</table>

We see from Table 3 that WPerRank outperforms WPer regardless of what word sequence is used as the target output. The choice of target output has no significant effect on DLM performance. Note that WPer with the 1-best setup yields the same WER as the baseline, as expected. This observation is also consistent with [10]. The WER improvement with WPerRank over WPer on the held-out set is an absolute 0.4% with a significance level of $p < 0.001$, which corresponds to 50% of the gain that could be obtained under the supervised setup. This improvement is also reflected on the test subset with a WER of 22.1%.

### 4.2.2. Unsupervised training of the CM

In the second set of experiments we investigate the case where the in-domain data (the data we would like the DLM to be based on) consists only of text, but we also have some other acoustic data available. We first build a CM using the acoustic data. This time, unlike semi-supervised training, the hypotheses in the N-best list are aligned to the chosen target output instead of the manual reference to determine the confusion probabilities. The learned CM can then be applied on the text data to generate artificial hypotheses for training the DLM. We use the manual transcriptions of the DLM subset as the text data, to be able to obtain comparable results to those of the previous experiments. Table 4 shows the WER of two training algorithms with this setup.

Table 4: Unsupervised CM training WER (%) (Baseline: hld 22.9%, tst 22.4%).

<table>
<thead>
<tr>
<th>Target output</th>
<th>WPer</th>
<th>WPerRank</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>hld</td>
<td>tst</td>
</tr>
<tr>
<td>1-best</td>
<td>22.7</td>
<td>22.3</td>
</tr>
<tr>
<td>MBR</td>
<td>22.9</td>
<td>22.4</td>
</tr>
<tr>
<td>SegMBR</td>
<td>22.8</td>
<td>22.4</td>
</tr>
<tr>
<td>Reference</td>
<td>22.6</td>
<td>22.2</td>
</tr>
</tbody>
</table>

The superiority of WPerRank over WPer is once again visible in Table 4, although the WER has slightly increased with respect to the previous results. However, we now see that the gap between the semi-supervised case (where manual reference is the target output, shown in the last row) and the unsupervised case has decreased, yielding only an absolute 0.1% change (which is not statistically significant) on both subsets.

### 4.2.3. Combination of methods

In the experiments which included confusion modeling so far, the (artificial) hypotheses used to train the DLM were generated on a different dataset than the (real) hypotheses used to train the CM. In this section we investigate whether there is any room for further improvement on WER by using a combination of these two hypothesis sets for DLM training. Table 5 shows the WER obtained with WPerRank for two experiments using this idea.

Table 5: Combination of methods WPerRank WER (%) (Baseline: hld 22.9%, tst 22.4%).

<table>
<thead>
<tr>
<th>Method</th>
<th>hld</th>
<th>tst</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sup. + Semi-Sup.</td>
<td>21.9</td>
<td>21.6</td>
</tr>
<tr>
<td>Unsup. DLM + Unsup. CM</td>
<td>22.2</td>
<td>22.0</td>
</tr>
</tbody>
</table>

The first row of Table 5 represents the case where the artificial hypotheses generated by the semi-supervised setup are combined with the real ASR hypotheses used for confusion modeling. Compared to the last row of Table 3 (the supervised case), the held-out WER is slightly increased but the test WER is unchanged. This means that adding artificial hypotheses to the training set brings no additional gain over supervised training. A similar comparison for the unsupervised setting is given in the second row of Table 5. With 1-best chosen as the target output, the same operation offers a slight but not significant improvement of 0.1% over the first row of Table 3.

### 5. Conclusions

In this paper we focus on the unsupervised discriminative language modeling problem where the manual transcriptions of acoustic inputs are not available for training the DLM. We used three different methods to choose the target output to replace the manual reference. We trained the discriminative models by (i) using the target output to determine the ranks of the real ASR hypotheses and (ii) building a confusion model to generate artificial examples on a text corpus. We showed that the ranking perceptron algorithm is more suited to unsupervised DLM training problem, offering better system accuracies than the structured perceptron. We also showed that by combining the two hypothesis sets for CM and DLM training, we can obtain a slight decrease in WER.

The superiority of WPerRank is because the algorithm considers each hypothesis in the N-best list instead of only two as in WPer. However training takes longer with this setup, so there is a tradeoff between algorithmic complexity and accuracy.

The main advantage of unsupervised DLM training is that it shows improvements in ASR accuracy even when matched acoustic and text data are not present. We believe that this technique is important especially for processing under-resourced languages. In the future we would like to experiment more using an enriched dataset to understand how discriminative modeling performance responds to changing data types and sizes.

### 6. Acknowledgements

The authors would like to thank Arda Çelebi for the artificial hypothesis generation framework, and the Center for Spoken Language Understanding for computational resources. This research is supported in part by TUBITAK project 109E142, the Turkish State Planning Organization under the TAM project 2007K120610, and the Bogazici University Research Fund (BU-BAP) project D-7948.
7. References


