CHARACTERISTICS OF WORD LANGUAGE MDOEL AND CHARACTER LANGUAGE MODEL FOR MANDARIN CHINESE

Roger H.Y. Leung, Hong C. Leung
Department of Electronic Engineering,
The Chinese University of Hong Kong
{hyleung, hcleung}@ee.cuhk.edu.hk.

ABSTRACT

In this paper, we analyze and compare the characteristics of Chinese Word Language Model (LM) and Character Language Model. While Word LM has been found to provide powerful constraints for speech recognition, it is also known that word LM suffers from out-of-vocabulary and sparse data problems. These problems are particularly severe in Chinese, as new Chinese words can be created with a large degree of freedom. Furthermore, as there are no clearly defined word boundaries in Chinese, some forms of word segmentation procedures must first be performed before word LM can be applied. In this paper, we explore the possibility of using character LM, which can potentially alleviate some of the known problems with word LM.

1. INTRODUCTION

The simplest lexical unit for Chinese is the character. There are more than 10,000 Chinese characters. Most Chinese word is composed of one to four characters. The combinations are usually based on the lexical meaning of the characters similar to prefix or suffix. However, as the rules of word formation are not well defined [1], the total number of Chinese words is not well defined as well. Hence, in this paper, we will investigate the use of word LM and character LM.

2. EXPERIMENTAL DATABASE

Our study is based on the second release (Apr95) of Mandarin CallHome corpus [2], distributed by the Linguistic Data Consortium (LDC). There are 80 conversations in the training set and 20 in the development test set. The training set contains 19,965 sentences. The character vocabulary size and word vocabulary size for the character and word LMs are 2098 and 5774, respectively. The average word length is about 1.39 character. The detailed statistics are shown in Table 1.

Transcription	**Training**	**Development Testing**
# of dialog	80	20
# of sentence	19,965	5,378
# of word	127,063	34,699
# of character	177,148	48,218
# of unique word	5,774	2,936
# of unique character	2,098	1,466
Ave. word length (character/word)	1.394	1.390

Table 1: Detailed statistics of Call Home Corpus

3. ACOUSTIC MODELING

Mandarin is a monosyllabic and tonal language. The total number of phonologically allowed syllables is about 1300. Each syllable is assigned a tone and there are a total of five lexicon tones. Tones can be separately recognized using the pitch contour information. If tone information is ignored, the 1300 tonal syllable can be reduced to only 408 base syllables. We used the 408 base syllables as speech units in all our experiments. We used the same architecture for all the recognizers in our experiments: hidden Markov Model (HMM) technique for acoustic modeling. Each of the 408 HMMs has 8-states and 8 mixtures. The HMM states are arranged in a left-to-right, no-state-skipping topology. The segmental k-means algorithm is used for training and the Viterbi algorithm is used for decoding. 13 MFCC, 13 ΔMFCC, Energy and an ΔEnergy are used as feature vectors for the recognizer. Since our main theme is to compare the LMs, no context-dependent models have been used to improve the recognition result.

4. LANGUAGE MODELING

4.1 Building Language Model

The bigram and trigram probability can be estimated by the simple relative frequency approach [3].

$$Bigram: P(w_n \mid w_{n-1}) = \frac{c(w_{n-1}, w_n)}{c(w_{n-1})}$$

$$Trigram: P(w_n \mid w_{n-2}, w_{n-1}) = \frac{c(w_{n-2}, w_{n-1}, w_n)}{c(w_{n-2}, w_{n-1})}$$

where the function $c(.)$ counts the number of string in the blanket

The use of relative frequencies as a way to estimate probabilities is known as Maximum Likelihood Estimation. Table 2 shows the bigram count of 7 words in Call Home corpus. The relative frequency is then calculated by normalizing the bigram with its unigram count. Table 3 shows the bigram probabilities after normalization.

W_{n-1} W_n							
	259	126	119	6	10	6	4
	21	2	1	7	6	5	1
	50	0	12	13	0	30	4
	11	6	5	73	10	17	7
	0	1	0	0	4	50	3
	100	0	2	44	6	1	7
	7	0	1	5	0	12	17

Table 2: Bigram count for 7 words (out of 5774) in Call Home Corpus

W_{n-1} W_n							
	0.0641	0.0312	0.0295	0.0015	0.0025	0.0015	0.0010
	0.0553	0.0053	0.0026	0.0184	0.0158	0.0132	0.0026
	0.1082	0.0000	0.0260	0.0281	0.0000	0.0649	0.0087
	0.0052	0.0028	0.0024	0.0345	0.0047	0.0080	0.0033
	0.0000	0.0045	0.0000	0.0000	0.0179	0.2242	0.0135
	0.0358	0.0000	0.0007	0.0158	0.0021	0.0004	0.0025
	0.0083	0.0000	0.0012	0.0060	0.0000	0.0143	0.0203

Table 3: Bigram probabilities for 7 word in Call Home Corpus

From Table 2, we also notice that the disfluency problem is very severe in Call Home corpus, for example, there are 259 number of bigram - , 73 number of bigram - and 17 number of bigram - . Generally, these bigrams would not appear in written texts. However, people tend to repeat their words in telephone conversation. This problem adds an extra difficulty for speech recognition [4].

In our simple backoff LM, the probability backoffs from a trigram to a bigram, and then to a unigram estimation. The idea is incorporated in the approximated formula:

$$P(w_i \mid w_{i-2}, w_{i-1}) = \begin{cases} \tilde{P}(w_i \mid w_{i-2}, w_{i-1}) & if \;\; c(w_{i-2}, w_{i-1}, w_i) > 0 \\ a(w_{n-2}^{n-1})\tilde{P}(w_i \mid w_{i-1}) & if \;\; c(w_{i-2}, w_{i-1}, w_i) = 0 \;\; and \; c(w_{i-1}, w_i) > 0 \\ a(w_{n-1})\tilde{P}(w_i) & otherwise \end{cases}$$

where $a(w_{n-2}^{n-1}), a(w_{n-1})$ are backoff rate that depend on the counts c and assure that the probability P when summed over all words w_i adds up to 1. $\tilde{P}$ is the discounted probability.

4.2 Perplexity and OOV of Language Models

In the development test set, the OOV rate of word LM has been found to be at 4.15%. As Table 4 shows, the simple backoff word trigram perplexity is 313, which is significantly higher than the simple backoff word bigram perplexity of 175. This suggests that the training data may be insufficient for training the trigram word LM, since well-trained trigrams are supposed to have a lower perplexity than bigrams.

Language Model	**Word Perplexity**	**OOV (%)**	**Vocab. Size**
Word Bigram	175.13	4.15%	5,774
Word Trigram	313.23	4.15%	5,774

Table 4: Perplexity of fair word bigram and fair word trigram

A similar analysis has been conducted on the character LM. Table 5 shows our results. The OOV rate of character LM has been found to be at 0.62%, which is significantly lower than the OOV of word LM. The simple backoff character bigram perplexity is 63.13, while the simple backoff character trigram perplexity is 96.03.

Language Model	**Character Perplexity**	**OOV (%)**	**Vocab. Size**
Character Bigram	63.13	0.62%	2,098
Character Trigram	96.03	0.62%	2,098

Table 5: Perplexity of fair character bigram and fair character trigram

4.3 Comparison of Word LM and Character LM

The LM information as shown in Table 4 and Table 5 are on two different bases. Table 4 is based on a word-level analysis, whereas Table 5 is based on a character-level analysis. In order to enable us to compare the two levels of information, we define [5]

$$PP_C = \sqrt[L]{PP_W}$$

where PP_W is the average word perplexity, PP_C is the average character perplexity, and L is the average length of a word.

Furthermore, we adjust the vocabulary size of the language models to achieve the same OOV rate of both character and word language model, at which the two language models are comparable. Figure 1 and Figure 2 illustrate the effect of the vocabulary size on the two language models.

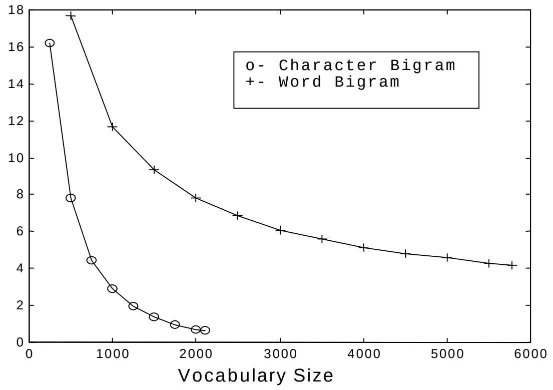

Figure 1: Effect of vocabulary size on OOV %

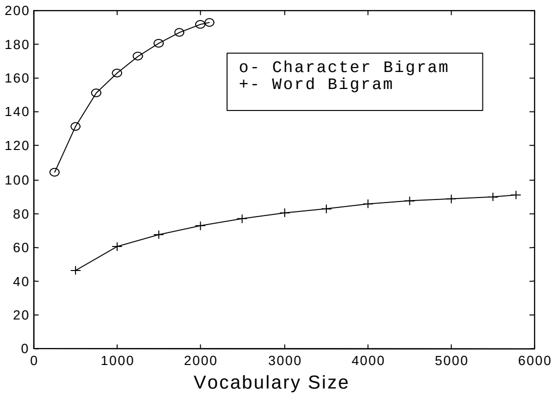

Figure 2: Effect of vocabulary size on word perplexity

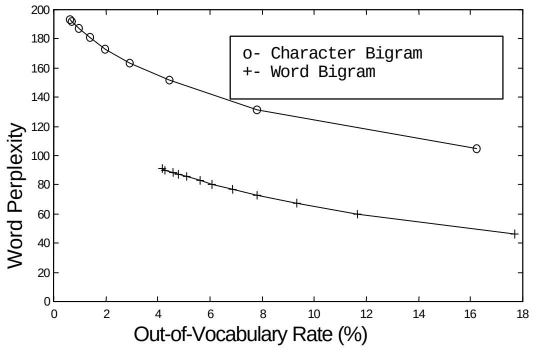

Figure 3 Perplexity vs. OOV % for Word LM & Character LM

Figure 3 compares the word perplexity of the word-level and character-level LM, as functions of the OOV. As usual, we have found that increasing the vocabulary size reduces the OOV rate. We have also found that the word-level bigram consistently results in lower word perplexity than the character-level bigram, suggesting that the word-level LM maybe more appropriate for speech recognition.

4.4 Comparison of smoothing techniques

We have investigated the use of different smoothing methods, with the goal of reducing the perplexity. Specifically, we have explored four different discount methods: linear, absolute [6], Witten Bell [7] and Good Turing (GT) [8]. As Figure 4 shows, both word bigrams and character bigrams have about 30% decrease in perplexity for all the discount methods. GT is the most effective among all four different methods. Specifically, the GT-smoothed word bigram and character bigram perplexities are 90.88 and 193.09, respectively. These results compare favorably to the simple backoff LM.

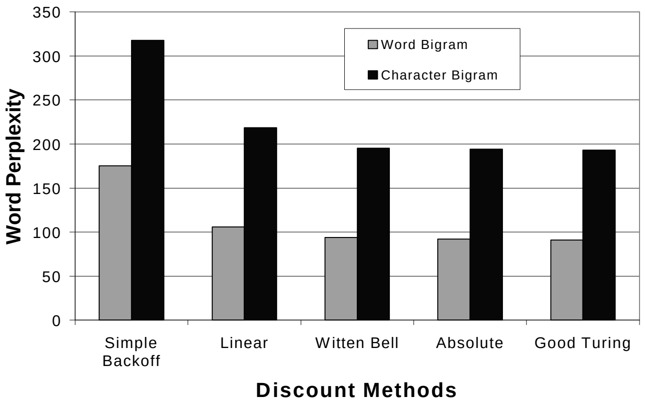

Figure 4 Comparison of different discounting methods for the Call Home task

5. SYSTEM EVALUATUION

Finally, we have incorporated our language models into our continuous HMM-based speech recognizer. The results are shown in Table 6. The word LM out-performs the character LM by 15.5% (31.46% vs. 26.56%). This result supports our perplexity and OOV analysis for the LMs in section 4.3. Moreover, by using GT smoothed word LM, we have increased the syllable accuracy from 29.72% to 31.46%, which is a 5.85% improvement when compared with simple backoff LM. However, the improvement for character LM is less significant. The syllable accuracy increases from 26.13% to 26.56%, which is only 1.64% improvement.

Language model	Word Perplexity	Syllable Acc.%	Character Acc.%
No-LM	N/A	20.17%	N/A
Simple Backoff Character Bigram	317.93	26.13%	21.89%
GT Smoothed Character Bigram	193.09	26.56%	22.61%
Simple Backoff Word Bigram	175.13	29.72%	25.73%
GT Smoothed Word Bigram	90.88	31.46%	27.72%

Table 6: Speech recognition results using different language models.

6. CONCLUSION

The following summarizes our findings:

1. Word LM consistently results in lower word perplexity than character LM, suggesting that word LM may be more appropriate for Mandarin Chinese speech recognition.

2. By using character LM instead of word LM, the vocabulary size is reduced from 5774 to 2098, which is a 63% reduction. Also, the out-of-vocabulary (OOV) rate is reduced from 4.15% to 0.62%. However, experiments show that character level LM results in lower accuracy than word LM by 7.3% due to the loss of linguistic information This results support our perplexity and OOV analysis of both LMs in our first finding.

3. The perplexity for word LM or character LM is reduced by about 30%, when GT smoothing is adopted with the backoff technique. Word LM increases its syllable accuracy by 5.85%, whereas character LM increases its syllable accuracy only by 1.64%.

7. ACKNOWLEDGEMENTS

This project is partially supported by Sir Edward Youde Memorial Fund.

8. REFERENCES

[1] L.S. Lee, et al, "Golden Mandarin(I)-a real time Mandarin Speech Dictation Machine for Chinese Language with Very Large Vocabulary", IEEE Trans. On Speech and Audio Processing, vol. 1, no. 2 P.P. 158-179, April 1993.

[2] R. Agarwal, B.Wheatley, Y. Muthusamy, and T.Staples, "Diagnostic Profiling for Speech Technology Development: Call Hone Analysis", Proceedings of Speech Research Symposium, P.P 131-137, June, 1995.

[3] L.R. Bahl, F. Jelinek, & R.L. Mercer, "A Maximum Likelihood approach to continuous Speech Recognition", IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-5(2): 179-190, 1983.

[4] H.Y. Leung, C.Y. Choy, Hong C. Leung, "Charcteristics of Chinese Language Models for Large Vocabulary Telephone Speech. 6th Eruopean Conference on Speech Communication & Technology.

[5] Kyuwoong Hwang, "Vocabulary Optimization Based on Perplexity", IEEE Trans ASSP, 1997.

[6] Hermann Hey, Ute Essen & Reinhard Kneser, "On the Estimation of 'Small' Probabilities by Leaving-One-Out", IEEE Transactions of Pattern Analysis & Machine Intelligence, vol. 17, No.12, p1202-1212, Dec. 1995.

[7] Timothy C. Bell, J.G. Cleary & Ian H. Witten, "Text Compression", Englewood Cliffs, N.J, Prentice-Hall, 1990.

[8] Good I.J., "The population frequencies of species and the estimation of population parameters.", Biometrika,40:237-264, 1953.

[9] R. Rosenfeld, "Adaptive Statistical Language Modeling: A Statistical approach PhD Thesis", School of Computer Science, Carnegie Mellon university, April 1994.