Combining Neural Networks and Statistics for Chinese Word sense disambiguation

Zhimao Lu Ting Liu Sheng Li

Information Retrieval Laboratory of Computer Science & Technology School,
Harbin Institute of Technology
Harbin, China, 150001
{lzm, tliu}@ir.hit.edu.cn

Abstract

The input of network is the key problem for Chinese Word sense disambiguation utilizing the Neural Network. This paper presents an input model of Neural Network that calculates the Mutual Information between contextual words and ambiguous word by using statistical method and taking the contextual words to certain number beside the ambiguous word according to (-M, +N). The experiment adopts triple-layer BP Neural Network model and proves how the size of training set and the value of M and N affect the performance of Neural Network model. The experimental objects are six pseudowords owning three word-senses constructed according to certain principles. Tested accuracy of our approach on a close-corpus reaches 90.31%,, and 89.62% on a open-corpus. The experiment proves that the Neural Network model has good performance on Word sense disambiguation.

1 Introduction

It is general that one word with many senses in natural language. According statistics, there are about 42% ambiguous words in Chinese corpus (Lu, 2001). Word sense disambiguation (WSD) is a method to determine the sense of ambiguous word given the context circumstance.

WSD, a long-standing problem in NLP, has been a very active research topic,, which can be well applied in many NLP systems, such as Information Retrieval, Text Mining, Machine Translation, Text Categorization, Text Summarization, Speech Recognition, Text to Speech, and so on.

With rising of Corpus linguistics, the machine learning methods based on statistics are booming (Yarowsky, 1992). These methods draw the support from the high-powered computers, get the statistics of large real-world corpus, find and acquire knowledge of linguistics automatically. They deal with all change by invariability, thus it is easy to trace the evaluation and development of natural language. So the statistic methods of NLP has attracted the attention of professional researchers and become the mainstream bit by bit. Corpus-based Statistical approaches are Decision Tree (Pedersen, 2001), Decision List, Genetic Algorithm, Naive-Bayesian Classifier (Escudero, 2000)、Maximum Entropy Model (Adam, 1996; Li, 1999), and so on.

Corpus-based statistical approaches can be divided into supervised and unsupervised according to whether training corpus is sense-labeled text. Supervised learning methods have the good learning ability and can get better accuracy in WSD experiments (Schütze, 1998). Obviously the data sparseness problem is a bottleneck for supervised learning algorithm. If you want to get better learning and disambiguating effect, you can enlarge the size and smooth the data of training corpus. According to practical demand, it would spend much more time and manpower to enlarge the size of training corpus. Smoothing data is merely a subsidiary measure. The sufficient large size of training corpus is still the foundation to get a satisfied effect in WSD experiment.

Unsupervised WSD never depend on tagged corpus and could realize the training of large real corpus coming from all kinds of applying field. So researchers begin to pay attention to this kind of methods (Lu, 2002). The kind of methods can

overcome the sparseness problem in a degree.

It is obvious that the two kinds of methods based on statistic have their own advantages and disadvantages, and cannot supersede each other.

This paper researches the Chinese WSD using the model of artificial neural network and investigates the effect on WSD from input model of neural network constructed by the context words and the size of training corpus.

2 BP Neural Network

At the moment, there are about more than 30 kinds of artificial neural network (ANN) in the domain of research and application. Especially, BP neural network is a most popular model of ANN nowadays.

2.1 The structure of BP Neural Network

The BP model provides a simple method to calculate the variation of network performance cased by variation of single weight. This model contains not only input nodes and output nodes, but also multi-layer or mono-layer hidden nodes. Fig1.1 is a construction chart of triple-layer BP neural network. As it is including the weights modifying process from the output layer to the input layer resulting from the total errors, the BP neural network is called Error Back Propagation network.

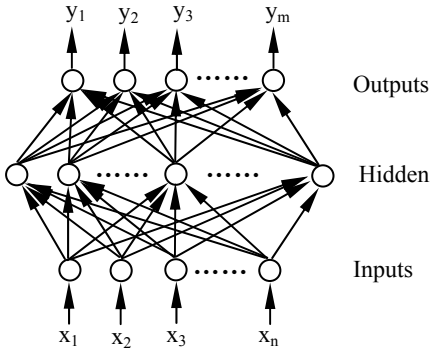

Fig. 1.1 BP Network

Fig.1.1 The structure of BP neural network Except for the nodes of input layer, all nodes of other layers are non-linear input and output. So the feature function should be differential on every part of function. General speaking, we can choose the sigmoid, tangent inspired, or linear function as the feature function because they are convenient for searching and solving by gradient technique. Formula (1) is a sigmoid function.

$$f(x) = \frac{1}{1+e^{-x}} \qquad (1)$$

The output of sigmoid function ranges between 0 and 1, increasing monotonically with its input. Because it maps a very large input domain to a small range of outputs, it is often referred to as the squashing function of the unit. The output layer and hidden layer should adopt the sigmoid inspired function under the condition of intervention on the output, such as confining the output between 0 and 1.

2.2 Back Propagation function of BP neural network

The joint weights should be revised many times during the progress of the error propagating back in BP networks. The variation of joint weights every time is solved by the method of gradient descent. Because there is no objective output in hidden layer, the variation of joint weight in hidden layer is solved under the help of error back propagation in output layer. If there are many hidden layers, this method can reason out the rest to the first layer by analogy.

1) the variation of joint weights in output layer

To calculate the variation of joint weights from input *i*'th to output *k*'th is as following:

$$\Delta w_{ik} = -\eta \frac{\partial E}{\partial w_{ik}} = -\eta \frac{\partial E}{\partial O_k} \frac{\partial O_k}{\partial w_{ik}} \qquad (2)$$

$$= \eta (t_k - O_k) f_2' O'_i = \eta \delta_{ik} O'_i$$

$$\delta_{ik} = (t_k - O_k) f_2' \qquad (3)$$

$$\Delta b_{ki} = -\eta \frac{\partial E}{\partial b_{ki}} = -\eta \frac{\partial E}{\partial O_k} \frac{\partial O_k}{\partial b_{ki}} \qquad (4)$$

$$= \eta (t_k - O_k) f_2' = \eta \delta_{ik}$$

2) the variation of joint weights in hidden layer

To calculate the variation of joint weights from input *j*'th to output *i*'th is as following:

$$\Delta w'_{ij} = -\eta \sum_{k=1}^{n} \frac{\partial E}{\partial w'_{ij}} = -\eta \sum_{k=1}^{n} \frac{\partial E}{\partial O_k} \frac{\partial O_k}{\partial O'_i} \frac{\partial O'_i}{\partial w'_{ij}} \qquad (5)$$

$$= \eta \sum_{k=1}^{n} (t_k - O_k) f_2' w_{ik} f_1' p_j$$

$$= \eta \delta_{ij} p_j$$

where: $\delta_{ij} = e_i f_1'$, $e_i = \sum_{k=1}^{n} \delta_{ik} w_{ik}$ (6)

$$\Delta b'_{ki} = \eta \delta_{ij} \quad (7)$$

3. The construction of WSD model

Under the consideration of fact that only numerical data can be accepted by the input and output of neural network, if BP neural network is used on WSD, the prerequisite is to vector the part of semantic meaning (words or phrases) and sense.

In the event of training BP model, the input vector P and objective vector O of WSD should be determined firstly. And then we should choose the construction of neural network that needs to be designed, say, how many layers is network, how many neural nodes are in every layer, and the inspired function of hidden layer and output layer.

The training of model still needs the vector added weight, output, and error vector. The training is over when the sum of square errors is less than the objection of error. Or the errors of output very to adjust the joint weight back and repeat the training.

3.1 To vector the vocabulary

WSD depends on the context to judge the meaning of ambiguous words. So the input of model should be the ambiguous words and the contextual words round them. In order to vector the words in the context, the Mutual Information (MI) of ambiguous words and context should be calculated. So MI can show the opposite distance of ambiguous words and contextual words. MI can replace every contextual word. That is suitable to as the input model. The function of MI is as follow:

$$MI(w_1, w_2) = \log \frac{P(w_1, w_2)}{P(w_1)P(w_2)} \quad (8)$$

$P(w_1)$ and $P(w_2)$ are the probability of word w_1 and w_2 to appear in the corpus separately. While $P(w_1, w_2)$ is the probability of word w_1 and w_2 to appear together.

The experimental corpus in this article stems from the People Daily of 1998. The extent is 123,882 lines (10,000,000 words), including 121,400 words and phrases.

3.2 The pretreatment of BP network model

The supervised WSD need artificial mark of meaning. But it is time consuming to mark artificially. So it is difficult to get the large scope and high quality training linguistic corpus. In order to overcome this difficulty and get large enough experimental linguistic corpuses, we should turn to seek the new way.

We use pseudoword in place of the real word. That can get the arbitrary large experimental corpus according to the real demand.

3.2.1 The construction of Pseudoword

Pseudoword is the artificial combination of several real words on the basis of experimental demand to form an unreal word that possesses many features of real words and instead of real word as the experimental object in natural language research.

In the real world, one word has many meanings derives from the variation and flexible application of words. That needs a long-term natural evolution. Frankly speaking, that evolution never ceases at all times. For example, the word –‘打’(da3) extends some new uses in recent years. Actually, in the endless history river of human beings, the development and variation of words meaning are rapid so far as to be more rapid than the replacement of dictionaries sometimes. Usually that makes an awkward position when you use dictionary to define the words meanings. Definitely, it is inconvenient for the research of natural linguistics based on dictionary.

But the meaning of pseudoword (Schütze, 1992) need not defined with the aid of dictionary and simulates the real ambiguous word to survey the effect of various algorithms of classified meanings.

To form a pseudoword need the single meaning word as a morpheme.

Set: $W_p = w_1 / w_2 / \ldots / w_i$

W_p is a pseudoword formed with w_i which contains i algorithms and meanings for every

algorithm of pseudoword is single meaning and every living example is about equal to a pseudoword marked meaning in corpus. That is similar to the effect of artificial marked meaning. But the effect is more stable and reliable than artificial marked meaning. What's more, the scope of corpus can enlarge endless according to the demand to avoid the phenomenon of sparse data.

To define the number of algorithm, we count the average number of meanings according to the large-sized Chinese dictionaries (Table 3.1). Table 3.2 show the overall number of ambiguous word and percentage of ambiguous word having 2~4 meanings in all ambiguous word. These two charts indicate that verb is most active in Chinese and its average number of meanings is most, about 2.56. The percentage of ambiguous word having 2~4 meanings is most in all ambiguous word.

part of speech	Average sense （including single-sense word）	Average sense （only ambiguous word）
noun	1.136452	2.361200
verb	1.220816	2.558158
adjective	1.144717	2.300774
adverb	1.059524	2.078431

Table 3.1 the average number of a Chinese word's sense

3.2.2 Define the input vector

It should be based on context to determine the sense of ambiguous word. The model's input should be the vector of the ambiguous word and context words. It is well-known that the number of context

ambiguous word	7955	/
Bi-senses word	5799	72.80%
Tri-senses word	1154	14.51%
Four-senses word	450	5.66%

Table 3.2 the distributing of ambiguous word

words showing on the both sides of ambiguous word is not fixed in different sentences. But the number of vectors needed by BP network is fixed. In other words, the number of neural nodes of input model is fixed in the training. If the extracting method of feature vector is (-M, +N) in context, in other words there are M vectors on the left of ambiguous word and N vectors on the right, the extraction of feature vectors must span the limit of sentences. If the number of feature vectors is not enough, the ambiguous words on the left and right boundaries of whole corpus do not participate in the training.

According to the extracting method of feature vector (-M, +N), the vector of model input is as following:

$$V_{输入} = \{MI_{11}, MI_{12}, \ldots, MI_{1i}, MI_{11}', MI_{12}', \ldots, MI_{1j}', MI_{21}, MI_{22}, \ldots, MI_{2i}, MI_{21}', MI_{22}', \ldots, MI_{2j}', MI_{31}, MI_{32}, \ldots, MI_{3i}, MI_{31}', MI_{32}', \ldots, MI_{3j}'\},\ 1 \le i \le M;\ 1 \le j \le N.$$

Where, MI_{1i} , MI_{1j}' are the MI of context and the first meaning of ambiguous word; MI_{2i} , MI_{2j}' are the MI of context and the second meaning of ambiguous word; MI_{3i} 、MI_{3j}' are the MI of context and the third meaning of ambiguous word. MI_{1i}, MI_{2i} and MI_{3i} are the feature words of ambiguous word on the left and MI of ambiguous word. MI_{1j}', MI_{2j}' and MI_{2j}' are the feature words of ambiguous word on the right and MI of ambiguous word.

pseudo-words	word ID	sample number	pseudo-words	word ID	sample number
W_1	34466	5550	W_4	84323	3773
	71345	3715		12751	2284
	31796	12098		52915	3900
	total	21363		total	9957
W_2	71072	9296	W_5	53333	1362
	78031	6024		29053	6135
	48469	1509		75941	1205
	total	16829		total	8702
W_3	7464	25925	W_6	39945	2346
	77375	2478		71335	1640
	23077	4704		51491	1012
	total	33107		total	4998

Table 3.3 the total number of the feature -vector sample of ambiguous word

Training corpus are 105,000 lines, and each line is a paragraph, totally about 10,000,000 words.

Table 3.3 shows the number of collected feature vector samples (the frequency of ambiguous word).

3.3 The definition of output model

Every ambiguous word has three meanings, totally eighteen meanings for six ambiguous words. Every ambiguous word trains a model and every model has three outputs showed by three-bit integer of binary system, such as the three meanings of ambiguous word W are showed as followed:

$s_{i1} = 100$ $s_{i2} = 010$ $s_{i3} = 001$

3.4 The definition of network structure

According to statistics, when (-M, +N) are (-8, +9) using the method of feature extraction, the cover percentage of effective information is more than 87% (Lu, 2001). However, if the sentence is very short, collecting the contextual feature words on the basis of (-8, +9) can include much useless information to the input model. Undoubtedly, that will increase more noisy effect and deduce the meaning-distinguish ability of verve network.

This article makes an on-the-spot investigation of experimental corpus, a fairly integrated meaning unit (the marks of border including comma, semicolon, ellipsis, period, question mark, exclamation mark, and the like), which average length is between 9~10 words. So this article collects the contextual feature words on the basis of (-5, +5) in the experiments, 10 feature words available that calculate MI with each meaning of ambiguous word separately to get 30 vectors. All punctuation marks should be filtered while the feature words are collected. The input layer of neural network model is regarded as 30 neural nodes. The triple-layer neural network adopts the inspired S function. From that, the number of neural nodes in hidden layer is defined as 12 on the basis of experimental contrast, and 3 neural nodes in output layer. Hence, the structure of model is 30 × 12 × 3, and the precision of differential training is defined as 0.3 based on the experimental contrast.

3.5 The test and training of model

The experimental corpus appeared in front are 123,882 lines. It is divided to three parts according to the demand of experiment, C1 (15,000 lines), C2 (60,000 lines), and C3 (105,000 lines). The open test corpus is 18,882 lines.

Table 3.3 tells us that there is a great disparity between the sample numbers of different ambiguous words in the experimental corpus of the same class. And the distribution of different meanings is not even for same ambiguous word. For the trained neural network has the good ability of differentiation for each word, the number of training sample should be about equal to each other for each meaning. So this experiment selects the least training samples. For example, there are 200 samples of the first meaning in training corpus, the second 400, and the third 500. To balance the input, each meaning merely has 200 samples to be elected for training.

Three groups of training corpus can train 3 neural networks possessing different vectors for every ambiguous word and make the unclose and open test for these networks separately.

4 The result of experiment

In order to analyze the effect that the extent of training corpus influences the meaning distinguish ability of neural network, this article trains the model of neural network using the experimental corpus individually, C_1, C2 and C_3, and makes the close and open test for 6 ambiguities separately.

The close test means the corpus are same in test and training.

The experiment is divided into two groups according to the extracting method of contextual feature words.

4.1 The first experiment one

Table 4.1 shows the result of the first experiment which extracts the contextual feature words using the method of （–5，＋5）.

In addition, the first experiment investigates that the extent of training corpus (the number of training samples big or small) influences the ability to distinguish the models. The result of test for 6

pseudo-words	close-test		open-test	
	accuracy	Training set	accuracy	Training set
W_1	0.8800	C_2	0.8951	C_3
W_2	0.8867	C_2	0.8775	C_2
W_3	0.8652	C_3	0.8574	C_3
W_4	0.8532	C_3	0.8687	C_3
W_5	0.8769	C_3	0.8745	C_3
W_6	0.8868	C_2	0.8951	C_3

Table 4.1 The contrast chat of experimental result for six ambiguities

ambiguities is showed in table 4.2 (close test), table 4.3 (open test), and table 4.4. Considering the length of this article, table 4.2 and table 4.3 shows the detailed data, and table 4.4 is brief.

pseudo-words		Training set		
		C_1	C_2	C_3
W_1	sense 1	0.9226	0.8169	0.8991
	sense 2	0.5513	0.8017	0.6872
	sense 3	0.8027	0.9564	0.9510
	average	**0.7589**	**0.8800**	**0.8720**
W_6	sense 1	0.8121	0.8780	0.9377
	sense 2	0.8389	0.8968	0.8804
	sense 3	0.7248	0.8856	0.8370
	average	**0.7919**	**0.8868**	**0.8850**

Table 4.2 The result of W_1 and W_6 in close test under the different training corpus

4.2 The second experiment

The second experiment investigates emphatically the effect that the method to collect the feature words influences the ability to distinguish BP model.

pseudo-words		Training set		
		C_1	C_2	C_3
W_1	sense 1	0.9019	0.7827	0.8942
	sense 2	0.4607	0.8097	0.7175
	sense 3	0.7792	0.9500	0.9515
	average	**0.7573**	**0.8798**	**0.8951**
W_6	sense 1	0.8233	0.9093	0.9535
	sense 2	0.8799	0.8182	0.8604
	sense 3	0.7278	0.8544	0.8038
	average	**0.8259**	**0.8683**	**0.8951**

Table 4.3 The result of W_1 and W_6 in open test under the different training corpus

There are many methods adopted in this experiment, including (–10，+10), (–3，+3), (–3，+7), (–7，+3), (–4，+6) and (–6，+4). Merely the ambiguous words W_1 and W_6 are regarded as the

pseudo-words		Training set		
		C_1	C_2	C_3
W_2	close	0.6628	0.8867	0.8772
W_3		0.6695	0.8453	0.8652
W_4		0.7414	0.8452	0.8532
W_5		0.8283	0.8537	0.8769
W_2	open	0.7287	0.8613	0.8700
W_3		0.8085	0.8384	0.8574
W_4		0.7920	0.8655	0.8687
W_5		0.8288	0.8775	0.8745

Table 4.4 The contrast chart of experimental result for four ambiguities

pseudo-words	feature collecting method	accuracy		Training set
		close-test	open-test	
W_1	(–10，+10)	0.8897	0.8685	C_1
	(–3，+3)	0.7917	0.7176	C_2
	(–4，+6)	0.8600	0.8888	C_3
	(–6，+4)	0.8797	0.8938	C_2
	(–3，+7)	0.8514	0.8827	C_3
	(–7，+3)	0.8431	0.8825	C_3
W_6	(–10，+10)	0.9031	0.8962	C_2
	(–3，+3)	0.8487	0.8460	C_2
	(–4，+6)	0.8982	0.8873	C_3
	(–6，+4)	0.8480	0.8772	C_2
	(–3，+7)	0.8669	0.8359	C_3
	(–7，+3)	0.8982	0.8895	C_3

Table 4.5 the experimental result under different feature collecting method

experimental objects in this group experiment. See table 4.5 for the correct percentage of WSD.

5. Analysis and discussion

See table 5.1 for the number of experimental corpus samples in experiment.

According to the table 3.3 and 5.1, the frequency of the each meaning (morpheme) of ambiguous word showing in corpus is quite different. That accords with the distribution of the every meanings of ambiguous word. However, there is one different point that the frequency of the each meaning of ambiguous word is rather high (that is the outcome selected by morpheme.). In other words, there are many examples showing for the each meaning of ambiguous word in training and test corpus. On the contrast, the difference of frequency is quite

pseudo-words	Morpheme ID	sample number	pseudo-words	Morpheme ID	sample number
W_1	34466	1040	W_4	84323	591
	71345	662		12751	484
	31796	2101		52915	829
W_2	71072	1296	W_5	53333	274
	78031	1043		29053	1153
	48469	315		75941	238
W_3	7464	4389	W_6	39945	430
	77375	469		71335	308
	23077	865		51491	158

Table 5.1 The number of experimental samples

obvious for the each meaning of real ambiguous word, because some meanings are used in oral language. But that never or seldom appears in experimental corpus.

The statistics can uncover this linguistic phenomenon. We find that the meaning of the most percentage of ambiguous word showing in the corpus is 83.54% on the whole percentage of each meaning. That illustrates the distribution of each meaning has a great disparity in real ambiguous word. Seeing that condition, to differentiate the meaning of ambiguous word is harder than that of real ambiguous word absolutely.

5.1 The analysis and discussion of the first experiment

Table 4.1 records the results of close and open tests in detail and the training materials to get these results.

Seeing from the experimental results, the correct percentage reaches 89.51% most (ambiguous word W1 and W_6) in open test of WSD, and 85.74% the least (ambiguous word W_3).

The relationship of correct percentage and the extent of training corpus can be deduced from the experimental results of table 4.2, 4.3 and 4.4.

The larger the extent of training corpus (the number of training sample), the larger the result of close test. It is obvious to see that from C1 to C2. From C2 to C3 one or two experimental results fluctuate more or less.

With the growing of training sample, the experimental results of open test increase steadily, except ambiguous word W2 (a little bit difference).

The experimental data prove the growing of training samples rise the correct percentage. However, when the rising reaches to a certain degree, more rising is not good for the improvement of model. What's more, the effect of noise is more and more remarkable. That decreases the model's ability of differentiation in a certain degree. On the other hand, after the growing of training corpus, the linguistic phenomenon around ambiguities is richer and richer, more and more complex. That makes it harder to determine the meaning.

5.2 The analysis and discussion of the second experiment

This article emphasizes on the collecting method of contextual feature words in experiment two, in other words, the effect that the different values of M and N influence the model of BP network. The experimental results (table 4.1 and 4.5) tell us that the context windows influence the correct percentage heavily. The correct percentage increases almost by leaps and bounds from (–3，+ 3) to（–5，+5）. The discrepancy is obvious despite close test or open test. The correct percentage increase again to(–10，+ 10), in which the close test of ambiguous word W6 is more than 90% and 89.62% the close test, with the exception of W1 which open test is slightly special. That illustrates the more widely the context windows open, the more the effective information is caught to benefit the WSD more.

Comparing the four feature methods of collection, including （–3，+ 7），(–7，+ 3），(–4，+ 6） and(–6，+ 4） with(–5，+ 5), the number of feature words besides the ambiguous word is various and the experimental results (table 4.1 and 4.5) are not same, although the windows are same. Among them, the correct percentage of （–5，+ 5）is the highest. And that of（–4，+ 6）and（–6，+ 4）is better than that of（–3，+ 7）and（–7，+ 3）a bit. That shows the more balanceable the feature words besides ambiguous word, the more advantageous to judge meaning, and the better the experimental results.

In addition, some experimental results of open test are better than that of close test. The main

reason is the experimental corpus of open test is smaller than training corpus. So the contextual meanings of ambiguous word in experimental corpus are rather explicit. Thereby, that explains why should be this kind of experimental result.

5.3 Conclusions

Considering the analysis of experimental data, the conclusions are as following:

First, the artificial model of neural network established in this article has good ability of differentiation for Chinese meaning.

Next, higher correct percentage of WSD stems from the large enough corpus.

At last, the larger the windows of contextual feature words, the more the effective information. At the same time, the more balanceable the number of feature words beside the ambiguous word, the more beneficial that for WSD.

6 Concluding remarks

Although the BP network is a classified model applied extensively, the report of research on WSD about it is seldom. Especially the report about the Chinese WSD is less, and only one report (Zhang, 2001) is available in internal reports.

Zhang (2001) uses 96 semantic classes to instead the all words in training corpus according to the *TongyiciCilin*. The input model is the codes of semantic class of contextual words and ambiguities. The experiment of WSD merely makes for one phrase '材料'(cai2liao4) in this document and the correct percentage of open test is 80.4%. '材料' has 3 meanings and that is similar to the ambiguities structured in my article.

Using BP for Chinese WSD, the key point and difficulty are on the determination of input model. The performance of input model may influence the construction of BP network and the output result directly.

We make the experiment on the input of BP network many times and finally find the input model introduced as above (table 3.1) which test result is satisfied.

Acknowledgements This work was supported by the National Natural Science Foundation of China (Grant No. 60203020).

Reference

Lu Song, Bai Shuo, et al. 2001. Supervised word sense disambiguation bassed on Vector Space Model, *Journal of Comouter Research & Development*, 38(6): 662-667.

Pedersen. 2001. *Lexical semantic ambiguous word resolution with bigram-based decision trees*, In Proceedings of the Second International Conference on Intelligent Text Processing and Computational Linguistics, pages 157-168, Mexico City, February.

Escudero, G., Marquez,L., et al, 2000. *Naive Bayes and examplar based approaches to word sense disambiguation revisited*. In Proceedings of the 14th Europear Conference on Artificial Intelligence, ECAI.

Adam,L.B. 1996. A maximum entropy approach to natural language proceeding. *Computational Linguistics*, 22(1):39-71.

Li, J. Z. 1999. An improved maximum language and its application. *Journal of software*, 3:257-263.

Yarowsky, D. *Word sense disambiguation using statistical models of Roget's categories trained on large corpora*. In: Zampolli, A., ed. Computation Linguistic'92. Nantas: Association for Computational Linguistics, 1992. 454~460.

Hinrich Schütze, 1998. Automatic word sense discrimination. *Computational Linguistics*, 24(1): 97-124.

Lu Song., Bai Shuo. 2002. An unsupervised approach to word sense disambiguation based on sense-word in vector space model. *Journal of Software*. 13(06):1082-08

Hinrich Schütze. 1992. *Context space*. In AAAI Fall Symposium on Probabilistic Approaches to Natural Language, pages 113–120, Cambridge, MA.

Lu Song Bai Shuo. 2001. *Quantitative Analysis of Context Field.* In Natural Language Processing, CHINESEJ.COMPUTERS, 24(7) , 742-747

Zhang Guoqing, Zhang Yongkui. 2001. A Neural-network Based Word Sense Disambiguation Method. *Computer Engineering*, 27(12).