Discriminative Training on language model

Zheng Chen, Kai-Fu Lee, Ming-jing Li

Microsoft Research

Abstract

Statistical language models have been successfully applied to a lot of problems, including speech recognition, handwriting, Chinese pinyin-input etc. In recognition, statistical language model, such as trigram, is used to provide adequate information to predict the probabilities of hypothesized word sequences. The traditional method relying on distribution estimation are sub-optimal when the assumed distribution form is not the true one, and that "optimality" in distribution estimation does not automatically translate into "optimality" in classifier design. This paper proposed a discriminative training method to minimize the error rate of recognizer rather than estimate the distribution of training data. Furthermore, lexicon is also optimized to minimize the error rate of the decoder through discriminative training. Compared to the traditional LM building method, our systems gets approximately 5%-25% recognition error reduction with discriminative training on language model building.

1. Introduction

Statistical language models have been successfully applied to a lot of problems, including speech recognition, handwriting, Chinese pinyin-input etc. In recognition, statistical language model, such as trigram, is used to provide adequate information to predict the probabilities of hypothesized word sequences.

Usually, Maximum likelihood estimation (MLE) is used in language model training. MLE, relying on distribution estimation, is probably optimal if the underlying models are correct. But the trigram model is not the "true model" of the language. Furthermore, trigram models try to separate the likely from the unlikely, without considering their actual confusability. However, for recognition, the relative scores of candidates are more important than the absolute scores.

Moreover, during training, only a small fraction of the data all over the world is selected as the training data. The inadequate training data make it difficult to obtain the complete knowledge of the form of the data distribution. So the traditional method relying on distribution estimation are sub-optimal when the assumed distribution form is not the true one, and that "optimality" in distribution estimation does not automatically translate into "optimality" in classifier design. [1]

In this paper, we apply "discriminative training"[1,2,3,4] into language model building. Different from the traditional method, "discriminative training" aims to minimize the error rate of recognizer, while "traditional statistical training" aims to optimize the estimation of the distribution. A key to the development of the discriminative method is to build an error function which can be evaluated and minimized by the system. Our approach is to first train an MLE model, and then iteratively improve it using discriminative training. There are some similar approaches, such as "Corrective Training in Computer Speech & Language" from IBM. But to our knowledge, this is the first application of discriminative training to language modeling.

In the next section, we will brief introduce the basic theory about discriminative training. Section 3 details the implementation of language model optimization through discriminative training. In the section 4, we evaluate the proposed approaches by some experiments. Finally, we give some conclusions.

2. Theory

Statistical language model plays an important role in speech recognition. Combined with acoustic model, it can help system to find the most possible word strings in speech recognition. Let X be the observation of the input; in the speech recognition, X is the acoustic data, and in the pinyin input method, X is the stream of the Roma letter, etc. What we want to solve is to find the correct Chinese characters (H), which can maximize the $\Pr(H \mid X)$, we can use the Bayes rule to decompose it into two problems as equation 2.1.

$$\Pr(H \mid X) = \frac{\Pr(X \mid H)\Pr(H)}{\Pr(X)} \qquad (2.1)$$

For given X, $\Pr(X)$ is constant. So we only need to consider $\Pr(X \mid H)$ and $\Pr(H)$. $\Pr(X \mid H)$ is the acoustic model or typing model, and $\Pr(H)$ is called language model [5]. In the traditional method, we will choose $H = H_i$ if $\Pr(H_i \mid X)$ is maximum. We call it "maximum a posteriori"(MAP) decision [1,6]. Unfortunately, lack of training data will cause it to be sub-optimal, and it cannot minimize the error rate of recognizer. Because it only aims to maximize the probability of the correct model instead of to minimize the probability of other incorrect competed models. Consider the equation 2.1, traditional maximal likelihood method does not consider the influence of $\Pr(X)$, in real, $\Pr(X)$ can be rewritten as equation 2.2.

$$\Pr(X) = \sum_{i}^{M} \Pr(X \mid H_i) \Pr(H_i) \qquad (2.2)$$

While in the recognition process, the relative score is important than the absolute score. If we cannot discriminate the different between correct and incorrect answer, the performance of system will not good. So we import the discriminative training to minimize sum of the probability of incorrect models. We define a misrecognition measure [1] as equation 2.3.

$$d_i(X) = -\Pr(X_i, H) + \left[\frac{1}{M-1} \sum_{j, j \neq i} \Pr(X_j, H)^{\eta} \right]^{\frac{1}{\eta}} \qquad (2.3)$$

Where η is a positive number. $d_i(X) > 0$ implies misrecognition and $d_i(X) \leq 0$ means correct decision. When η approaches ∞, the bracket becomes $\max_{j, j \neq i} \Pr(X_j, H)$. By varying the value of η, we can take all the competing models into consideration.

We can use sigmoid function to define the loss function [1] as equation 2.4.

$$l_i(X) = l(d_i(X)) = \frac{1}{1 + \exp(-\gamma d_i(X) + \theta)} \qquad (2.4)$$

So we try to find suitable parameters to minimize the loss $l(X)$ [1] as defined in equation 2.5.

$$l(X) = \sum_{i=1}^{M} l_i(X) = \sum_{i=1}^{M} l(d_i(X)) \qquad (2.5)$$

3. Optimization by Discriminative Training

The training process is based on the recognition results on original language model. Traditional maximum likelihood estimation can be used to build the original language model, such as Trigram. For each sentence in training corpus, the corresponding recognition results would be obtained with the statistical language model. Correct answer and hypothesis can be aligned through dynamic programming. For each different word pair, we try to enhance the correct word pair and weaken the error word pair at the same time. All these modifications can be done on count file of word pair directly. After discriminative training, we can train another language model from the updated count file. After several iterations, we could reduce the error rate of recognizer.

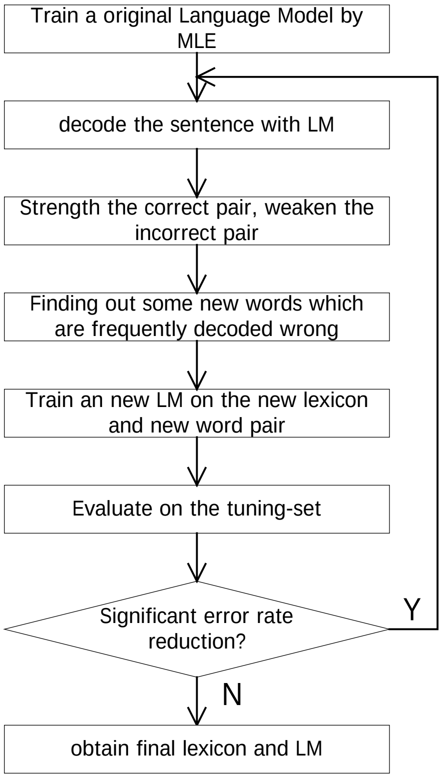
For example, supposing we are training Trigram language model and S_i is one sentence in training corpus. Based on original language model, S_i can be segment into $(w_1, w_2, \cdots, w_n)$. After recognition, new segmentation results $(w'_1, w'_2, \cdots, w'_m)$ will be obtained. We can align these two results, then we can tag the error words on word sequence. Let's suppose w_i is aligned with w'_j. And both of these two words contain error character. Then we can modify the count file as equation 3.1.

$$\begin{aligned}
&C(w_{i-2}, w_{i-1}, w_i) = C(w_{i-2}, w_{i-1}, w_i) + \alpha \\
&C(w'_{j-2}, w'_{j-1}, w'_j) = C(w'_{j-2}, w'_{j-1}, w'_j) - \beta \\
&C(w_{i-1}, w_i, w_{i+1}) = C(w_{i-1}, w_i, w_{i+1}) + \alpha \\
&C(w'_{j-1}, w'_j, w'_{j+1}) = C(w'_{j-1}, w'_j, w'_{j+1}) - \beta \\
&C(w_i, w_{i+1}, w_{i+2}) = C(w_i, w_{i+1}, w_{i+2}) + \alpha \\
&C(w'_j, w'_{j+1}, w'_{j+2}) = C(w'_j, w'_{j+1}, w'_{j+2}) - \beta \qquad (3.1)
\end{aligned}$$

Furthermore, discriminative training also optimized the lexicon. In the discriminative training, some new words, which are frequently decoded wrong, are selected as word candidates. Through training, some significant important new words are added into lexicon so that recognitions errors due to these words can be eliminated. There are three kinds of new words which are frequently decoded wrong.

- Some words which are not consider by the linguists
- Domain specific words
- Proper noun, such as personal name, place name, date, number, etc.

These new words are not included in the traditional dictionary. The probabilities of these new words are estimated by the trigram of single characters. In the process of recognition, the discrimination between these new words and other similar characters is little. So they are frequently decoded wrong. Through counting the error number of decoded strings, some new words are chosen and added into dictionary to increase the discrimination.



Train a original Language Model by MLE
decode the sentence with LM
Strength the correct pair, weaken the incorrect pair
Finding out some new words which are frequently decoded wrong
Train an new LM on the new lexicon and new word pair
Evaluate on the tuning-set
Significant error rate reduction?
Y
N
obtain final lexicon and LM

Figure 3.1 Unified approach on discriminative training

These two optimization processes can be combined together to form the unified approach to minimize the recognition error rate of decoder. Figure 3.1 shows the dataflow of this unified approach. Language model optimization and lexicon optimization can be processed in one approach. Furthermore, LM optimization can help system to obtain more powerful lexicon, and new lexicon will improve the quality of LM. The optimization process is stop until no significant improvement is achieved on the tuning set.

4. Experiments

In our experiments, we applied discriminative training into speech recognition. In order to simplify the decoder, we only consider the effect of language model instead of acoustic model. More than 600 mega bytes newspapers are collected as training data, and 2 mega bytes balanced corpus are collected as testing data. A baseline LM is built with CMU LM Toolkit. Then we prune the language model to the proper size with entropy-based cutoff method [7].

Next step is to optimize the LM with discriminative training. We use the held-out data [5] selected from training data as the tuning data. The training corpus is divided into n parts. Each time, n-1 parts are selected as the training data and the other one as tuning data. There will be n parallel sub processes to do the discriminative training. The iteration continues until there is no significant error rate reduction on the tuning data. Then we evaluate the language model on the test-set.

From the experiments, we show that the discriminative training can obtain 5%-25% error reduction on different sizes of language model. The results are shown as table 4.1. For 10 MB LM, the error rate reduced from 6.56% to 6.25%, the error reduction is 4.7%. While for the 100 MB LM, the error rate reduced from 4.01% to 3.0%, the error reduction is 25.2%. For large size, more discriminative pairs can be added into LM to discriminate the confusion between the words. On the contrary, rare space is left for LM to store the confusion pairs under the small language model size. Although large model size can get better performance, a proper size will be considered according to the detail applications.

LM Size (M)	Error Rate (MLE)	Error Rate (Discriminative Training)	Error Reduction
10	6.56%	6.25%	4.7%
100	4.01%	3.0%	25.2%

Table 4.1 Comparison between MLE and Discriminative Training

We also evaluate the language model on the different test-set. The results are shown as table 4.2. We found that the effect of discriminative training is varied on the test-sets with different styles. If the training data is similar with testing data, then the effect will significant. However, if the training data is different from testing data, the effect will becomes lower. The theme of many_news test-set is same as training data, so the effect is greater than other test-set. Opentest and People's daily are also similar to the training data, so their effect is fairish. While the webdata is unlike the training data, so the effect is little. From the experiment, we can infer that if some information about real application can be gathered, then we can optimize the language model to fit the need of real environments.

Test-Set	Error rate (MLE)	Error Rate (Discriminative Training)	Error Reduction
Many_news	3.50%	3.29%	6%
People's Daily	4.61%	4.37%	5.2%
Opentest	6.56%	6.25%	4.7%
IME	7.91%	7.60%	3.9%
Webdata	9.24%	9.02%	2.4%

Table 4.2 Comparison between different test-set (Language Model size is pruned to 10M)

Another experiment is done to optimize the lexicon. Some word pairs which are frequently decoded wrong are chosen as the candidates of new words. We classified these new words into three categories:

- Word consists with high frequency co-occurrences characters, for example, 我的(mine), 你的(your), 他的(his), 她的(her), 一个(one), etc. From the point of the linguists, these words cannot be included in the lexicon. But adding these words will get approximately 1%-2% error reduction on different test-set.
- Domain specific words. Lacking the information of one specific domain, the performance of system is dramatically dropped compared to the general domain. Lacking sufficient training data on these specific words, the probabilities of these new words cannot be estimated correctly. In our experiment, some new words about Internet are frequently decode wrong, e.g. 域名(domain name), 网页(Web page), 网址(net address), 超链接(hyperlink), etc.
- Proper noun. There are many kinds of proper noun in the corpus. Unfortunately, only a small fraction of proper noun is included in the lexicon. Furthermore, the distribution of proper noun is scatter. So it is impossible to gather all of the proper noun into the lexicon. Through discriminative training, some proper noun are detected and added into the dynamic lexicon [8] to improve the performance of the recognizer. In our experiment, some proper noun are detected, e.g. 唐学逵(Chinese personal name), 葛罗夫(Foreign personal name), 安钢(Chinese organization), 二十几岁(twenties), etc.

All these candidates are sorted by their frequency and added into the lexicon to optimize the new language model.

5. Conclusion

In this paper, we proposed a new approach in training LM to improve the performance of recognition. But the models are not generally usable any more, e.g., LM for speech recognition may not be good for handwriting or spelling correction. Compared to the traditional LM building method, our systems gets approximately 5%-25% recognition error reduction with discriminative training on language model building.

6. Acknowledgements

Our thanks to DongFeng Cai, Hai-Feng Wang, JianFeng Gao for their helps on the data collection and model training.

7. Reference

[1] B.-H. Juang, W. Chou and C.-H. Lee, statistical and discriminative methods for speech recognition,

[2] W. Chou, C.-H. Lee, B.-H. Juang, Minimum Error Rate Training based on N-best String Models, Proc. 1993 Int. Conf. On Acoustics, Speech and Signal Processing, Minneapoils, MN, Vol.2, pp. 662-655, April 1993.

[3] W. Reichl, G. Ruske, Discriminative Training for Continuous Speech Recognition, Proc. 1995 Europ. Conf. On Speech Communication and Technology, Madrid, Vol. 1, pp. 537-540, September 1995.

[4] V. Valtechev, J. J. Odell, P.C. Woodland, S. J. Young, Lattice-Based Discriminative Training For Large Vocabulary Speech Recognition, In Proc. Int. Conf. Acoustics, Speech and Signal Process. 1996, Atlanta, GA, Vol. 2, pp. 605-608, May 1996.

[5] Frederick Jelinek, Statistical Methods for Speech Recognition, The MIT Press, Cambridge, Massachusetts, 1997.

[6] Kai-Fu Lee, Automatic Speech Recognition, Kluwer Academic Publishers, 1989.

[7] A. Stolcke, “Entropy-based Pruning of Backoff Language Models” in Proc. DRAPA News Transcription and Understanding Workshop, Lansdowne, VA. 1998. pp.270-274.

[8] Jun Zhao, Zheng Chen, Refining Language model via Lexicon Optimization, submitted to ACL2000 workshop, hongkong, 2000.