論文 / 著書情報
Article / Book Information

Title	Discriminative Lexicon Adaptation for Improved Character Accuracy - A New Direction in Chinese Language Modeling
Author	Yi-Cheng Pan, Lin-chan Lee, Sadaoki Furui
Journal/Book name	47th Annual Meeting of the ACL and 4th IJCNLP of the AFNLP, , , page 755-763
発行日 / Issue date	2009, 8

Discriminative Lexicon Adaptation for Improved Character Accuracy – A New Direction in Chinese Language Modeling

Yi-cheng Pan
Speech Processing Labratory
National Taiwan University
Taipei, Taiwan 10617
thomashughPan@gmail.com

Lin-shan Lee
Speech Processing Labratory
National Taiwan University
Taipei, Taiwan 10617
lsl@speech.ee.ntu.edu.tw

Sadaoki Furui
Furui Labratory
Tokyo Institute of Technology
Tokyo 152-8552 Japan
furui@furui.cs.titech.ac.jp

Abstract

While OOV is always a problem for most languages in ASR, in the Chinese case the problem can be avoided by utilizing character n-grams and moderate performances can be obtained. However, character n-gram has its own limitation and proper addition of new words can increase the ASR performance. Here we propose a discriminative lexicon adaptation approach for improved character accuracy, which not only adds new words but also deletes some words from the current lexicon. Different from other lexicon adaptation approaches, we consider the acoustic features and make our lexicon adaptation criterion consistent with that in the decoding process. The proposed approach not only improves the ASR character accuracy but also significantly enhances the performance of a character-based spoken document retrieval system.

1 Introduction

Generally, an automatic speech recognition (ASR) system requires a lexicon. The lexicon defines the possible set of output words and also the building units in the language model (LM). Lexical words offer local constraints to combine phonemes into short chunks while the language model combines phonemes into longer chunks by more global constraints. However, it's almost impossible to include all words into a lexicon both due to the technical difficulty and also the fact that new words are created continuously. The missed out words will never be recognized, which is the well-known OOV problem. Using graphemes for OOV handling is proposed in English (Bisani and Ney, 2005). Although this sacrifices some of the lexical constraints and introduces a further difficulty to combine graphemes back into words, it is compensated by its ability for open vocabulary ASR. Morphs are another possibility, which are longer than graphemes but shorter than words, in other western languages (Hirsimäki et al., 2005).

	5.8K characters	61.5K full lexicon
bigram	63.55%	73.8%
trigram	74.27%	79.28%

Table 1: Character recognition accuracy under different lexicons and the order of language model.

Chinese language, on the other hand, is quite different from western languages. There are no blanks between words and the definition for words is vague. Since almost all characters in Chinese have their own meanings and words are composed of the characters, there is an obvious solution for the OOV problem: simply using all characters as the lexicon. In Table 1 we see the differences in character recognition accuracy by using only 5.8K characters and a full set of 61.5K lexicon. The training set and testing set are the same as those that will be introduced in Section 4.1. It is clear that characters alone can provide moderate recognition accuracies while augmenting new words significantly improves the performance. If the words' semantic functionality can be abandoned, which definitely can not be replaced by characters, we can treat words as a means to enhance character recognition accuracy. Such arguments stand at least for Chinese ASR since they evaluate on character error rate and do not add explicit blanks between words. Here we formulate a lexicon adaptation problem and try to discriminatively find out not only OOV words beneficial for ASR but also those existing words that can be deleted.

Unlike previous lexicon adaptation or construction approaches (Chien, 1997; Fung, 1998; Deligne and Sagisaka, 2000; Saon and Padmanabhan, 2001; Gao et al., 2002; Federico and Bertoldi, 2004), we

Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP, pages 755–763,
Suntec, Singapore, 2-7 August 2009.

consider the acoustic signals and also the whole speech decoding structure. We propose to use a simple approximation for the character posterior probabilities (PPs), which combines acoustic model and language model scores after decoding. Based on the character PPs, we adapt the current lexicon. The language model is then re-trained according the new lexicon. Such procedure can be iterated until convergence.

Characters, are not only the output units in Chinese ASR but also have their roles in spoken document retrieval (SDR). It has been shown that characters are good indexing units. Generally, characters can at least help OOV query handling; in the subword-based confusion network (S-CN) proposed by Pan *et al.* (2007), characters are even better than words for in-vocabulary (IV) queries. In addition to evaluating the proposed approach on ASR performance, we investigate its helpfulness when integrated with an S-CN framework.

2 Related Work

Previous works for lexicon adaptation were focused on OOV rate reduction. Given an adaptation corpus, the standard way is to first identify OOV words. These OOV words are selected into the current lexicon based on the criterion of frequency or recency (Federico and Bertoldi, 2004). The language model is also re-estimated according to the new corpus and new derived words.

For Chinese, it is more difficult to follow the same approach since OOV words are not readily identifiable. Several methods have been proposed to extract OOV words from the new corpus based on different statistics, which include associate norm and context dependency (Chien, 1997), mutual information (Gao et al., 2002), morphological and statistical rules (Chen and Ma, 2002), and strength and spread measure (Fung, 1998). The used statistics generally help find sequences of characters that are consistent to the general concept of words. However, if we focus on ASR performance, the constraint of the extracted character strings to be word-like is unnecessary.

Yang *et al.* (1998) proposed a way to select new character strings based on average character perplexity reduction. The word-like constraint is not required and they show a significant improvement on character-based perplexity. Similar ideas were found to use mutual probability as an effective measure to combine two existing lexicon words into a new word (Saon and Padmanabhan, 2001). Though proposed for English, this method is effective for Chinese ASR (Chen et al., 2004). Gao *et al.* (2002) combined an information gain-like metric and the perplexity reduction criterion for lexicon word selection. The application is on Chinese pinyin-to-character conversion, which has very good correlation with the underlying language model perplexity.

The above works actually are all focused on the text level and only consider perplexity effect. However, as pointed by Rosenfeld (2000), lower perplexity does not always imply lower ASR error rate. Here we try to face the lexicon adaptation problem from another aspect and take the acoustic signals involved in the decoding procedure into account.

3 Proposed Approach

3.1 Overall Picture

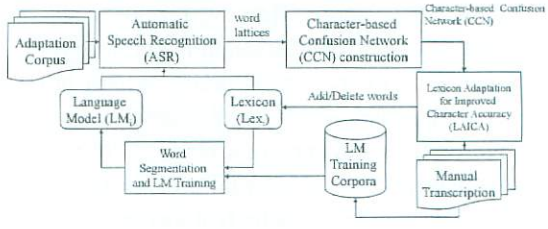

Figure 1: The flow chart of the proposed approach.

We show the complete flow chart in Figure 1. At the beginning we are given an adaptation spoken corpus and manual transcriptions. Based on a baseline lexicon (Lex_0) and a language model (LM_0) we perform ASR on the adaptation corpus and construct corresponding word lattices. We then build character-based confusion networks (CCNs) (Fu et al., 2006; Qian et al., 2008). On the CCNs we perform the proposed algorithm to add and delete words into/from the current lexicon. The LM training corpora joined with the adaptation corpus is then segmented using Lex_1 and the language model is in turn re-trained, which gives LM_1. This procedure can be iterated to give Lex_i and LM_i until convergence.

3.2 Character Posterior Probability and Character-based Confusion Network (CCN)

Consider a word W as shown in Figure 2 with characters $\{c_1 c_2 c_3\}$ corresponding to the edge e starting at time τ and ending at time t in a word lattice. During decoding the boundaries between c_1

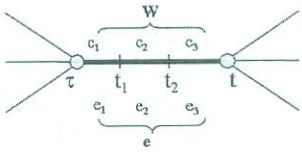

Figure 2: An edge e of word W composed of characters $c_1c_2c_3$ starting at time τ and ending at time t.

and c_2, and c_2 and c_3 are recorded respectively as t_1 and t_2. The posterior probability (PP) of the edge e given the acoustic features A, $P(e|A)$, is (Wessel et al., 2001):

$$P(e|A) = \frac{\alpha(\tau) \cdot P(x_\tau^t|W) \cdot P_{LM}(W) \cdot \beta(t)}{\beta_{start}}, \quad (1)$$

where $\alpha(\tau)$ and $\beta(t)$ denote the forward and backward probability masses accumulated up to time τ and t obtained by the standard forward-backward algorithm, $P(x_\tau^t|W)$ is the acoustic likelihood function, $P_{LM}(W)$ the language model score, and β_{start} the sum of all path scores in the lattice. Equation (1) can be extended to the PP of a character of W, say c_1 with edge e_1:

$$P(e_1|A) = \frac{\alpha(\tau) \cdot P(x_\tau^{t_1}|c_1) \cdot P_{LM}(c_1) \cdot \beta(t_1)}{\beta_{start}}. \quad (2)$$

Here we need two new probabilities, $P_{LM}(c_1)$ and $\beta(t_1)$. Since neither is easy to estimate, we make some approximations. First, we assume $P_{LM}(c_1) \approx P_{LM}(W)$. Of course this is not true, the actual relation being $P_{LM}(c_1) \geq P_{LM}(W)$, since the set of events having c_1 given its history includes a set of events having W given the same history. We used the above approximation for easier implementation. Second, we assume that after c_1 there is only one path from t_1 to t: through c_2 and c_3. This is more reasonable since we restrain the hypotheses space to be inside the word lattice, and pruned paths are simply neglected. With this approximation we have $\beta(t_1) = P(x_{t_1}^t|c_2c_3) \cdot \beta(t)$. Substituting these two approximate values for $P_{LM}(c_1)$ and $\beta(t_1)$ in Equation (2), the result turns out to be very simple: $P(e_1|A) \approx P(e|A)$. With similar assumptions for the character edges e_2 and e_3, we have $P(e_2|A) \approx P(e_3|A) \approx P(e|A)$. Similar results were obtained by Yao *et al.* (2008) from a different point of view.

The result that $P(e_i|A) \approx P(e|A)$ seems to diverge from the intuition: approximating an n-segment word by splitting the probability of the entire edge over the segments – $P(e_i|A) \approx \sqrt[n]{P(e|A)}$. The basic meaning of Equation (1) is to calculate the ratio of the paths going through a specific edge divided by the total paths while each path is weighted properly. Of course the paths going through a sub-edge e_i should be definitely more than the paths through the corresponding full-edge e. As a result, $P(e_i|A)$ should usually be greater than $P(e|A)$, as implied by the intuition. However, the inter-connectivity between all sub-edges and the proper weights of them are not easy to be handled well. Here we constrain the inter-connectivity of sub-edges to be only inside its own word edge and also simplify the calculation of the weights of paths. This offers a tractable solution and the performance is quite acceptable.

After we obtain the PPs for each character arc in the lattice, such as $P(e_i|A)$ as mentioned above, we can perform the same clustering method proposed by Mangu *et al.* (2000) to convert the word lattice to a strict linear sequence of clusters, each consisting of a set of alternatives of character hypotheses, or a character-based confusion network (CCN) (Fu et al., 2006; Qian et al., 2008). In CCN we collect the PPs for all character arc c with beginning time τ and end time t as $P([c;\tau,t]|A)$ (based on the above mentioned approximation):

$$P([c;\tau,t]|A) = \frac{\sum_{\substack{H = w_1 \ldots w_N \in \text{lattice}: \\ \exists i \in \{1 \ldots N\}: \\ w_i \text{ contains } [c;\tau,t]}} P(H)P(A|H)}{\sum_{\text{path } H' \in \text{lattice}} P(H')P(A|H')}, \quad (3)$$

where H stands for a path in the word lattice. $P(H)$ is the language model score of H (after proper scaling) and $P(A|H)$ is the acoustic model score. CCN was known to be very helpful in reducing character error rate (CER) since it minimizes the expected CER (Fu et al., 2006; Qian et al., 2008). Given a CCN, we simply choose the characters with the highest PP from each cluster as the recognition results.

3.3 Lexicon Adaptation with Improved Character Accuracy (LAICA)

In Figure 3 we show a piece of a character-based confusion network (CCN) aligned with the corresponding manual transcription characters. Such alignment can be implemented by an efficient dynamic programming method. The CCN is composed of several strict linear ordering clusters of

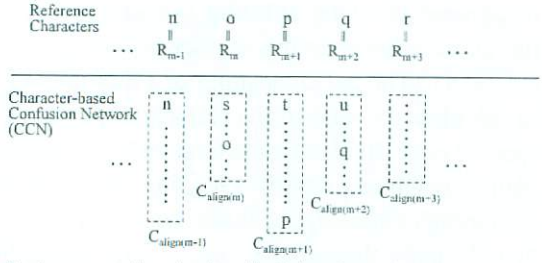

Figure 3: A character-based confusion network (CCN) and corresponding reference manual transcription characters.

character alternatives. In the figure, $C_{align(m)}$ is a specific cluster aligned with the m^{th} character in the reference, which contains characters $\{s \dots o \dots\}$ (The alphabets $n, o \dots u$ are symbols for specific Chinese characters) . The characters in each cluster of CCN are well sorted according to the PP, and in each cluster a special null character ϵ with its PP being equal to 1 minus the summation of PPs for all character hypotheses in that cluster. The clusters with ϵ ranked first are neglected in the alignment.

After the alignment, there are only three possibilities corresponding to each reference character. (1) The reference character is ranked first in the corresponding cluster (R_{m-1} and the cluster $C_{align(m-1)}$). In this case the reference character can be correctly recognized. (2) The reference character is included in the corresponding cluster but not ranked first. ($[R_m \dots R_{m+2}]$ and $\{C_{align(m)}, \dots, C_{align(m+2)}\}$) (3) The reference character is not included in the corresponding cluster (R_{m+3} and $C_{align(m+3)}$). For cases (2) and (3), the reference character will be incorrectly recognized.

The basic idea of the proposed lexicon adaptation with an improved character accuracy (LAICA) approach is to enhance the PPs of those incorrectly recognized characters by adding new words and deleting existing words in the lexicon. Here we only focus on those characters of case (2) mentioned above. This is primarily motivated by the minimum classification error (MCE) discriminative training approach proposed by Juang *et al.* (1997), where a sigmoid function was used to suppress the impacts of those perfectly and very poorly recognized training samples. In our approach, the case (1) is the perfect case and case (3) is the very poor one. Another motivation is that for characters in case (1), since they are already correctly recognized we do not try to enhance their PPs.

The procedure of LAICA then becomes simple. Among the aligned reference characters and clusters of CCN, case (1) and (3) are anchors. The reference characters between two anchors then become our focus segment and their PPs should be enhanced. By investigating Equation (3), to enhance the PP of a specific character we can adjust the language model ($P(H)$), and the acoustic model ($P(A|H)$), or we can simply modify the lexicon (the constraint under summation). We should add new words to cover the characters of case (2) to enlarge the numerator of Equation (3) and at the same time delete some existing words to suppress the denominator.

In Figure 3, reference characters $[R_m R_{m+1} R_{m+2}$=$opq]$ and the clusters $\{C_{align(m)}, \dots, C_{align(m+2)}\}$ show an example of our focus segment. For each such segment, we at most add one new word and delete an existing word. From the string $[opq]$ we choose the longest OOV part from it as a new word. To select a word to be deleted, we choose the longest in-vocabulary (IV) part from the top ranked competitors of $[opq]$, which are then $[stu]$ in clusters $\{C_{align(m)}, \dots, C_{align(m+2)}\}$. This is also motivated by MCE that we only suppress the strongest competitors' probabilities. Note that we do not delete single-characters in the procedure.

The "*at most one*" constraint here is motivated by previous language model adaptation works (Federico, 1999) which usually try to introduce new evidences in the adaptation corpus but with the least modification of the original model. Of course the modification of language models led by the addition and deletion of words is hard to quantify and we choose to add and delete as fewer words as possible, which is just a simple heuristic. On the other hand, adding fewer words means that longer words are added. It has been shown that longer words are more helpful for ASR (Gao et al., 2004; Saon and Padmanabhan, 2001).

The proposed LAICA approach can be regarded as a discriminative one since it not only considers the reference characters but also those wrongly recognized characters. This can be beneficial since it reduces potential ambiguities existing in the lexicon.

The Expectation-Maximization algorithm
1. Bootstrap initial word segmentation by maximum-matching algorithm (Wong and Chan, 1996)
2. Estimate unigram LM
3. Expectation: Re-segment according to the unigram LM
4. Maximization: Estimate the n-gram LM
5. Expectation: Re-segment according to the n-gram LM
6. Go to step 4 until convergence

Table 2: EM algorithm for word segmentation and LM estimation

3.4 Word Segmentation and Language Model Training

If we regard the word segmentation process as a hidden variable, then we can apply EM algorithm (Dempster et al., 1977) to train the underlying n-gram language model. The procedure is described in Table 2. In the algorithm we can see two expectation phases. This is natural since at the beginning the bootstrap segmentation can not give reliable statistics for higher order n-gram and we choose to only use the unigram marginal probabilities. The procedure was well established by Hwang *et al.* (2006).

Actually, the EM algorithm proposed here is similar to the n-multigram model training procedure proposed by Deligne and Sagisaka (2000). The role of multigrams can be regarded as the words here, except that multigrams begin from scratch while here we have an initial lexicon and use maximum-matching algorithm to offer an acceptable initial unigram probability distributions. If the initial lexicon is not available, the procedure proposed by Deligne and Sagisaka (2000) is preferred.

4 Experimental Results

4.1 Baseline Lexicon, Corpora and Language Models

The baseline lexicon was automatically constructed from a 300 MB Chinese news text corpus ranging from 1997 to 1999 using the widely applied PAT-tree-based word extraction method (Chien, 1997). It includes 61521 words in total, of which 5856 are single-characters. The key principles of the PAT-tree-based approach to extract a sequence of characters as a word are: (1) high enough frequency count; (2) high enough mutual information between component characters; (3) large enough number of context variations on both sides; (4) not dominated by the most frequent context among all context variations. In general the words extracted have high frequencies and clear boundaries, thus very often they have good semantic meanings. Since all the above statistics of all possible character sequences in a raw corpus are combinatorially too many, we need an efficient data structure such as the PAT-tree to record and access all such information.

With the baseline lexicon, we performed the EM algorithm as in Table 2 to train the trigram LM. Here we used a 313 MB LM training corpus, which contains text news articles in 2000 and 2001. Note that in the following Sections, the pronunciations of the added words were automatically labeled by exhaustively generating all possible pronunciations from all component characters' canonical pronunciations.

4.2 ASR Character Accuracy Results

A set of broadcast news corpus collected from a Chinese radio station from January to September, 2001 was used as the speech corpus. It contained 10K utterances. We separated these utterances into two parts randomly: 5K as the adaptation corpus and 5K as the testing set. We show the ASR character accuracy results after lexicon adaptation by the proposed approach in Table 3.

	LAICA-1			LAICA-2		
	A	D	A+D	A	D	A+D
Baseline	+1743	-1679	+1743 -1679	+409	-112	+314 -88
79.28	80.48	79.31	80.98	80.58	79.33	81.21

Table 3: ASR character accuracies for the baseline and the proposed LAICA approach. Two iterations are performed, each with three versions. A: only add new words, D: only delete words and A+D: simultaneously add and delete words. + and - means the number of words added and deleted, respectively.

For the proposed LAICA approach, we show the results for one (LAICA-1) and two (LAICA-2) iterations respectively, each of which has three different versions: (A) only add new words into the current lexicon, (D) only delete words, (A+D) simultaneously add and delete words. The number of added or deleted words are also included in Table 3.

There are some interesting observations. First, we see that deletion of current words brought much

less benefits than adding new words. We try to give some explanations. Deleting existing words in the lexicon actually is a passive assistance for recognizing reference characters correctly. Of course we eliminate some strong competitive characters in this way but we can not guarantee that reference characters will then have high enough PP to be ranked first in its own cluster. Adding new words into the lexicon, on the other hand, offers explicit reinforcement in PP of the reference characters. Such reinforcement offers the main positive boosting for the PP of reference characters. These boosted characters are under some specific contexts which normally correspond to OOV words and sometimes in-vocabulary (IV) words that are hard to be recognized.

From the model training aspect, adding new words gives the maximum-likelihood flavor while deleting existing words provides discriminant ability. It has been shown that discriminative training does not necessarily outperform maximum-likelihood training until we have enough training data (Ng and Jordan, 2001). So it is possible that discriminatively trained model performs worse than that trained by maximum likelihood. In our case, adding and deleting words seem to compliment each other well. This is an encouraging result.

Another good property is that the proposed approach converged quickly. The number of words to be added or deleted dropped significantly in the second iteration, compared to the first one. Generally the fewer words to be changed the fewer recognition improvement can be expected. Actually we have tried the third iteration and simply obtained dozens of words to be added and no words to be deleted, which resulted in negligible changes in ASR recognition accuracy.

4.3 Comparison with other Lexicon Adaptation Methods

In this section we compare our method with two other traditionally used approaches: one is the PAT-tree-based as introduced in Section 4.1 and the other is based on mutual probability (Saon and Padmanabhan, 2001), which is the geometrical average of the direct and reverse bigram:

$$P_M(w_i, w_j) = \sqrt{P_f(w_j|w_i)P_r(w_i|w_j)},$$

where the direct ($P_f(\cdot)$ and reverse bigram ($P_r(\cdot)$) can be estimated as:

$$P_f(w_j|w_i) = \frac{P(W_{t+1} = w_j, W_t = w_i)}{P(W_t = w_i)},$$

$$P_r(w_j|w_i) = \frac{P(W_{t+1} = w_j, W_t = w_i)}{P(W_{t+1} = w_j)}.$$

$P_M(w_i, w_j)$ is used as a measure about whether to combine w_i and w_j as a new word. By properly setting a threshold, we may iteratively combine existing characters and/or words to produce the required number of new words. For both the PAT-tree- and mutual-information-based approaches, we use the manual transcriptions of the development 5K utterances to collect the required statistics and we extract 2159 and 2078 words respectively to match the number of added words by the proposed LAICA approach after 2 iterations (without word deletion). The language model is also re-trained as described in Section 3.4. The results are shown in Table 4, where we also include the results of our approach with 2 iterations and adding words only for reference.

	PAT-tree	Mutual Probability	LAICA-2(A)
Character Accuracy	79.33	80.11	80.58

Table 4: ASR character accuracies on the lexicon adapted by different approaches.

From the results we observe that the PAT-tree-based approach did not give satisfying improvements while the mutual probability-based one worked well. This may be due to the sparse adaptation data, which includes only 81K characters. PAT-tree-based approach relies on the frequency count, and some terms which occur only once in the adaptation data will not be extracted. Mutual probability-based approach, on the other hand, considers two simple criterion: the components of a new word occur often together and rarely in conjunction with other words (Saon and Padmanabhan, 2001). Compared with the proposed approach, both PAT-tree and mutual probability do not consider the decoding structure.

Some new words are clearly good for human sense and definitely convey novel semantic information, but they can be useless for speech recognition. That is, character n-gram may handle these words equally well due to the low ambiguities with other words. The proposed LAICA approach tries to focus on those new words which can not be handled well by simple character n-grams. Moreover, the two methods discussed here do not offer possible ways to delete current words, which can be considered as a further advantage of the proposed LAICA approach.

4.4 Application: Character-based Spoken Document Indexing and Retrieval

Pan *et al.* (2007) recently proposed a new Subword-based Confusion Network (S-CN) indexing structure for SDR, which significantly outperforms word-based methods for IV or OOV queries. Here we apply S-CN structure to investigate the effectiveness of improved character accuracy for SDR. Here we choose characters as the subword units, and then the S-CN structure is exactly the same as CCN, which was introduced in Section 3.2.

For the SDR back-end corpus, the same 5K test utterances as used for the ASR experiment in Section 4.2 were used. The previously mentioned lexicon adaptation approaches and corresponding language models were used in the same speech recognizer for the spoken document indexing. We automatically choose 139 words and terms as queries according to the frequency (at least six times in the 5K utterances). The SDR performance is evaluated by mean average precision (MAP) calculated by the `trec_eval`[1] package. The results are shown in Table 5.

	Character Accuracy	MAP
Baseline	79.28	0.8145
PAT-tree	79.33	0.8203
Mutual Probability	80.11	0.8378
LAICA-2(A+D)	81.21	0.8628

Table 5: ASR character accuracies and SDR MAP performances under S-CN structure.

From the results, we see that generally the increasing of character recognition accuracy improves the SDR MAP performance. This seems trivial but we have to note the relative improvements. Actually the transformation ratios from the relative increased character accuracy to the relative increased MAP for the three lexicon adaptation approaches are different. A key factor making the proposed LAICA approach advantageous is that we try to extensively raise the incorrectly recognized character posterior probabilities, by means of adding effective OOV words and deleting ambiguous words. Actually S-CN is relying on the character posterior probability for indexing, which is consistent with our criterion and makes our approach beneficial. The degree of the raise of character posterior probabilities can be visualized more clearly in the following experiment.

[1] http://trec.nist.gov/

4.5 Further Investigation: the Improved Rank in Character-based Confusion Networks

In this experiment, we have the same setup as in Section 4.2. After decoding, we have character-based confusion networks (CCNs) for each test utterance. Rather than taking the top ranked characters in each cluster as the recognition result, we investigate the ranks of the reference characters in these clusters. This can be achieved by the same alignment as we did in Section 3.3. The results are shown in Table 6.

	# of ranked reference characters	Average Rank
baseline	70993	1.92
PAT-tree	71038	1.89
Mutual Probability	71054	1.81
LAICA-2(A+D)	71083	1.67

Table 6: Average ranks of reference characters in the confusion networks constructed by different lexicons and corresponding language models

In Table 6 we only evaluate ranks on those reference characters that can be found in its corresponding confusion network cluster (case (1) and (2) as described in Section 3.3). The number of those evaluated reference characters depends on the actual CCN and is also included in the results. Generally, over 93% of reference characters are included (the total number is 75541). Such ranks are critical for lattice-based spoken document indexing approaches such as S-CN since they directly affect retrieval precision. The advantage of the proposed LAICA approach is clear. The results here provide a more objective point of view since SDR evaluation is inevitably effected by the selected queries.

5 Conclusion and Future Work

Characters together is an interesting and distinct language unit for Chinese. They can be simultaneously viewed as words and subwords, which offer a special means for OOV handling. While relying only on characters gives moderate performances in ASR, properly augmenting new words significantly increases the accuracy. An interesting question would then be how to choose words to augment. Here we formulate the problem as an adaptation one and try to find the best way to alter the current

lexicon for improved character accuracy.

This is a new perspective for lexicon adaptation. Instead of identifying OOV words from adaptation corpus to reduce OOV rate, we try to pick out word fragments hidden in the adaptation corpus that help ASR. Furthermore, we delete some existing words which may result in ambiguities. Since we directly match our criterion with that in decoding, the proposed approach is expected to have more consistent improvements than perplexity based criterions.

Characters also play an important role in spoken document retrieval. This extends the applicability of the proposed approach and we found that the S-CN structure proposed by Pan *et al.* for spoken document indexing fitted well with the proposed LAICA approach.

However, there still remain lots to be improved. For example, considering Equation 3, the language model score and the summation constraint are not independent. After we alter the lexicon, the LM is different accordingly and there is no guarantee that the obtained posterior probabilities for those incorrectly recognized characters would be increased. We increased the path alternatives for those reference characters but this can not guarantee to increase total path probability mass. This can be amended by involving the discriminative language model adaptation in the iteration, which results in a unified language model and lexicon adaptation framework. This can be our future work. Moreover, the same procedure can be used in the construction. That is, beginning with only characters in the lexicon and using the training data to alter the current lexicon in each iteration. This is also an interesting direction.

References

Maximilian Bisani and Hermann Ney. 2005. Open vocabulary speech recognition with flat hybrid models. In *Interspeech*, pages 725–728.

Keh-Jiann Chen and Wei-Yun Ma. 2002. Unknown word extraction for chinese documents. In *COLING*, pages 169–175.

Berlin Chen, Jen-Wei Kuo, and Wen-Hung Tsai. 2004. Lightly supervised and data-driven approaches to mandarin broadcast news transcription. In *ICASSP*, pages 777–780.

Lee-Feng Chien. 1997. Pat-tree-based keyword extraction for Chinese information retrieval. In *SIGIR*, pages 50–58.

Sabine Deligne and Yoshinori Sagisaka. 2000. Statistical language modeling with a class-based n-multigram model. *Comp. Speech and Lang.*, 14(3):261–279.

A. P. Dempster, N. M. Laird, and D. B. Rubin. 1977. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistics Society*, 39(1):1–38.

Marcello Federico and Nicola Bertoldi. 2004. Broadcast news LM adaptation over time. *Comp. Speech Lang.*, 18:417–435.

Marcello Federico. 1999. Efficient language model adaptation through MDI estimation. In *Intersspech*, pages 1583–1586.

Yi-Sheng Fu, Yi-Cheng Pan, and Lin-Shan Lee. 2006. Improved large vocabulary continuous Chinese speech recognition by character-based consensus networks. In *ISCSLP*, pages 422–434.

Pascale Fung. 1998. Extracting key terms from chinese and japanese texts. *Computer Processing of Oriental Languages*, 12(1):99–121.

Jianfeng Gao, Joshua Goodman, Mingjing Li, and Kai-Fu Lee. 2002. Toward a unified approach to statistical language modeling for Chinese. *ACM Transaction on Asian Language Information Processing*, 1(1):3–33.

Jianfeng Gao, Mu Li, Andi Wu, and Chang-Ning Huang. 2004. Chinese word segmentation: A pragmatic approach. In *MSR-TR-2004-123*.

Teemu Hirsimäki, Mathias Creutz, Vesa Siivola, Mikko Kurimo, Sami Virpioja, and Janne Pylkkönen. 2005. Unlimited vocabulary speech recognition with morph language models applied to Finnish. *Comp. Speech Lang.*

Mei-Yuh Hwang, Xin Lei, Wen Wang, and Takahiro Shinozaki. 2006. Investigation on mandarin broadcast news speech recognition. In *Interspeech-ICSLP*, pages 1233–1236.

Bing-Hwang Juang, Wu Chou, and Chin-Hui Lee. 1997. Minimum classification error rate methods for speech recognition. *IEEE Trans. Speech Audio Process.*, 5(3):257–265.

Lidia Mangu, Eric Brill, and Andreas Stolcke. 2000. Finding consensus in speech recognition: Word error minimization and other applications of confusion networks. *Comp. Speech Lang.*, 14(2):373–400.

Andrew Y. Ng and Michael I. Jordan. 2001. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In *Advances in Neural Information Processing Systems (14)*, pages 841–848.

Yi-Cheng Pan, Hung-Lin Chang, and Lin-Shan Lee. 2007. Analytical comparison between position specific posterior lattices and confusion networks based on words and subword units for spoken document indexing. In *ASRU*.

Yao Qian, Frank K. Soong, and Tan Lee. 2008. Tone-enhanced generalized character posterior probability (GCPP) for Cantonese LVCSR. *Comp. Speech Lang.*, 22(4):360–373.

Ronald Rosenfeld. 2000. Two decades of statistical language modeling: Where do we go from here? *Proceeding of IEEE*, 88(8):1270–1278.

George Saon and Mukund Padmanabhan. 2001. Data-driven approach to designing compound words for continuous speech recognition. *IEEE Trans. Speech and Audio Process.*, 9(4):327–332, May.

Frank Wessel, Ralf Schlüter, Klaus Macherey, and Hermann Ney. 2001. Confidence measures for large vocabulary continuous speech recognition. *IEEE Trans. Speech Audio Process.*, 9(3):288–298, Mar.

Pak-kwong Wong and Chorkin Chan. 1996. Chinese word segmentation based on maximum matching and word binding force. In *Proceedings of the 16th International Conference on Computational Linguistic*, pages 200–203.

Kae-Cherng Yang, Tai-Hsuan Ho, Lee-Feng Chien, and Lin-Shan Lee. 1998. Statistics-based segment pattern lexicon: A new direction for chinese language modeling. In *ICASSP*, pages 169–172.