International Journal of Knowledge and Language Processing
Volume **2**, Number 1, January **2011**

www.ijklp.org
KLP International ©2011 ISSN 2191-2734
pp. **59-69**

Polyphone Disambiguation with Machine Learning Approaches

Jinke Liu[1,3], Weiguang Qu[1,3], Xuri Tang[2], Yizhe Zhang[1,3], Yuxia Sun[1]

[1] School of Computer Science and Technology
Nanjing Normal University
Nanjing, Jiangsu, 210046, China
{ lyliujinke; wgqu_nj }@163.com

[2] School of Chinese Language and Literature
Nanjing Normal University
Nanjing, Jiangsu, 210097, China
xrtang @126.com

[3] The Research Center of Information Security and Confidentiality Technology of Jiangsu Province
Nanjing, Jiangsu, 210097, China

Received December 2010; revised January 2011

ABSTRACT. *To obtain a more satisfactory solution to polyphonic word disambiguation, five different classification models, namely RFR_SUM, CRFs, Maximum Entropy, SVM and Semantic Similarity Model, are employed for polyphonic disambiguation. Based on observation of the experiments of these models, an additional improving ensemble method based on majority voting is proposed, which achieves an average precision of 97.39%, much better than the results obtained in previous literatures.*
Keywords: Polyphone disambiguation, Ensemble model, RFR_SUM, CRFs, Maximum Entropy, SVM, Semantic Similarity

1. **Introduction.** A TTS(Text-to-Speech) system transforms a sequence of characters into a sequence of Chinese Pinyin. It generally includes two modules: text normalization and text-to-phoneme conversion. The core of the first module is polyphone disambiguation, which still awaits a satisfactory solution. Furthermore, polyphony is one of the crucial problems in Chinese TTS systems and is common in Chinese. In some worst cases, one character may have up to five different types of pronunciation. For instance, the character "和" may be spoken in one of the following sound: hé、hè、hú、huó and huò[1]. According to [5], among the 10 most frequent characters, 6 of them are polyphones: "的", "一", "了", "不", "和", "大". Thus correct determination of how a character is read can improve TTS performance to a great extent.

[1] These are Chinese Pinyin, annotated with tones.

Modern Chinese Dictionary2 collects 1036 polyphonic characters and 580 polyphonic words. However, not all of them are frequently used. About 180 characters and 70 words take 95% and 97% of cumulative frequencies respectively in actual language use [1]. Among these 180 characters and 70 words, only 41 characters and 22 words needs disambiguation. To put it in another way, if the polyphonic ambiguity of these 41 characters and 22 words are successfully solved, the problem of polyphone ambiguity in Chinese should be largely solved. The present study is to tackle these polyphones with machine-learning approach.

The choice of pronunciation of polyphones is determined by language convention and semantic content. There are currently two paradigms to approach the ambiguity: rule-based paradigm and statistics-based paradigm. Recent years have witnessed a growing number of researches on polyphone disambiguation with statistical machine learning. [1] proposes to use the ESC(extended stochastic complexity)-based stochastic decision list to learn pronunciation rules for polyphones. In [2], polyphones are divided into two categories and disambiguate on POS level and semantic level separately. [3] presents a rule-based method of polyphone disambiguation, integrated with SVM-based weight estimation and [4] makes use of maximum entropy model to solve polyphone ambiguity.

This paper proposes an ensemble-learning approach for polyphonic disambiguation. The approach experiments with five machine learning models in polyphonic disambiguation and ensembles these five models with majority-voting to determine the final pronunciation of polyphones. The rest of the paper is organized as follows. Section II gives an overview of the five models. In Section III, experiments with the five models and the ensemble learning are described in detail. IV compares the results obtained by ensemble model with related documents, which is followed by conclusions and plans for future work in Section V.

2. **Machine Learning Models.** In this section, we describe the principle of models used in the experiments.

2.1. RFR_SUM Model. Qu[6] presents the concept of relative frequency ratio (RFR) and proposes the RFR_SUM model which disambiguates with context before and after the word in question. RFR of a word is the frequency ratio associated with relative position to the ambiguous word and is calculated between local frequency and global frequency. In RFR_SUM, the context is categorized into pre-context, the context before the word in question, and post-context, the context after the word in question. Thus the context of the word W_i in question can be characterized by the following formula:

$$SUM_m = \sum_{i=-l}^{-k} f_{m,left}(W_i) + \sum_{i=l}^{k} f_{m,right}(W_i)$$

Disambiguation can thus be done by comparing individual SUMs in different occurrences.

In fact, the polyphony disambiguation is from different context. We can make use of much information of context, such as the words in pre-context and post-context, the

[2] Institute of Linguistics of Chinese Academy of Social Sciences. Revised edition 3, 1996.7.

position of these words and the special sequence between these words, to eliminate polyphony disambiguation. So the RFR_SUM model would be used to process the polyphonic disambiguation

2.2. Conditional Random Fields. Conditional Random Fields, presented firstly by Laferty[10], is a conditional probability model used for tagging and partitioning sequence data. The model is an undirected graph that can calculate the conditional probability of output node based on the conditions of given input node. The input sequence x and output sequence y can be defined as a linear CRF model, defined as below:

$$P(y \mid x) = \frac{1}{Z(x)} \exp[\sum \lambda_k f_k(y_{i-1}, y_i, x) + \sum \mu_k g_k(y_i, x)]$$

where fk is the state transition function at position i and i-1 in sequence x, and gk is the state feature function at position i in sequence x. λ and μ are the weights of the function and the Z is normalization factor.

In CRF, normalization is not made in every node, globally on the whole features. It has the ability to express the long distance dependence and overlapping. At the same time, the relevant field knowledge is well included in the CRF model, gaining global optimal value. The tool kit adopted in our experiment is the CRF++ (version 0.50)[3] created by TakuKudo.

2.3. Maximum Entropy Model. Maximum entropy is proposed by Jaynes[11] in 1957 and firstly applied to NLP in Berger's paper[12] in 1996. The model is a method based on maximum entropy theory, in which the category with maximum entropy is selected as the optimal. In maximum entropy model, probability distribution is estimated, and hypothesis is presented if the model meets restriction condition. In other words, the condition probability whose entropy is maximum is selected.

The model has been applied in various fields of NLP, such as word segmentation, POS tagging and semantic disambiguation. In this model, the problems to be solved are determination of feature space(issue field), choice of feature(searching for restriction conditions), establishment of statistical model (establishing model whose entropy is maximum based on maximum entropy theory), system input(features) and system output(optimal model whose entropy is max).

In our experiment, the toolkit developed by Zhang Le is used.[4] The experiment procedure consists of four steps: training (input feature files extracted from train corpus), outputting training model, identification (input feature files extracted from test corpus) and outputting predicted results.

2.4. Support Vector Machine. Recent years have witnessed Support Vector Machines (SVMs) as a prevailing machine learning tool applied in various fields. They are a set of related supervised learning methods used for data analysis, pattern recognition, classification or regression analysis. The original SVM algorithm is proposed by Vapnik[13]

[3] Accessible at http://crfpp.sourceforge.net
[4] Accessible at http://homepages.inf.ed.ac.uk/s0450736/ME_toolkit.html

in 1995 for pattern recognition, which seeks to find a hyperplane which has the largest distance to the nearest training data points, called support vectors, and thus best divides the two categories.

Considering N-dimensional space, Y={1,-1} stands for two categories. Training sample set is translated into (xi,yi), i=1,2,…n. The above xi signifies vector of sample i in feature space and the yi belongs to Y. We suppose linear discriminating function as g(x)=w·x+b, so the interface can be indicated by w·x+b=0. All samples are made to fulfill the condition of g(x)≥1 by normalization, then the distance of two categories is indicated as 2/||w||. Furthermore, the maximum distance is required to acquire, so we obtain the formula of $\min \frac{1}{2} \| w \|^2$ whose restriction condition is yi[(w·x)+b]-1≥0,i=1,2…n.

We adopt libSVM implemented by Doctor Lin Chih-Jen of Taiwan University for experiment5.

2.5. Semantic Similarity Model. The word similarity calculation based on HowNet has been widely studied. In this paper, we employ the calculation method presented by Liu Qun. The similarity of two words can be reduced to the similarity of two concepts. The idea can be denoted as below:

$$sim(w_1, w_2) = \max_{i=1..n,\, j=1..m} sim(s_{1i}, s_{2j})$$

Moreover, semantic similarity model calculates semantic similarity between sentences and then employs K nearest neighbor classifier to decide which category the polyphone should fall into. The tool for word similarity calculation is based on HowNet and the algorithm is proposed in [7] and [8]. Given two sentences SEN1 and SEN2, the procedure of disambiguation can be briefly described as below:

a. Given the polyphone word W and its position i and j in SEN1 and SEN2, and a window size N, four word set can be obtained: frontsen1= W_{i-1}^{i-N} , backsen1=W_{i+1}^{i+N} , frontsent2=W_{j-1}^{j-N} and backsen2=W_{j+1}^{j+N} .
b. Obtain front context semantic similarity FrontSim(frontsent1, frontsent2) and back context semantic similarity BackSim (backsent1, backsent2).
c. Obtain sentence semantic similarity：
SenSim = FrontSim(frontsent1, frontsent2)＋BackSim(backsent1, backsent2).
d. Apply K nearest neighbor classifier to classify W.

3. **Experiments and Analysis.**

3.1. Experiment data and evaluation. The 1998 People's Daily Corpus compiled by Peking University is used for experiment. The corpus contains half a year's newspaper of People's Daily in 1998, with a total of 13 million words. 40 polyphonic characters and 20 polyphonic words are selected from related literature for study. For each polyphone, 1600 sample sentences are retrieved from the corpus, 75% of which are used as training set and

[5] Accessible at http://www.csie.ntu.edu.tw/~cjlin/libsvm

25% of which as testing set.

Due to space limitation, only 18 are selected in the paper for illustration. Table 1 gives the detailed experiment data, where Low stands for the number of low-frequency pronunciation sentences, High for the number of high-frequency pronunciation sentences, and B_L for baseline. Note that the average baseline is 74.23%. The results are evaluated by P, defined as follows:

$$P = \frac{\text{numberof correct samples}}{\text{numberof overallsamples}} \qquad (1)$$

TABLE 1. Sample sentences statistical

Word	Low	High	B_L
背	202	332	62.20
长	300	1396	82.31
重	232	693	74.92
得	512	735	69.65
干	90	1319	93.61
种	513	2208	81.15
倒	161	194	54.65
曾	412	2414	85.42
还	580	2765	82.66
只	754	2913	79.43
处	277	947	77.37
担	56	169	75.11
为	684	744	52.10
藏	113	121	51.71
合计	12	134	91.78
孙子	44	109	71.24
朝阳	85	138	61.88
地方	1281	1623	55.88

3.2. RFR_SUM experiment results. In RFR_SUM model, the relative frequency ratio of word, the relative frequency ratio of POS, and the relative frequency ratio of the combination of POS and word are independently experimented within a window of 5 for disambiguation. Table 2 gives the average precisions in the three experiments.

From the Table, the average precision of POS is 5.69% higher than word, so POS plays a main role in identifying pronunciation. Apparently the combination of word and POS is the best in effect.

TABLE 2. Results of RFR_SUM

Feature	Word	POS	Word&POS
P	87.75	92.44	94.65

3.3. CRFs experiment results. The advantage of CRFs is that new features can be added freely. Four feature templates and the results obtained in the experiment are given in Table 3. As is seen in Table 3, Template 1 and Template 2 use word form and POS respectively. Both word form and POS are used as features in Template 3. Moreover, Template 4 uses the binding of word and POS respectively in addition to word form and POS. However, the average precision of Template 4 drops 0.69% than that of Template 3.

TABLE 3. Results of CRFs

Template 1	**Template 2**	**Template 3**	**Template 4**
U1:%x[-2,0]	U1:%x[-2,1]	U1:%x[-2,0]	U1:%x[-2,0]
U2:%x[-1,0]	U2:%x[-1,1]	U2:%x[-1,0]	U2:%x[-1,0]
U3:%x[0,0]	U3:%x[1,1]	U3:%x[0,0]	U3:%x[0,0]
U4:%x[1,0]	U4:%x[2,1]	U4:%x[1,0]	U4:%x[1,0]
U5:%x[2,0]		U5:%x[2,0]	U5:%x[2,0]
		U6:%x[-2,1]	U6:%x[-2,1]
		U7:%x[-1,1]	U7:%x[-1,1]
		U8:%x[1,1]	U8:%x[1,1]
		U9:%x[2,1]	U9:%x[2,1]
			U10:%x[-2,0]/%x[-1,0]
			U11:%x[1,0]/%x[2,0]
			U12:%x[-2,-1]/%x[-1,-1]
			U13:%x[1,-1]/%x[2,-1]
92.46	94.14	95.27	94.58

3.4. SVM experiment results. In the experiment of SVM, the relative frequency ratio of word and POS in a window size of 4 is used as vector features. Table 4 gives the results obtained through different kernel functions.

In table 4, it can be seen that kernel function has an effect on the disambiguation result, but the difference is not significant. All kernel functions have a precision close to 90% and RBF kernel function achieves the best precision, reaching 91.86%.

TABLE 4. Results of SVM

K_F	**Linear**	**Polynomial**	**RBF**	**Sigmoid**
P	90.33	90.47	91.86	89.94

3.5. Semantic similarity model experiment results. As the semantic similarity model uses K-nearest neighbor classifier for disambiguation, it is obvious that the parameter K will affect the final outcome. Table 5 lists out the experiment results for different Ks. The best experiment results are obtained when the value of K is 4.

TABLE 5. K values and experiment results

	3	4	5	6
	90.56	91.23	91.05	90.37

3.6. Ensemble learning. Table 6 lists out the best experiment results of 18 polyphones obtained with the five models discussed above, in which AVG stands for average precision of all 60 polyphones obtained in the models.

TABLE 6. Results of five models and majority voting

Word	**S_S**	**SVM**	**M_E**	**RFR_SUM**	**CRF**	**M_V**
背	79.59	63.27	69.39	93.88	83.67	91.84
长	89.31	85.89	90.32	92.74	92.94	91.53
重	89.89	95.88	91.01	94.76	97.00	97.0
得	90.42	88.12	89.78	90.23	87.33	91.67
干	93.49	77.22	95.86	81.66	94.67	95.27
种	95.32	96.93	95.05	97.33	96.93	98.40
倒	79.63	85.53	80.26	97.76	89.47	94.74
曾	93.33	99.68	94.81	98.10	99.79	99.79
还	98.35	81.87	98.13	98.13	99.45	99.34
只	95.31	98.90	93.92	98.31	99.40	99.00
处	91.94	93.62	94.97	96.64	98.32	98.32
担	82.86	94.29	95.71	97.14	94.29	94.29
为	83.99	85.92	85.04	81.82	89.44	90.32
藏	75.33	76.67	78.33	93.33	90.00	96.67
合计	92.42	96.97	95.45	98.48	93.94	98.48
孙子	94.12	97.06	95.59	95.59	97.06	97.06
朝阳	91.04	91.04	85.07	94.03	97.01	98.51
地方	92.47	93.49	90.43	94.26	96.68	97.45
ARG	91.23	91.86	92.64	94.65	95.27	96.78

As can be seen in Table 6, results of five models (S_S, SVM, M_E, RFR_SUM and CRF) are diverse and complementary. For some polyphonic words, their precisions are probably very low in some models, but very high in some other models. For instance, precision of '背' is only 63.27% in SVM model, but goes up to 93.88% in RFR_SUM model.

In addition, in order to make comparison among the five models, the features of five models are all from five widows of central word context. So we can learn that the average precision of CRF and RFR_SUM is higher than other's.

Also, the nature of instability in individual model and complementation among different models lead us to consider and adopt ensemble method for the final disambiguation. The principle adopted in the ensemble method is majority voting, namely the pronunciation which receives the most votes in the five models is chosen as the final pronunciation. The

experiment result (M_V) given in Table 6 obtains an average precision of 96.78%, showing that the ensemble effect is better than any single model, because it well ensembles advantages of every model and effectively eliminates the instability of individual model.

TABLE 7. Results of two majority voting experiments

Word	M_V1	M_V2
背	91.84	92.12
长	91.53	91.53
重	97.0	97.56
得	91.67	92.75
干	95.27	95.87
种	98.40	98.92
倒	94.74	95.34
曾	99.79	99.79
还	99.34	99.34
只	99.00	99.22
处	98.32	98.58
担	94.29	94.69
为	90.32	92.65
藏	96.67	97.43
合计	98.48	98.48
孙子	97.06	97.88
朝阳	98.51	98.76
地方	97.45	98.75
ARG	96.78	97.39

4. Two Majority Voting Experiment.

4.1. **Original experiment.** It can be seen in the above analysis that the precision of single model ranges from highness of 99% to lowness of 63%. To one polyphone, its precision is probably very low in one model, but is very high in another model. We adopt the ensemble method of majority voting based on the five experiment results, improving the effect of polyphone disambiguation. The majority voting experiment is introduced as follows.

The basic principle of majority voting method is described as below. When polyphone is tagged by five models, the voting is given to the pronunciation that is tagged by max number models and the max voting pronunciation is seen as the pronunciation of polyphone.

The formula is $value\_vote_i = num\_vote_i * W_i$, where num_votei refers to the count of models voting and Wi =1.

4.2. **Improving experiment.** The weight of every model is imported and the method of

getting weight is described as below:

$$W_i = \lambda_i \frac{P_i}{\sum_{i=1}^{n} P_i}.$$

Where P_i stands for average precision of every model, W_i for weight of every model and λ_i for weight factor which ranges from 1 to 2.

Finally, $value\_vote_i = num\_vote_i * W_i$, where num_votei refers to the count of models voting.

The pronunciation which receives the most value in the five models is chosen as the final pronunciation and two experiment results are list in Table 7. It is obvious that the average of M_V2 is 0.61% higher than M_V1 and the weight is effective in disambiguation.

5. **Comparison.** In order to evaluate the effect of ensemble method, the experiment results reported in [1] and [4] are selected to contrast the results obtained in ensemble method in the present study. The contrast is given in Figure 1 and Figure 2, where ARG is the average precision of 13 polyphones in the Figure1 and Figure2 respectively.

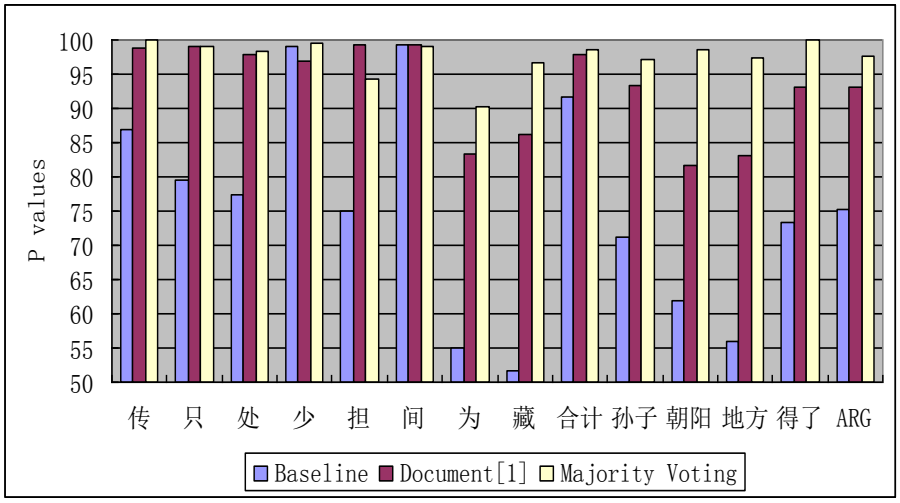

FIGURE 1. Comparison with [1]

Figure 1 and Figure 2 show that the average precision of ensemble method is 4.56% and 3.69% higher than [1] and [4] respectively. Low precision polyphones such as '藏', '朝阳' and '地方' in [1] and [4] gain very high precision in majority voting method. High precision polyphones still keep high in ensemble method experiment. Obviously, the ensemble model absorbs the merits of every model to improve low precision and keep high precision.

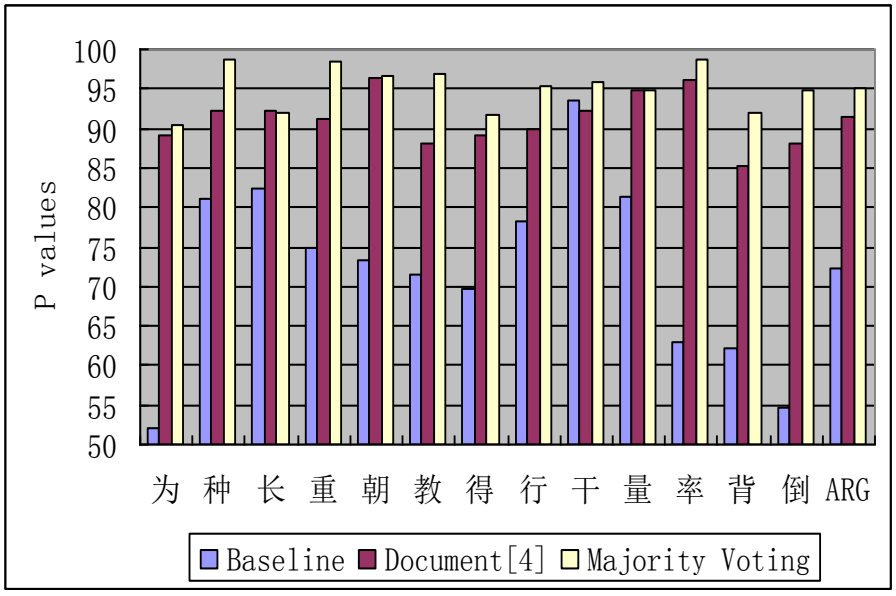

FIGURE 2. Comparison with [4]

6. **Conclusion.** This paper has applied five different classification models, namely RFR_SUM, CRFs, Maximum Entropy, SVM and Semantic Similarity Model, for the task of polyphonic disambiguation. The experiments show that CRF and RFR_SUM are able to simulate the human cognitive process more smoothly in disambiguation, and thus perform better than the other three models. However, the effects of single model are uneven, so the ensemble method is considered. The improving ensemble method is based on majority voting with the five models and reaches an average precision of 97.39%, which is better than the results of [1] and [4]. This shows that the ensemble model has more advantage than any individual model.

For the future work, we will continue searching for the better ensemble methods among the five models based on the present experiments. For instance, we plan to conduct more experiments to obtain a better ensemble method, to combine RFR_SUM and Semantic Similarity Model so as to improve the instability of RFR_SUM in conditions of sparse data. In addition, more knowledge will be introduced into the system[9].

Acknowledgment. This work is supported by Chinese National Fund of Natural Science under Grant 60773173, 61073119 and Jiangsu Province Fund of Natural Science under Grant BK2010547.

REFERENCES

[1] Zi-Rong Zhang, Min Chu, A Statistical Approach for Grapheme-to-Phoneme Conversion in Chinese, *Journal of Chinese Information Processing*, 2002.

[2] Ming Fan, Guo-ping Hu, Multi-level Polyphone Disambiguation for Mandarin Grapheme-Phoneme Conversion, *Computer Engineering and Applications*, 2006.

[3] Guo-ping Hu, Zhi-gang Chen, Ren-hua Wang, A Rule-Based Approach with SVM-Based Weight

Estimation for Phoneme Disambiguation of Polyphone, *Proceedings of the 20th International Conference on Computer Processing of Oriental Languages*, Shenyang, 2003.

[4] Fang-zhou Liu, Qin Shi, Maximum Entropy Based Homograph Disambiguation, *Proceedings of the 9th National Conference on Human-Machine Language Communications*, 2007.

[5] Yu-Qi Sun, Kai Zhang, Polyphone Study of Combination Based on rule and statistical, *Proceedings of the 5th National Conference on Human-Machine Language Communications*, 1998.

[6] Wei-guang Qu, *Disambiguation Study on Modern Chinese Word-Level Ambiguity*, Beijing: Science Press, 2008.

[7] Qun Liu, Su-Jian Li, Word Similarity Computing Based on How-Net, *The Third Symposium on Chinese Lexical Semantics*, Taipei, 2002.

[8] Zheng-Dong Dong, Qiang Dong, *How-Net*, http://www.keenage.com.

[9] Yao Liu, Zhifang Sui, Qingliang Zhao, Yongwei Hu, Research on Construction of Medical Ontology, *International of Knowledge and Language Processing*, vol.1, no.1, pp.19-35, 2010.

[10] John Lafferty, Andrew McCallum, Fernando Pereira, Conditional random fields: Probabilistic models for segmenting and labeling sequence data, *Proceedings of the 18th International Conference on Machine Learning*, San Francisco: Morgan Kaufmann, pp. 282-289, 2001..

[11] E. T. Jaynes, *Information theory and statistical mechanics, Phys. Rev.*, vol.106, no.4, pp.620-630, 1957.

[12] Adam L. Berger, Stephen A. Della Pietra, and Vincent J. Della Pietra, A maximum entropy approach to natural language processing, *Computational Linguistics*, vol.22, no.1, pp.39-71,1996.

[13] V. Vapnik, *The Nature of Statistical Learning Theory*, New York, Springer-Verlag, 1995.