Language, Information and Computation(PACLIC 11), 1996, 119-128

Neural Networks in Chinese Lexical Classification

Md Maruf Hasan *Kim-Teng Lua*
Department of Information Systems & Computer Science
National University of Singapore
{maruf, luakt}@iscs.nus.sg

Abstract

Lexical attributes, like syntactic (part-of-speech) and semantic (semantic category) attributes, are in most cases, ambiguous in every languages. Automatic resolution of ambiguity of these attributes can be achieved using different techniques; rule-based, statistical, NN-based and their hybrids. Moreover, one linguistic feature also has influence over the resolution of ambiguity of another feature; eg.. knowledge of syntactical category can assist smooth disambiguation of semantic category and vice versa. Properly disambiguated syntactic and semantic properties of lexicon may significantly help us in word sense disambiguation, text analysis, information retreival, natural language understanding and speech processing etc. In this paper, we have presented our neural network based Classification Tool. We have used this tool in Part-of-Speech tagging and Semantic-Category tagging of Chinese lexicon with the help of thesaurus and large training corpus. Experimental results are analysed and compared.

1 Introduction to Lexical ambiguity in Chinese

Lexical attributes, such as part-of-speech and semantic category disambiguation in Chinese is more or less similar to that of western languages. We explain an example of such ambiguity of the Chinese lexicon, 不准 (*BuZhun*). In sentence, 这里*不准*吸烟 [ZheLi *BuZhun* Xi Yan / Smoking is *prohibited* here], the part-of-speech and semantic category of the word, 不准 is *verb* and *prohibition* respectively. But in another sentence, 我的手表*不准* [Wo De ShouBiao *BuZhun* / My watch is (showing) i*naccurate* (time)], the same word, 不准 belongs to the category of *adjective* and *similarities and differences* respectively.

Disambiguation of part-of-speech is well addressed by Greene et. al. (1973), Brill (1994) (rule-based: English); Chang et. al. (1993) (statistical: Chinese); Church (1988), Kempe (1993) (statistical: English, German); Nakamura et. al. (1990), Schmid (1994) (connectionist: English, German). To our knowledge, very few people has dealt with the connectionist approach for the Chinese lexicon.

Word sense disambiguation is well addressed by Lam (1995) (dictionary-clue word: Chinese); Lesk et. al. (1986) (dictionary-clue word: English); DeRose (1988), Yarowsky (1992) (statistical: English); Ide et. al.

(1990), Cottrell (1989) (connectionist: English). Again, to our knowledge, connectionist approaches are less explored for Chinese lexical research.

In disambiguation of Chinese lexicon, also there are approaches to combine word-selection (segmentation) and part-of-speech tagging together (Chang et. al. 1993, Bai 1995). In English NLP, Richard and Bronwyn (1995) used semantic categorical attributes to assist part-of-speech tagging too. Yarowsky (1992) has made use of part-of-speech tags information to assign semantic category tags in his statistical model of word sense disambiguation. He has calculated *concordances* of 100 surrounding words (50 to the left and 50 to the right) to decide the right category tag to the current word.

Nakamura et. al. (1990), in their speech recognition system, trained a 4-layered feed-forward network with up to three preceding words' tags as input to predict the word category (POS) of the current word. Using this predictor, they were able to enhance their Speech Recognition System's performance from 81.0% to 86.9%. Schmid (1994) claimed 96.22% POS tagging accuracy using his 2-layered Net-Tagger. This accuracy is comparable to that of Trigram-based tagger and better than that of HMM tagger.

In this paper, we have presented a multilayer feedforward network with backpropagation training algorithm to tag part-of-speech and semantic-category of the Chinese lexicon with the help of large tagged corpora. The accuracy achieved is comparable to the accuracy of similar other systems.

2 Artificial Neural Networks in classification and recognition.

Minsky and Pappert's results dampened the enthusiasm in neural network based research for about 20 years (Minsky 1969). Renewed interest in ANN began since 1980 due to Hopfield's *energy approach* and Werbos' *backpropagation learning algorithm* for multilayer perceptron (Werbos 1974). Multilayer perceptron (a.k.a. multilayer feedforward neural network) was widely popularized after Rumelhart's work (Rumelheart 1986). We found MLFF is the best generalized tool in lexical classification task.

2.1 Our NN-based system.

Due to its strong capability in classification and recognition, we have used multilayer feedforward network with backpropagation learning algorithm (cf. Figure 1) in our classification system. The system is implemented in such a way that it can be used as a *generalized* tool for classification (or disambiguation) of any lexical attributes: part-of-speech, semantic-category and so on. This system can be used to build MLFF networks of different scale and architecture, e.g. with or without hidden layers, various number of nodes in each layers, different context window size etc. We have used this system with large corpora and examined its usefulness in part-of-speech tagging and semantic categorization. In the rest of this paper, we will describe the detail of our neural network system, our finding in training and tagging of part-of-speech and semantic-category using this system.

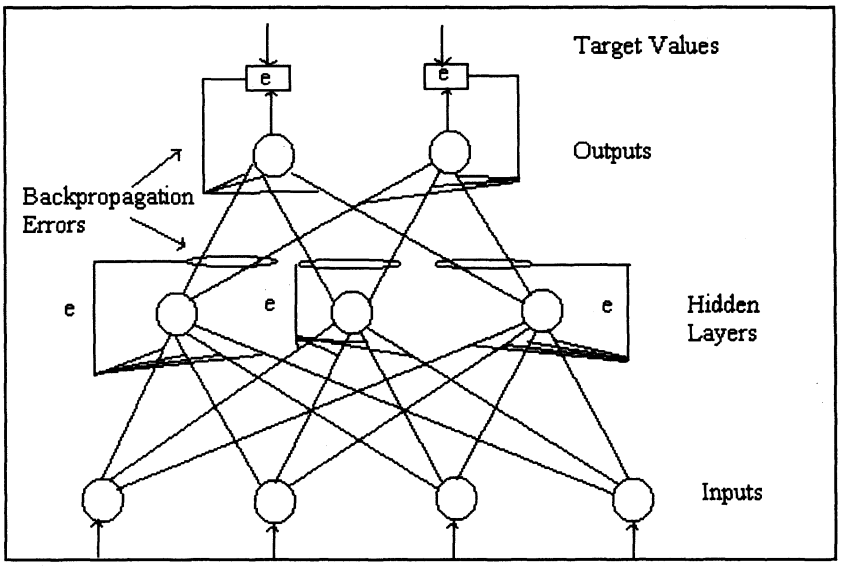

Figure 1: MLFF Network with Backpropagation

2.2 MLFF Neural Networks with Backpropagation Learning

Multilayer Feedforward Networks prove to be very useful for arbitrary mapping. Our task of classifying lexical attributes can be interpreted as such a mapping of **g': x→z**, where **x** and **z** are random vectors in $\mathbf{R}^n$ and $\mathbf{R}^m$ respectively with joint probability of $\psi(\mathbf{x},\mathbf{z})$. We made a training set (of P pairs of patterns, ie. hand-tagged corpus) available to the network with a form of $\{\mathbf{x}^p, \mathbf{t}^p \mid p=1,2, ..., P\}$, where the input patterns $\mathbf{x}^p$ have some unknown probability distribution $\rho(\mathbf{x})$, and the $\mathbf{t}^p$ are the desired or known target vector values for the corresponding input vectors $\mathbf{x}^p$. Given input $\mathbf{x}^p$ the computed output from the network is $\mathbf{z}^p$, where $\mathbf{z} = \mathbf{g}(\mathbf{x})$

We have implemented a **MLFF** network with *Backpropagation* learning algorithm in our system. Figure 1 illustrates such kind of a network. A very elaborate description of MLFF can be found in Patterson (1996), pp141-243.

During the processing in a MLFF network, activations are propagated from input units through hidden units to output units. At each unit **j**, the weighted input activations $\mathbf{a_i w_{ij}}$ are summed and a bias parameter θ_j is added.

$$net_j = \sum_i a_i w_{ij} + \theta_j$$

The resulting network input, $\mathbf{net_j}$ is then passed through a sigmoid function to obtain resulting activation $\mathbf{a_j}$ within interval **[0,1]**.

$$a_j = \frac{1}{1+e^{-net_j}}$$

The network learns by adapting the weights of the connections between units until the correct output is produced. We have used a backpropagation algorithm which performs a gradient descendent search on the error surface. The weight update Δw_{ij} is defined as follows:

$$\Delta w_{ij} = \eta a_{pi} \delta_{pj}$$

$$\delta_{pj} = \begin{cases} a_{pj}(1 - a_{pj})(t_{pj} - a_{pj}) \\ (j \text{ is an output unit}) \\ \\ a_{pj}(1 - a_{pj}) \sum_{k} \delta_{pk} w_{jk} \\ (j \text{ is an hidden unit}) \end{cases}$$

To improve the rate of convergence, we have added a momentum term in our gradient expression. Our updated formula with momentum term (*underlined part*), (Rumelhart 1984).

$$\Delta w_{ij}(t+1) = \eta a_{pi} \delta_{pj} + \underline{\alpha \Delta w_{ij}(t)}$$

2.2.1 Recursive Training Algorithm

The output layer of the network consists of one neuron for each tag (part-of-speech or semantic-category). The input layer consists of one neuron for each category *times* context window size. A relatively small context window size is quite enough. Usually two or three preceding words and one or two succeeding words is sufficient. The number of hidden layers and the number of nodes in a hidden layer are a critical issue. Using our system we could easily implement networks with different hidden-layer architecture and come to a decission after taking into consideration of their performance fluctuation.

The activation of an input unit represents the probability of the corresponding category for the respective word in the context. For the currently classified word and the following words, this probability is given by the lexical probability (MLE) of them. For the preceding words, there is more information available, since these words have already been classified. Therefore, the output activations of the network at the time of their classification are used instead of the lexical probabilities. *Recurrence* (Schmid 1994), in the network results from this and requires a special training procedure. Detail training procedure is stated in Table 1 below.

Rather than adjusting the weight after each pattern representation, we have accumulated the errors, $\mathbf{E}^p$ over the whole training set $\{\mathbf{x}^p, \mathbf{t}^p \mid p= 1, 2, \ldots P\}$ and then make the adjustments. This method is known as *off-line* training or *batch processing* training and is able to speed up training.

Phase 1: The *target values* of the output units (which the network must learn) are fed back to the input units instead of the activation values produced by the network. This deals with the fact, that the output activations are quite irregular at the beginning.
Phase 2: The *weighted average* of the target output and the actual output is fed back.
Phase 3: The *actual output* is directly fed back to the network.

Table 1: Recursive 3-phase training procedure

3 Experimental Results

In this section, we will expain our experimental result on part-of-speech tagging and semantic categorization using the above-mentioned neural networks. Direct comparison of performance between researchers is difficult, compounded by variance in corpora, tagset and grading criteria. Therefore, we are not strictly claiming and comparing our result here. Rather, we are willing to make our system available to others for evaluation and further improvement.

3.1 Part-of-Speech Tagging

We have trained our neural network with a tagged corpus of about two million words from *Sinica 1.0 Balanced Corpus* (CKIP 1995). Sinica 1.0 is so far the largest widely available tagged Chinese corpus of its size with part-of-speech tags. A detail about its architecture, tagging principle, tagset etc. can be found in Chinese Knowledge Information Processing Group's technical report and other publication from Academia Sinica, CKIP group.

3.1.1 Tagset and Corpora

Sinica 1.0 is a collection of over two million words and is tagged with a tagset of 46 part-of speech tags. It is balanced (Hsu and Huang, 1995) as its constituent entries are gathered from various sources ranging from reportage to everyday conversation with insightful selection criteria.

We have performed both *inside* and *outside* tests. We have used two different test corpora for outside test. One is a reserved portion (not included in training) of Sinica corpus. Another test corpus is taken from Tsinghua corpus (Bai 1995). The latter test corpus has shown degraded performance.

3.1.2 Experimental Result of POS tagging

We have used two different version of MLFF to train our system. The first model consists of only input and output layers with no hidden nodes. The context window size is 3 preceding and 2 succeeding words. In the second model, we have used one hidden layer with 6 hidden nodes; context size used is 2 preceding and 2 succeeding words. We have organized our experimental results in the following table (Table 2). Better tagging result is achieved due to the fact that our system works with back propagation error feedback and the recursive training algorithm (stated above). More than one possible output tag (along with its output activation) can be printed for each word. Erroneous results usually have lower activation. This made manual post-processing easier.

Training Corpus Size = 2,030,000 words	
Test Corpus Size = 4,987 words (*Outside Test: Sinica*) and 2,400 words (*Inside Test: Sinica*)	
Tagset = 46 Classes (CKIP tagset)	
% Accuracy: Two Layered MLFF (*3 preceding & 2 succeeding words*)	**% Accuracy: Three Layered MLFF** (*2 preceding & 2 succeeding words*)
Outside Test: 95.81% (4,778 words)	96.21% (4,798 words)
Inside Test: 97.33% (2,336 words)	98.79% (2,371 words)
Tagging Speed = 310 words/sec (SUN SPARC 20)	

Table 2: Part-of-Speech tagging result

Similar outside test is performed with the other set of test corpus taken from Tsinghua corpus (Bai, 1995). This time the tagging accuracy is decreased by about 2.4%. While analyzing errors, we have found that some of the errors are due to different linguistic usage in China and Taiwan. Different proper nouns are used with different frequency in China and Taiwan based text. Scientific and other special terminologies also differ very often.

For all the cases above, a few of the tagging errors are due to system's ignorance of appropriate semantic attribute of the words. Unregistered words introduced a significant portion of erroneous output as they are tagged quite randomly. Unlike semantic-category tagging (section 3.2.2), we did not implement any mechanism to handle unknown words during part-of-speech tagging.

3.2 Semantic Category Tagging

We have trained our MLFF network with a training corpus of 348,000 words. We have also accomodated a probablistic algorithm to assign tags to

unregistered words (see section 3.2.2 below) and analysed the performance of semantic categorization.

3.2.1 Tagset and Corpora

TongYiCi CiLin has gathered about 52,206 Chinese lexicons and Classified them in a 3-level hierarchy. Level 1 consists of 12 classes; *Level 2 consists of 94 classes* and Level 3 consists of 1428 classes. More than 15% of the total entries have more than one semantic tag.

In our experiment, we have examined results up to the second level; ie. we have dealt with only 94 semantic categories. Compared to Roget's thesaurus, in the second level of TongYiCi CiLin's classification hierarchy, we can get sufficient (may be, not abundant) semantic information for NLP application. (Our program can similarly be used for tagging at the third level categories, provided that a larger training corpus is available to train it.)

Due to the unavailability of tagged training corpus, we have semi-automatically built our training corpus. Detail of this unsupervised method can be found in Lua (1996). Unlike the Sinica Corpus (used in POS-tagging above), this semi-automatically generated semantic corpus has lesser level of accuracy. (Please take note that the accuracy of this training corpus is 91.05%). This is one of the few reasons, we have achieved a comparatively lower performance in semantic classification.

3.2.2 Unknown word tagging

Unknown words are tagged with the help of a suffix tree. Suffix lexicons are automatically built from the training corpus. We have generated suffix tree of length 3 through statistical analysis of the training corpus. Each node of the tree, except the root, is labelled with one character. Tag probability vectors are also attached at the leaves. Each node of the suffix tree was annotated with an *open-class-tag* set. *Open-class-tags* are those, which allow production of new word, (eg. social & cultural activities, 文体活动; politics, 政治 etc.).

Raw suffix tree needs to be validated and pruned as there are some entries which are not meaningful. Validation and pruning of the suffix tree is made with the help of the following information measure, I(Suff).

$$I(Suff) = -\sum_{s} P(s|Suff)\log_2(s|Suff)$$

P(s|Suff) is the probability of semantic tag, s, given a word with suffix *Suff*. A threshold [*Threshold* = (*Frequency* of the suffix in the corpus *times Information* measure of it)] of 20 is used to prune less frequent suffixes. Once pruning of a leaf is done, the probability vector of the leaf node is also accumulated to its parents node (ie. parent node becomes leaf).

For an unknown word, the suffix tree is traversed from the root. If a match is found there (ie. if it reaches to a leaf node), the corresponding tag probability is extracted and given as activation for this word. Otherwise, a

default value is returned as activation. Being an idegraphic language, Chinese word tagging has benefitted from this approach. Words like 同学们, 电脑学会 etc has correctly tagged through this approach. Note, 们 is a plural suffix for animate in Chinese. 学会 (society) is another suffix used widely to name academic societies like Computer Science Society, Languistic Society etc. According to the thesaurus categorization all these words should be categorized in a group. Alphabetic languages are less likely to be benefitted from this kind of approach.

3.2.3 Experimental result of Semantic tagging

We have used 2 layered (no hidden layer) and 3-layered (with a single hidden layer of 47 nodes) MLFF to train two different sets of parameters. For the two-layered version, we have used 3 preceding and 2 succeeding word as context which created a sum of 53,016 network parameters (ie. 94*6*94). With the 3 layered version we have considered 2 preceeding and 1 succeeding word(s). Total number of network parameters is 1,661,168. (For this 3-layered network, the training corpus size is inadequate enough to fail to optimize the huge number of network parameters and therefore, performed poorly)

We used a test corpus of 7,251 words not included in the training corpus and tagged with the two sets of network parameters. For the 2-layered and 3-layered model, we have achieved 88.18% (6394 entries are correctly

Training Corpus Size = 348,000 words	
Test Corpus Size = 7,251 words (Outside Test)	
Tagset = 94 Classes (second level hierarchy of TongYiCi CiLin)	
% Accuracy: Two Layered MLFF (*3 preceding & 2 succeeding words*)	**% Accuracy: Three Layered MLFF** (*2 preceding & 1 succeeding words*)
88.18% (6394 words)	83.99% (6090 words)
Tagging Speed = 330 words/sec (SUN SPARC 20)	

Table 3: Semantic tagging result

categorized) and 83.99% (6090 entries) accuracy through manual checking.

Adding hidden layers of 47 nodes increased number of parameters significantly. As we could not provide a larger training corpus, the performance has degraded seriously. 2-layered model has parameters less than that of a tri-gram system. We have achieved a tagging speed of 330 words/sec on a Sparc20 workstation. We admit the drawback of longer training time. But as training need to be done only once it would be acceptable.

Recalculating the test result at the first level (12 categories) of TongYiCi CiLin's categorization, we have found that the accuracy for 2-layered model 90.10% (correct category for 6533 entries) and the accuracy for 3-layer

model is 84.2% (correct for 6105 entries). This implies that the parameters of 2-layered networks are more optimized and stable than that of the 3-layer model.

Our training corpus is semi-automatically created with an accuracy level of 91.05%. Thus an accuracy of 87.3% is quite acceptable. The output is generated with extra information like the *activation value* and *category-of-judgement tags* for each word. Category-of-judgement tags include the following three types of information: Registered words, ie. found in the dictionary, **f**; Not found in the corpus as a fullform but matches with suffix tree, **s**; Tagged randomly, **r**.

We have found that most of the erroneous tags either have very low output activation (less than 0.40) or tagged with the default activation or suffix anlysis. Some errors are also due to the systems ignorance of the proper part-of-speech tag of the relevant word (ie. noun-verb or noun-adjective ambiguity). Including part-of-speech information in the input vector would increase the accuracy level. A rule-based or manual post-processing task would be easier as the output activation and category-of-judgement information can be used as indication of probable errors.

Inside-Test outperforms O*ut-side Test*. Again 3-layered MLFF performs better than 2-layered one. It can be supported with the fact that memorization (rather than generalization) occured with the help of huge parameters of the 3-layered network.

4 Conclusion and Further Research

By observing the tagging result, we have found that most of the errors are due to the system's ignorance of the relevant part-of-speech knowledge (while tagging semantic category) and semantic knowledge (while tagging part-of-speech). Another common reason in erroneous tagging is unregistered words. Although we could tackle the unknown word problem with the help of the probablistic algorithm (section 3.2.2), we could not handle unknown word problem while tagging part-of speech.

We have only fed the maximum likelihood probability and contexual information to the network. Part-of-speech tag probability posses significant information for semantic tag disambiguation and vice versa. Input vectors can be redesigned to accomodate these extra knowledge. Reconstructing the algorithm to accomodate extra features would give better results. Building a complete dictionary is also essential. A hybrid network architecture with concurrent part-of-speech and semantic tagging parts interconnected with a well-defined feedbacking algorithm can also be tried.

Acknowledgements

We would like to thank the CKIP group of Academia Sinica to put their initiative to build Sinica 1.0 Corpus (the first large tagged Chinese corpus ever available to the researchers). We are particularly indebted to Helmut Schmid,

IMS, University of Stuttgart, Germany, for various support at different stage of this research. We are also grateful to Bai Shan Hu, ISS, National University of Singapore for providing Chinese text Corpus.

References

Bai Shan-hu, *An Integrated Model of Chinese Word Segmentation and Part of Speech Tagging.* Advances and Applications on Computational Linguistics. Tsing Hua University Press, Beijing. pp56-61. In Chinese, 1995.

Brill, Eric. *Some Advances in Transformation-Based Part of SpeechTagging.* In Proceedings of **AAAI'94**. Also available in electronic form: http://xxx.lanl.gov/ps/cmp-lg/9406010

Chang Chao-Huang and Chen Cheng-Der, *A study on Integrating Chinese Word Segmentation and Part-of-Speech Tagging,* Communications of COLIPS, Vol. 3, No. 1, pp69-77, 1993.

Church, K. *Stochastic parts program and noun phrase parser for unrestricted text.* Proceedings of Second Conference on Applied Natural Language Processing, Austin, TX. 1988.

CKIP, *The Tagging Principle of CKIP Tagged-Corpus*, Technical Report #95-02, Institute of System Science, Academia Sinica, Taipei, 1995.

Cottrell, Garrison W. *A Connectionist Approach to Word Sense Disambiguation,* Pitman: Morgan, 1989.

DeRose, Steven J. *Grammatical Category Disambiguation by Statistical Optimization,* Computational Linguistics. Vol. 14, No. 1, pp31-39, 1988.

Greene, Barbara B. and Rubin, Gerald M. *Automated Grammatical tagging of English.* Department of Linguistics, Brown University, 1971.

Hsu , Hui-Li and Huang, Chu-Ren, Design Criteria for a Balanced Modern Chinese Corpus. In Proceedings of ICCPOL, Honolulu, Hawaii, 1995

Ide, N. M., Veronis, J., *Very large neural networks for word sense disambiguation.* **ECAI-90**, Proceedings of the European Conference on Artificial Intelligence, Stolckholm 1990.

Kempe, A., *A stochastic Tagger and an Analysis of Tagging Errors.* Internal Paper. Institute for Computational Linguistics, University of Stuttgart, Germany. 1993.

Lam Sze-Sing, *A New Approach for extracting Inter-Word Semantic Relationship from a Contemporary Chinese Thesaurus Sense Disambiguation Using On-line Dictionaries.* M.Phil Thesis. Dept of SE & EM, The Chinese University of Hong Kong, 1995.

Lesk, M., *Automated word sense disambiguation using machine-readable dictionaries: How to tell a pine cone from an ice cream cone.* Proceedings of the 1986 SIGDOC Conference, pp 24-26, 1986.

Lua, K-T, *An Efficient Inductive Unsupervised Semantic Tagger*, submitted to Computer Processing of Oriental Languages, May, 1996. Can be extracted from "http://xxx.lanl.gov/ps/cmp-lg/9606012/". 1996.

Mei Jia-Ju,, Zhu Yi-Ming, Gao Yun-Qi and Yin Hong-Xiang, *TongYiCi CiLin*, ShangHai DianShu ChuBanShe, 1983.

Minsky, M. and Pappert, S. *Perceptrons: An introduction to Computational Geometry*, MIT Press. Cambridge, Mass., 1969

Nakamura, M.; K. Maruyama; T. Kawabata and K. Shikano. *Neural network approach to word category prediction in English texts.* Proceedings of **COLING-90**, Helsinki University, pp 213-218, 1990.

Patterson, D. W., *Artificial Neural Networks: theory and Application*, Prentice Hall, pp141-243, 1996.

Richard F. E. & Bronwyn E. A. Slater, *Disambiguation by association as a practical method: Experiment and findings.* Journal of Quantititive Linguistics, Vol. 2, No. 1, pp 43-52. Sweets & Zeitlinger, 1995.

Rumelhart, D. E. and McClelland J. L., *Parallel Distributed processing*, MIT-Press, Cambridge, MA,. 1984.

Rumelhart, D. E. and McClelland J. L., *Parallel Distributed processing: Exploration in the Microstructure of Cognition*, MIT-Press, Cambridge, Mass., 1986.

Schmid, H. *Part-of-Speech Tagging with Neural Networks*, Proceeding of COLING-94, pp172-176, 1994. Also electronically available from *URL: http://xxx.lanl.gov/ps/cmp-lg/9410018/*

Werbos, P. *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*, PhD Thesis, Dept. of Applied Mathematics, Harvard University, Cambridge, Mass., 1974.

Yarowski, David. *Word Sense Disambiguation using statistical models of roget's categories trained on large corpora.* Proceedings of COLING-92.