INEQUALITY MAXIMUM ENTROPY CLASSIFIER WITH CHARACTER FEATURES FOR POLYPHONE DISAMBIGUATION IN MANDARIN TTS SYSTEMS

Xinnian Mao[1], Yuan Dong[1, 2], Jinyu Han[2], Dezhi Huang[1] and Haila Wang[1]
[1]France Telecom R&D Center (Beijing), Beijing, 100080, P.R.China
{xinnian.mao, yuan.dong, dezhi.huang, haila.wang}@orange-ft.com
[2]Beijing University of Posts and Telecommunications, Beijing, 100876, P.R.China
yuandong@bupt.edu.cn; jinyuhan@gmail.com

ABSTRACT

Grapheme-to-phoneme (G2P) conversion is an important component in TTS systems. The difficulty in Chinese G2P conversion is to disambiguate the polyphones. In this paper, we formulate the polyphone disambiguation problem into a classification problem and propose a language independent classifier based on maximum entropy to address the issue. Furthermore, we introduce inequality smoothing to alleviate data sparseness and exploit language independent character features as linguistic knowledge. Experimental results show that the character features perform as well as the language dependent features such as words and part-of-speech, compared with the widely-used Gaussian smoothing, the inequality smoothing can greatly reduce the active features used in the classifier and achieve better performance. Our classifier achieves 96.35% in term of overall accuracy, greatly superior to 81.22% by using high-frequent "pinyin"(Romanization of Chinese phoneme). Finally, we explore to merge all key polyphones into 6 groups and find that the overall accuracy only decreases about 2% and the active features are reduced more than 33% further.

Index Terms—Grapheme-to-phoneme conversion, Maximum Entropy, Inequality Smoothing, Polyphone, Character Features

1. INTRODUCTION

Grapheme-to-phoneme (G2P) conversion is an important module in TTS systems, correct G2P conversion is the prerequisite to subsequent acoustic processing. Although many approaches including finite state transducer (FST) [5] and HMM [8], have been successfully applied to this task in alphabetic language such as English and French, they can not be transferred directly to pinyin language like Chinese and Japanese, because the formers are used to generate pronunciations for words which are out of vocabulary (OOV) and the latter are required to pick out the correct pronunciations (equal to pinyins in pinyin language) for polyphones.

As we known, polyphones existing in Chinese texts can be categorized into three types:

(1) Single-character polyphonic words (SCPW), such as 中(zhong1, zhong4) in the following sentence:

(1) Single-character polyphonic words (SCPW), such as 中 (zhong1, zhong4) in the following sentences:

项目/正在/进行/中(zhong1) (The project is going on).

他/中(zhong4)/了/大奖(He won the big award).

(2) Multi-character polyphonic words (MCPW), such as 朝阳(chao2yang2, zhao1yang2) in following sentences:

朝阳(chao2yang2)/的/方向/阳光/充足(There is plenty of sunlight in the direction of sun rising).

计算机/软件/是/朝阳(zhao1yang2)/产业(Computer software is the sun rising industry).

(3) Multi-character monophonic words (MCMW), such as 人参(ren2shen1) and 参加(can1jia1) in the sentences:

请/告诉/我/人参/的/食用/方法(Please tell me how to eat Panax).

大家/参加(can1jia1)/了/研讨会(All joined the seminar).

The slashes mark the boundaries of Chinese words. Obviously, MCMWs can be disambiguated easily by dictionaries look-up. So we focus on the first two in this paper.

At present, polyphone disambiguation is not an easy job in general. The common way of doing the job is using rules which are either hand-crafted [6] or learned automatically [7] [15], these approaches have three limitations: firstly, they do not generalize well to situations which we have not encountered. Secondly, these rules are usually language dependent. And thirdly, these rules often use words and/or part-of-speech as linguistic features, the errors during word segmenting or part-of-speech tagging will propagate to the polyphone disambiguation phrase, and these rules cannot adapt to different segmentation and tagging standards.

In this paper, we propose a maximum entropy (Maxent) classifier to disambiguate polyphones. We only use character features in the classifier which are language independent and easily computed. At the same time, we introduce inequality smoothing to alleviate data sparseness and embed feature selection seamlessly during the classifier training. Previous studies often process polyphones seperately, this is to say, a set of rules are acquired for each polyphone, which makes rule acquisition ineffective and the number of rules very large. In this paper, we exploit grouping the polyphones according to their pinyin frequency distribution.

The remainder of the paper is structured as follows. In Section 2, we introduce the inequality Maxent classifier.

Specially, we describe the smoothing measures and parameter optimization. In Section 3, we enumerate the features which may be used for the classifier. We evaluate the classifier in Section 4. Followed by exploiting grouping the polyphones in Section 5, we conclude the paper in Section 6.

2. MAXENT CLASSIFIER

In recent years, Maxent modeling [1] has been successfully applied in various classification tasks [3] [14]] due to its robustness, generality and superior performance. It stipulates that we should select the unique probability distribution which satisfies all known constraints but assume nothing about what we do not observe so that it maximizes the entropy subject to all known observations. In this paper, we formulate the polyphone disambiguation problem into a classification problem and employ Maxent classifier to address it. For a given polyphone, the classifier produces a probability for each pinyin and the probability can be calculated by the equation (1):

$$P(Seg|C)=\frac{1}{Z_{\lambda}(C)}\exp(\sum_{i}^{k}\lambda_i f_i(Seg,C)) \quad (1)$$

Where C represents the features of the polyphone and Seg is one of pinyin candidates of the polyphone. $Z_{\lambda}(C)$ is a normalization factor.

Standard Maxent requires that for each feature, the following constraint should be satisfied.

$$E\overline{p}[fi] = Ep[fi] \quad (2)$$

Where $E\overline{p}[fi]$ represents the empirical expectations of the ith feature and $Ep[fi]$ is the model expectation.

Although the standard Maxent model is already as uniform as possible given the above constraints, which alleviates data sparseness successfully, it is prone to over-fitting of training data because it also is a kind of maximum likelihood exponential model in certain contexts. Like other maximum likelihood methods, when the training data is sparse, smoothing is indispensable [11].

2.1. Gaussian Smoothing

Several smoothing algorithms have been proposed to overcome data sparseness and Chen and Rosenfeld [11] demonstrate that Gaussian smoothing performs as well as or better than all others. The Gaussian prior aims to penalize the features with excessively large or small weights using the equation (3); it is essential of relaxing the equality constraints in equation (2), which makes the model fit the training data less exactly.

$$E\overline{p}[fi] - Ep[fi] = \frac{\lambda i}{\sigma_i^2} \quad (3)$$

2.2. Inequality Smoothing

Recent progress on Maxent smoothing is inequality smoothing [2], it violates the equality constraints in equation (2) as follows:

$$-Bi \le E\overline{p}[fi]-Ep[fi] \le Ai$$

$$(Ai > 0 \wedge Bi > 0) \quad (4)$$

Empirical results demonstrate that it slightly outperforms the Gaussian smoothing, so we take advantage of it to smooth our classifier and contrast it with the frequently-used Gaussian smoothing. What's more, the inequality smoothing makes feature selection seamlessly embedded in parameter optimization, which makes the features with a zero weight be removed from the classifier without affecting its classification behavior, in such way, the active features will be reduced greatly.

2.3. Parameter Training

The parameters λi in equation (1) can be optimized with the iterative algorithms such as GIS [4], IIS [10] and general gradient-based algorithms. Malouf [9] compares these algorithms and reveals that limited-memory variable metric (LMVM, a kind of gradient-based algorithm) requires much less time and memory on four classification datasets, recently, Kazama and Tsujii [2] utilize BLMVM [12] (a variant of LMVM) to train the parameters and shows BLMVM can converge faster. So we employ it for parameter solving.

3. FEATURE SET

The success of applying Maxent depends to a large extent on the selection of suitable feature set. For our purpose, we want to exploit the features that are inexpensive to compute, language independent and effective as possible. We start with an exhaustive list of all features which might be useful and decided whether they are used in the classifier by experiments.

Character Features

(A) Uni-Gram: C_n *(n=-5,-4, -3,-2,-1, 1, 2, 3, 4, 5)*

(B) Bi-Gram: C_nC_{n+1} *(n=-5,-4, -3,-2, 1, 2, 3, 4)*

(C) Tri-Gram: $C_nC_{n+1}\ C_{n+2}$*(n=-5,-3, 1, 3)*

Where C refers to a Chinese character, it is obvious that the character features are language independent and can be acquired from the corpus directly; now, the work we need do is to validate their effective for the classifier.

Word and Part-of-Speech Features

(D) Word: W_n *(n= -3,-2,-1, 1, 2, 3)*

(E) Neighbor Part-of-Speech: P_n *(n= -3,-2,-1, 1, 2, 3)*

Part-of-Speech Itself

Besides the features mentioned above, we observe that the part-of-speech of the polyphone itself has strong discriminative ability for pinyin selection, so we also use it as the linguistic sources for the classifier.

(F) Part-of-Speech Itself

4. EVALUATION

4.1. Corpus and Key Polyphones

There are 804 SCPWs in GKB [13]. After investigating their distributions in the 2000th People Daily corpus with pinyin transcription annotated. We find 608 SCPWs occur at lease once which account for 8.631% of the whole corpus and their distributions are similarly to [15]. We choose 76 SCPWs from the top-270 SCPWs on the condition that the occurrence of each SCPW is not less than 100 times and its high-frequent pinyin is not more than 98%, the cumulative frequencies of the top-270 SCPWs account for 98.45% of all the 608 SCPWs. Another 109 MCPWs exist in GKB and 107 occur at once in the corpus, which accounts for 0.258% of the whole corpus. By the similar criterion, we select 31 MCPWs from top-38 MCPWs whose cumulative frequencies account for 96%. In sum, we choose these 107 key polyphones as targets for the classifier to disambiguate.

All the samples in the corpus of each target are extracted and divided into two parts: 70% and 30% randomly. The training set and the testing set of each target are randomly extracted from the two parts respectively. For each target, we extract at most 3000 samples for training and 1500 samples for testing.

4.2. Training and Testing

We train one inequality classifier for each target using the features defined in Section 3 and compute A_i and B_i by equation (5) as follows:

$$A_i = B_i = W/L \qquad (5)$$

(L represents the number of the samples)

We investigate the width factor W in three points: 1e-2, 1e-3 and 1e-4, for studying its effectiveness more deeply, another range from 0.1 to 1.0 with the step of 0.1 is also searched. For each target, we measure the accuracy with the percentage correctly classified samples; and we evaluate the overall accuracy by averaging the accuracy of each target, and the base line is 81.22% when we use the most frequent pinyin.

4.2.1. Effect of Features

We evaluate our classifier with a variety of classifier configurations (different set of features and different ways of using features, as indicated in the section 3). Table 1 displays the experimental results. The first column is the features and their combinations and the others are the overall accuracy using part-of-speech itself features (F) or not.

Feature	Accuracy(-F)	Accuracy (+F)
A	90.90	96.35
A+B	90.93	96.35
A+B+C	90.97	96.36
D	89.96	96.33
E	89.68	95.79
D+E	90.46	96.05

Table 1: Overall accuracies under each configuration

From the table, some observations can be obtained:

- The part-of-speech itself can make great contributions to performance improvement; comparing the second column (using features without the part-of-speech itself) with the third column (using the part-of-speech itself) under each kind of feature configuration, we find it enhances the accuracy by about 6%, which indicates that the part-of-speech itself is necessary for the classifier.
- Character features can achieve competitive performance with word and part-of-speech features. It is especially meaningful because the character features are language independent and have nothing to do with word segmentation and part-of-speech tagging. Another factor we should notice that the overall accuracies of D (word features), E (part-of-speech features) and D+E (their combinations) are obtained on the condition they are exacted from annotated corpus, which means all of them are correct. If they are obtained by word segmenter and part-of-speech tagger, the overall accuracies will decrease further.
- All the configurations we have tested achieve roughly the same performance. Among them, the accuracy on the part-of-speech features is the smallest, this mean that only part-of-speech features are not enough for the classifier. And we can conclude that feature combinations have little impact on the performance improvement.

4.2.2. Effect of Smoothing

To demonstrate the superiority of the inequality smoothing, we compare it with the frequently-used Gaussian smoothing. We train Gaussian classifier similar to the inequality classifier and search the σ used in equation (3) from 50 to 400 with the step of 50. Table 2 lists the results on the features A, D and E, from which, we confirm that inequality classifier slightly outperforms the Gaussian classifier and greatly reduce active features.

T	Accuracy (+F)		Feature Number (+F)		
	#G	#I	#G	#I	Reduction
A	95.89	96.35	124108	10208	12 times
D	95.77	96.33	151335	9374	16 times
E	94.79	95.79	28905	4545	6 times

Table 2: Inequality smoothing vs Gaussian smoothing
#G: Gaussian smoothing; #I: Inequality smoothing

5. POLYPHONE MERGING

Up to now, polyphones are usually dealt with separately. For applications with memory-constrained environment (embedded TTS on a cell phone for example), such applications require a balance between the need for small model, fast computation and optimal accuracy. Training each classifier for each polyphone requires more memory to load the model trained. In this section, we try to merge them into groups according to their pinyin frequency distribution. Among the 107 polyphones, 103 polyphones have two main pinyins and the other 4 have three main pinyins. We merge the 103 into 5 groups and put the other 4 into the 6th group. Table 3 lists the criterion for grouping.

Group	The first class	The second class
1st	0<LF<=10%	90%=<HF<=98%
2nd	10%<LF<=20%	80%=<HF<90%
3rd	20%<LF<=30%	70%=<HF<80%
4th	30%<LF<=40%	60%=<HF<=70%
5th	40%<LF<=50%	50%=<HF<60%

Table 3: Grouping criterion

Correspondently, for each group, we merge their training samples and testing samples together to form the new training set and testing set. And we define the low frequency (LF in table 3) pinyin as the first class and the higher (HF in table 3) as the second class. The same features are used and same experiments are conducted. Table 4 shows the overall accuracy as well as the number of the active features. The results indicate that the performance only decreases about 2% and the active features are reduced by at least 33%.

T	Accuracy(+F)			Feature Number (+F)		
	#M	#S	#Dec	#S	#M	#Dec
A	94.71	96.43	1.72	9790	5404	44.8%
D	94.97	96.40	1.43	9253	6204	33.0%
E	93.81	95.86	2.05	4462	2674	40.1%

Table 4: Merging polyphones into 6 groups
#M: merging; #S: Separate; #Dec: Decrease

6. CONCLUSION AND FUTURE WORK

In this paper, a language independent classifier based on inequality Maxent using character features has been constructed. To our knowledge, our work is the first attempt to employ Maxent to disambiguate polyphones, our work demonstrates that the character features can achieve comparatively performance with the language dependent features such as word and part-of-speech features; we also validate the strong discriminative ability of the part-of-speech itself; and the inequality smoothing performs slightly better than the Gaussian smoothing. With the help of its feature selection ability, the unnecessary features can be removed from the classification model without changing the classification behavior, which makes the classification model smaller. Experimental results also reveal that feature combinations have litter impact on performance improvement and polyphone merging can reduce active features further without decreasing the performance much. But in our experiments, we assign the same width factor to each kind of feature, which does not make full use of the advantage of the inequality smoothing. So employing more sophisticated width factor is one of the future works.

We believe that our approach can be directly transferred to other pinyin language like Japanese, and we also believe it can handle the polyphonic phenomena in alphabetic language such as English and French. For example, the word 'record' pronounces as [rɪˈkɔːd] and [ˈrekɔːd] when its part-of-speech is noun and verb respectively (For English, the character features can be converted into word features). related experiments have not been conducted to validate our hypothesis because of lack of annotated corpora.

7. REFERENCE

[1] A.L. Berger, S. A. Della Pietra and V.J. Della Pietra, “A maximum entropy approach to natural language processing”, *Computational Linguistics*, 22(1):39-71, 1996.

[2] J. Kazama and J. Tsujii, “Maximum entropy models with inequality constraints: a case study on text categorization”, *Machine Learning*, 60: 159-194, 2005.

[3] J. K. Hong-Kwang and Y. Q. Gao, “Maximum entropy direct models for speech recognition”, *In Proc. ASRU-* 2003, pp.1-6.

[4] J.N. Darroch and D. Ratcliff, “Generalized iterative scaling for log-linear models”, *The Annals of Mathematical Statistics*, 43 (5):1470-1480, 1972.

[5] L. Galescu and J. Allen, “Bi-directional conversion between graphemes and phonemes using a joint ngram model”, *In Proc. of the 4th ISCA Tutorial and Research Workshop on Speech Synthesis*, 2001, Perthshire, Scotland.

[6] M. Zheng, L.H. Cai, “A New Rule-based Method of Automatic Phonetic Notation on Polyphones”, *ICSP-2004*.

[7] M. Zheng, L.H. Cai, “Grapheme-to-Phoneme conversion based on a fast TBL algorithm in mandarin TTS system”, *ICNC-FSDK-2005*.

[8] P. Taylor, “Hidden markov models for grapheme phoneme conversion”, *In Proc. of Interspeech-2005*, Lisbon, Portugal, pp. 1973-1976.

[9] R. Malouf, “A comparison of algorithms for maximum entropy parameter estimation”, *In Proceedings of CoNLL-2002*, pp.49-55.

[10] S.D. Pietra, V. Della Pietra, and J. Lafferty, “Inducing features of random fields”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(4):380-393, 1997.

[11] S.F. Chen and R. Rosenfeld, “A survey of smoothing techniques for ME models”, *IEEE Transactions on Speech and Audio Processing*, 2000, 8(2):37-50.

[12] S. J. Benson and J. J. Mor´e, “A limited memory variable metric method for bound constrained optimization”, *Technical Report ANL/MCS-P909-0901*, Argonne National Laboratory, 2001.

[13] S.W. Yu, etc, The Grammatical Knowledgebase of Contemporary Chinese --- A Complete Specification. Tsinghua University Press, Beijing, 2002.

[14] Y. Dong, M. Mahajan, P. Mau and A. Acero, “Maximum entropy based generic filter for language model adaptation”, *In Proceedings of the ICASSP*-2005, pp 597-600.

[15] Z.R. Zhang, M. Chu and E. Chang, “An efficient way to learn rules for grapheme-to-phoneme conversion in Chinese”, *In Processing of ISCSLP-2002*, pp: 59-64.