Markov random field based English Part-Of-Speech tagging system

Sung-Young Jung , Young C. Park, Key-Sun Choi and Youngwhan Kim*

Computer Science Department
Korea Advanced Institute of Science and Technology
Taejon, Korea
*Multimedia Research Laboratories
Korea Telecom
{chopin,ycpark,kschoi}@csone.kaist.ac.kr

Abstract

Probabilistic models have been widely used for natural language processing. Part-of-speech tagging, which assigns the most likely tag to each word in a given sentence, is one of the problems which can be solved by statistical approach. Many researchers have tried to solve the problem by hidden Markov model (HMM), which is well known as one of the statistical models. But it has many difficulties: integrating heterogeneous information, coping with data sparseness problem, and adapting to new environments. In this paper, we propose a Markov radom field (MRF) model based approach to the tagging problem. The MRF provides the base frame to combine various statistical information with maximum entropy (ME) method. As Gibbs distribution can be used to describe a posteriori probability of tagging, we use it in maximum a posteriori (MAP) estimation of optimizing process. Besides, several tagging models are developed to show the effect of adding information. Experimental results show that the performance of the tagger gets improved as we add more statistical information, and that MRF-based tagging model is better than HMM based tagging model in data sparseness problem.

1 Introduction

Part-of-speech tagging is to assign the correct tag to each word in the context of the sentence. There are three main approaches in tagging problem: rule-based approach (Klein and Simmons 1963; Brodda 1982; Paulussen and Martin 1992; Brill et al. 1990), statistical approach (Church 1988; Merialdo 1994; Foster 1991; Weischedel et al. 1993; Kupiec 1992) and connectionist approach (Benello et al. 1989; Nakamura et al. 1989). In these approaches, statistical approach has the following advantages :

- a theoretical framework is provided
- automatic learning facility is provided
- the probabilities provide a straightforward way to disambiguate

Many information sources must be combined to solve tagging problem with statistical approach. It is a significant assumption that the correct tag can generally be chosen from the local context. Not only local sequences of words and tags are needed to solve tagging problem, but syntax, semantic and morphological level information is also required in general. Usually information sources such as bigram, trigram and unigram are used in the tagging systems which are based on statistical method. Traditionally, linear interpolation and its variants have been used to combine the information sources, but these are shown to be seriously deficient.

ME (Maximum Entropy) estimation method provides the facility to combine several information sources. Each information source gives rise to a set of constraints, to be imposed on the combined estimate. The function with the highest entropy within the constraints is the ME solution. Given consistent statistical evidence, a unique ME solution is guaranteed to exist and an iterative algorithm is provided.

MRF (Markov random field) model is based on ME method and it has the facility to combine many information sources through feature functions. MRF model has the following advantages: robustness, adaptability, parallelism and the facility of combining information sources. MRF-based tagging model inherits these advantages.

In this paper, we will present one of the statistical models, namely MRF-based tagging system. We will show that several information sources including unigram, bigram and trigram, can be combined in MRF-based tagging model. Experimental results show that the MRF-based tagger has very good performance especially when training data size is small.

Section 2 describes the tagging problem , Section 3 describes statistical model already known

and section 4 the research for combining statistical information. Section 5 provides MRF-based tagging model and section 9 showes the experimental results. Section 10 compares MRF with HMM. Finally we conclude in section 11.

2 The Problem of Tagging

When sentence $W = w_1, w_2, ..., w_n$ is given, there exist corresponding tags $T = t_1, t_2, ..., t_n$ of the same length. We call the pair (W,T) an alignment. We say that word w_i has been assigned the tag t_i in this alignment. We suppose that a set of tags is given. Tagging is assigning correct tag sequence $T = t_1, t_2, ..., t_n$ for given word sequence $W = w_1, w_2, ..., w_n$.

3 Probabilistic Formulation(HMM)

Let us assume that we want to know the most likely tag sequence $\phi(W)$, given a particular word sequence W. The tagging problem is defined as finding the most likely tag sequence T

$$\phi(W) = arg \max_T P(T|W) \tag{1}$$

$$= arg \max_T \frac{P(W|T)P(T)}{P(W)} \tag{2}$$

$$= arg \max_T P(W|T)P(T) \tag{3}$$

where $P(T)$ is the a priori probability of a tag sequence T, $P(W|T)$ is the conditional probability of word sequence W, given the sequence of tags T, and $P(W)$ is the unconditioned probability of word sequence W. The probability $P(W)$ in (2) is removed because it has no effect on $\phi(W)$. Consequently, it is sufficient to find the tag sequence T which satisfies (3).

We can rewrite the probability of each sequence as a product of the conditional probabilities of each word or tag given all of the previous tags.

$P(W|T)P(T)$

$$= \Pi_{i=1}^n \left\{ \begin{array}{l} P(w_i|t_i, ..., t_1, w_{i-1}, ..., w_1) \\ \times P(t_i|t_{i-1}, ..., t_1) \end{array} \right\}$$

Typically, one makes two simplifying assumptions to cut down on the number of probabilities to be estimated. First, rather than assuming w_i depends on all previous words and all previous tags, one assumes w_i depends only on t_i. Second, rather than assuming the tag t_i depends on the full sequence of previous tags, we can assume that local context is sufficient. This locality assumed is refered to as a Markov independence assumption.

Using these assumption, we approximate the equation to the following

$$P(W|T) \simeq \Pi_{i=1}^n P(w_i|t_i) \tag{4}$$

$$P(T) \simeq \Pi_{i=1}^n P(t_i|t_{i-1}) \tag{5}$$

Accordingly, $\phi(W)$ is derived by applying (4) and (5) to (3).

$$\phi(W) = arg \max_T \Pi_{i=1}^n P(w_i|t_i)P(t_i|t_{i-1}) \tag{6}$$

We can get each probability value from the tagged corpus which is prepared for training by using (7) and (8).

$$P(w_i|t_i) = \frac{C(w_i, t_i)}{C(t_i)} \tag{7}$$

$$P(t_i|t_{i-1}) = \frac{C(t_{i-1}, t_i)}{C(t_i)} \tag{8}$$

where $C(t_i), C(t_i, t_j)$ is the frequency obtained from training data.

Viterbi algorithm (Forney73) is the one generally used to find the tag sequence which safisfies (6) and this algorithm guarantees the optimal solution to the problem.

This model has several problems. First, some words or tag sequences may not occur in training data or may occur with very low frequency; nevertheless, the words or tag sequences can appear in tagging process. In this case, it usually causes very bad result to compute (6), because the probability has zero value or very low value. This problem is called *data sparseness* problem. To avoid this problem, smoothing of information must be used. Smoothing process is almost essential in HMM because HMM has severe data sparseness problem.

4 combining information sources

4.1 linear interpolation

Various kinds of information sources and different knowledge sources must be combined to solve the tagging problem. The general method used in HMM is linear interpolation, which is the weighted summation of all probability information sources.

$$P_{combined}(w|h) = \sum_{i=1}^k \lambda_i P_i(w|h) \tag{9}$$

where $0 < \lambda_i \leq 1$ and $\sum_i \lambda_i = 1$.

This method can be used both as a way of combining knowledge sources and smoothing information sources.

HMM based tagging model uses unigram, bigram and trigram information. These information sources are linearly combined by weighted summation.

$$\tilde{P}(t_i|t_{i-1}, t_{i-2}) = \lambda_1 P(t_i|t_{i-1}, t_{i-2}) + \lambda_2 P(t_i|t_{i-1})$$
(10)

where $\lambda_1 + \lambda_2 = 1$. The parameter λ_1 and λ_2 can be estimated by forward-backward algorithm (Deroua86+) (Charniak93+) (HUANG90+).

Linear interpolation is so advantageous because it reconciles the different information sources in a straightforward and simple-minded way. But such simpliticy is also the source of its weaknesses:

- Linearly interpolated information is generally inconsistent with their information sources because information sources are heterogeneous for each other in general.

- Linear interpolation does not make optimal combination of information sources.

- Linear interpolation has over-estimation problem because it adjusts the model on the training data only and has no policy for untrained data. This problem occur seriously when the size of the training data is not large enough.

4.2 ME(maximum entropy) principle

There is very powerful estimation method which combines information sources objectively. ME(maximum entropy) principle (Jaynes57) provides the method to combine information sources consistently and the ability to overcome over-estimation problem by maximizing entropy of the domain with which the training data do not provide information.

Let us describe ME principle briefly. For given x, the quantity x is capable of assuming the discrete values $x_i, (i = 1, 2, ..., n)$. We are not given the corresponding probabilities p_i; all we know is the expectation value of the function $f_r(x), (r = 1, 2, ..., m)$:

$$E[f_r(x)] = \sum_{i=1}^{n} p_i(x_i) f_r(x_i)$$
(11)

On the basis of this information, how can we determine the probability value of the function $p_i(x)$? At first glance, the problem seems insoluble because the given information is insufficient to determine the probabilities $p_i(x)$.

We call the function $f_r(x_i)$ a *constraint function* or *feature*. Given consistent constraints, a unique ME soluton is guaranteed to exist and to be of the form:

$$p_i(x_i) = e^{-\sum_r \lambda_r f_r(x_i)}$$
(12)

where the λ_r's are some unknown constants to be found. This formula is derived by maximizing the entropy of the probability distribution p_i as satisfying all the constraint given. To search the λ_r's that make $p_i(x)$ satisfy all the constraints, an
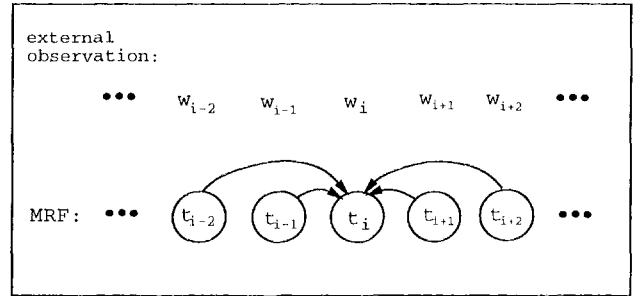

Figure 1: MRF T is defined for the neighborhood system with distance 2

iterative algorithm, "Generalized Iterative Scaling" (GIS), exists, which is guaranteed to converge to the solution (Darroch72+).

(12) is similar to Gibbs distribution, which is the primary probability distribution of MRF model. MRF model uses ME principle in combining information sources and parameter estimation. We will describe MRF model and its parameter estimation method later.

5 MRF-based tagging model

5.1 MRF in tagging

Neighborhood of given random variable is defined by the set of random variables that directly affect the given random variable. Let $N(i)$ denote a set of random variables which are neighbors of ith random variable. Let's define the neighborhood system with distance L in tagging for words $W = w_1, ..., w_n$, and tags $T = t_1, ..., t_n$.

$$N(i) = \{i - L, ..., i - 1, i + 1, ..., i + L\} \quad (13)$$

This neighborhood system has one dimensional relation and describes the one dimenstional structure of sentence. Fig. 1 showes MRF T which is defined for the neighborhood system with distance 2. The arrows represent that the random variable t_i is affected by the neighbors $t_{i-2}, t_{i-1}, t_{i+1}, t_{i+2}$. It also showes that t_i, t_{i-1} and t_i, t_{i+1} have the neighborhood relation connected by bigram, and that t_i, t_{i-1}, t_{i-2} and t_i, t_{i+1}, t_{i+2} have the neighborhood relation connected by trigram.

A *clique* is defined as the set of random variables that all of the pairs of random variables are neighborhood in it. Let's define the clique as the tag sequence with size L in tagging problem.

$$C_i = \{t_{i-L}, t_{i-(L-1)}, ..., t_i\} \quad (14)$$

A clique concept is used to define clique function that evaluates current state of random variables in clique.

The definition of MRF is presented as following.

Definition of MRF: *Random variable T is Markov random field if T satisfies the following two properties.*

Positivity:

$$P(T) > 0, \forall T \qquad (15)$$

Locality :

$$P(t_i|t_j, \forall j, j \neq i) = P(t_i|t_j, \forall j, j \in N(j)) \qquad (16)$$

We assume that every probablity value of tag sequence is larger than zero because ungrammatical sentences can appear in human language usage, including meaningless sequence of characters. So the positivity of MRF is satisfied. This assumption results in the robustness and adaptability of the model, even though untrained events occur.

The locality of MRF is consistent with the assumption of tagging problem in that the tag of given word can be determined by the local context. Consequently, the random variable T is MRF for neighborhood system $N(i)$ as T satisfies the positivity and the locality.

5.2 A Posteriori Probability

A posteriori probability is needed to search for the most likely tag sequence. MRF provides the theoretical background about the probability of the system (Besag74) (Geman84+).

Hammersley-Clifford theorem: *The probability distribution $P(T)$ is Gibbs distribution if and only if random variable T is Markov random field for given neighborhood system $N(i)$,*

$$P(T) = \frac{e^{-\frac{1}{Tm}U(T)}}{Z} \qquad (17)$$

$$Z = \sum_T e^{-\frac{1}{Tm}U(T)} \qquad (18)$$

where Tm is temperature, Z is normalizing constant, called partition function and U(T) is energy function. The a priori probability $P(T)$ of tag sequence T is Gibbs distribution because the random variable T of tagging is MRF.

It can be proved that a posteriori probability $P(T|W)$ for given word sequence W is also Gibbs distribution (Chun93). Consequently, a posteriori probability of T for given W is

$$P(T|W) = \frac{1}{Z'}e^{-\frac{1}{Tm}U(T|W)} \qquad (19)$$

We use (19) to carry out MAP estimation in the tagging model . The energy function $U(T|W)$ is of this form.

$$U(T|W) = \sum_c V_c(T|W) \qquad (20)$$

where V_c is clique function with the property that V_c depends only on those random variable in clique c. This means that energy function can be obtained from each clique funtion which splits the set of random variables to subsets.

6   Clique function design

The more state of random variables are near to the solution, the more the system becomes stable, and energy function has lower value. Energy function represents the degree of unstability of current state of random variables in MRF. It is similar to the behaviour of molecular particles in the real world.

Clique function is proportional to energy function, and it represents the unstability of current state of random variables in clique or it has high value when the state of MRF is bad, low value when the state of MRF is near to solution. Clique function contributes to reduce the computation of evaluation function of entire MRF by clique concept that separates random variables to the subsets.

Clique function $V_i(T|W)$ is described by the features that represent the constraint or information sources of given problem domain.

$$V_i(T|W) = \sum_r \lambda_r f_r^i(T|W) \qquad (21)$$

6.1   MRF Model 1 (Basic model)

The basic information sources which are used in statistical tagging model are unigram, bigram and trigram. MRF model 1 uses unigram, bigram and trigram. We write the feature function of unigram as

$$f_{unigram}^i = (1 - P(t_i|w_i)) \qquad (22)$$

and the feature function of n-gram, including bigram, trigram as

$$f_{n-gram}^i = \sum_{j \in N(i)} (1 - P(t_i|j)) \qquad (23)$$

where

$$P(t_i|j) = \begin{cases} P(t_i|t_{i-j}, t_{i-j+1}, ..., t_{i-1}), & \text{if } i > j \\ P(t_i|t_{i+1}, t_{i+2}, ..., t_{i+j}), & \text{if } i < j \end{cases}$$

The clique function of the model 1 is made as follows.

$$V_i(T|W) = \lambda_1 \cdot f_{unigram} + \lambda_2 \cdot f_{n-gram} \qquad (24)$$

6.2   Model 2 (Morphological information included)

Morphological level information helps tagger to determine the tag of the word, more especially of the unknown word. The suffix of a word gives very useful information about the tag of the word in English. The clique function of model 2 is defined as

$$f_{suffix}^i = (1 - P(t_i|suffix(w_i))) \qquad (25)$$

We used the statistical distribution of the sixty suffixes that are most frequently used in English.

We can expand the clique function of the model 1 easily by just adding suffix information to the clique function of the model 2.

$$V_i(T|W) = \lambda_1 \cdot f_{unigram} + \lambda_2 \cdot f_{n-gram} + \lambda_3 \cdot f_{suffix}$$
(26)

6.3 Model 3 (error correction)

There exist error prone words in every tagging system. We adjust error prone words by collecting the error results and adding more information of the words. The feature function of Model 3 is for adjusting errors in word level.

$$f^i_{error1} = (1 - P(t_i|t_{i-1}, w_i, t_{i+1}))$$
(27)

$$f^i_{error2} = (1 - P(t_i|w_{i-2}, t_{i-1}))$$
(28)

We used the probability distribution of five hundred error prone words in Model 2 in order to reduce the number of parameters.

7 Optimization

The process of selecting the best tag sequence is called as optimization process. We use MAP (Maximum A Posteriori) estimation method. The tag sequence T is selected to maximize the a posteriori probability of tagging (19) by MAP.

Simulated annealing is used to search the optimal tag sequence as Gibbs distribution provides simulated annealing facility with temperature and energy concept. We change the tag candidate of one word selected to minimize the energy function in k-th step from $T^{(k)}$ to $T^{(k+1)}$, and repeat this process until there is no change. The temperature Tm is started in high value and lower to zero as the above process is doing. Then the final tag sequence is the solution. Simulated annealing is useful in the problem which has very huge search space, and it is the approximation of MAP estimation (Geman84+).

There is another algorithm called *Viterbi* algorithm to find optimal solution. Viterbi algorithm guarantees optimal solution but it cannot be used in the problem which has very huge search space. So it is used in the problem which has small search space and used in HMM. MRF model can use both Viterbi algorithm and simulated anealing, but it is not known to use simulated annealing in HMM.

8 parameter estimation

The weighting parameter λ in the clique function (19) can be estimated from training data by ME principle (Jaynes57).

Let us describe ME principle and IIS algorithm briefly. For given $x = (x_1, ..., x_n)$, the corresponding probabilities $p_i(x_i)$ is not known. All we know is the expectation value of the function $f_r(x), (r = 1, 2, ..., m)$:

$$E[f_r(x)] = \sum_{i=1}^{n} p_i(x_i) f_r(x_i)$$
(29)

Given consistent constraints, we can find the probability distribution p_i that makes the entropy $-\sum p_i \ln p_i$ value maximum by using Largrangian multipliers in the usual way, and obtain the result:

$$p_i(x_i) = exp(-\sum_r \lambda_r f_r(x_i))$$
(30)

This formula is almost similar to Gibbs distribution (17), also f_r corresponds to the feature of clique function in MRF (20) (21). Using this fact, we can use ME in parameter estimation in MRF.

We can derive (31) to be used in parameter estimation from training data.

$$-\frac{\partial}{\partial \lambda_r} \ln Z = \sum_i p_i f_r(x_i)$$
(31)

$$Z = \sum_i exp(\sum_r \lambda_r f_r(x_i))$$
(32)

To solve the solution of it, a numerical analysis method GIS (Generlaized Iterative Scaling) was suggested (Darroch72+). Pietra used his own algorithm IIS (Improved Iterative Scaling) based on GIS to induce the features and parameters of random field automatically (Pietra95) . Following is IIS algorithm

IIS(Improved Iterative Scaling)

- Initial data
  A reference distribution $\tilde{p}$, an initial model q_0 and $f_0, f_1, ..., f_n$.
- Output
  q_* and λ by ME estimation

- Algorithm
  (0) Set $q^{(0)} = q_0$
  (1) For each i find λ_i, the unique solution of

$$\sum_T q^{(k)} f_i(T) e^{\lambda_i^{(k)} \sum_r f_r(T)} = \sum_T \tilde{p}(T) f_i(T)$$
(33)

  (2) k ← k+1, set q^{k+1} with new λ_i
  (3) If $q^{(k)}$ has converged, set $q_* = q^{(k)}$ and terminate. Otherwise go to step(1)

where $q^{(k)}$ is the distribution of the model in k-th step, and it corresponds to the posteriori probability of the tagging model (19). λ, the solution of (33) can be obtained by Newton method (Curtis89+), one of numerical analysis method.

The reference distribution $\tilde{p}$ is the probability distribution which is obtained directly from training data. $\tilde{p}$ corresponds to the posterior distribution $P(T|W)$ in the tagging model. We use the

Model	Tagging accuracy (%)
HMM	96.11
MRF(1)	96.2
MRF(2)	96.5
MRF(3)	97.1

Table 1: Measuring the accuracy of HMM and MRF models.

posterior probability of the words sequence of window size n (especially 3 in this model) by counting the entry on training data. Training data means tagged corpus here.

$$\tilde{p} \simeq P(t_1, t_2, ..., t_n | w_1, w_2, ..., w_n) \qquad (34)$$

9 Experiments

The main objective of this experiments is to compare the MRF tagging model with the HMM tagging model. We constructed a MRF tagger and a HMM tagger using same information on the same environment.

It is necessary to do smoothing process for data sparseness problem which is severe in HMM, while MRF has the facility of smoothing in itself like neural-net . We used linear interpolation method (Deroua86+) (jelinek89) and assigning frequency 1 for unknown word (Weisch93+) for smoothing in HMM.

We used the Brown corpus in PennTree Bank, described in (Marcus93+) with 48 different tags. A set of 800,000 words is collected for each part of Brown corpus and used as training data, which is used to build the models. And a set of 30,000 words corpus is used as test data, which is used to test the quality of the models.

Table 1 shows the accuracy of each tagging model. The average accuracy of the HMM-based tagger is similar to that of MRF(1) tagger because they use the same information.

Fig. 2 shows that the error rate as the size of training data is increased. MRF(1) has lower error rate than that of HMM when the size of training data is small. The error rate of MRF(2) is decreased especially when the size of the training data is small, because morphological information helps the process of unknown words. Finally, MRF(3) show improvement as the size of training data grows but converges to the limit on some points.

These experiments show that MRF has better addaptability with small training data than HMM does, and that MRF tagger has less data sparseness problem than HMM tagger.
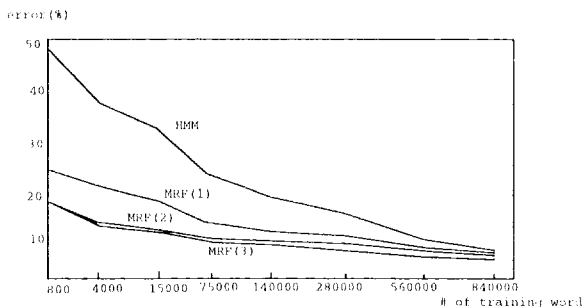

Figure 2: Error rate of each model for given size of training word

10 Comparison of MRF with HMM

We can derive the simplified equation of HMM only with bigram :

$$P(T|W) = P(t_2|t_1)P(t_3|t_2)...P(t_n|t_{n-1}) \qquad (35)$$

(35) is considered as the multiplied probabilties of a the local events. The nearer the probability value of local event is to zero , the more it affects the probability of the entire event. This property strictly reflects on the events which does not occur in training data. But it prohibits even the event that does not occur in training data, although the event is legal.

MRF can be simplified by the summation of clique function as (36).

$$P(T|W) = \frac{1}{Z} e^{-\frac{1}{T_m}\{V_1 + V_2 + ... + V_n\}} \qquad (36)$$

MRF uses evaluation function by summation, while HMM does by multiplication. Even if a clique function value is very bad, other clique function can compensate adequately because the clique functions are connected by summation. There is no critical point of posteriori probablity in MRF, while HMM has critical point in zero value. This property results in the robustness and the adaptability of the model and makes MRF model stronger in data sparseness problem.

11 Conclusion

We proposed a MRF-based tagging model. Information sources for tagging are combined by ME principle which is used in MRF as theoretical background. All parameters used in the model are estimated from training data automatically. As a result, our MRF-based tagging model has better performance than HMM tagging model, especially when the size of the training data is small. We have seen that the performance of the MRF-based tagging model can be improved by adding information to the model.

241

References

Besag, J. "Spatial interaction and the statistical analysis of lattice systems(with discusstion)," *J. Royal Statist. Soc.*, series B, vol. 36, pp. 192-326, 1974.

Besag, J. "On the Statistical Analysis of Dirty Pictures", *J. Royal Statist. Soc.*, vol. B48, 1986.

Brill, E. "A Simple Rule-Based Part of Speech Tagger", In *Proceedings of the 3rd Conf. on Applied Natural Language Processing*, pages 153-155, April, 1992.

Charniak, E., C. Hendrickson, N. Jacobson and M. Perkowitz, "Equations for Part-of Speech Tagging," *Proc. of Nat'l Conf. on Artificial Intelligence(AAAI-86)*, pp. 784-789, 1993.

In, G. Chun, "Range Image Segmentation Using Multiple Markov Random Fields", Ph.D. thesis, KAIST, KOREA, 1993.

Church, K. W., "A Stochastic PARTS Program and Noun Phrase Parser for Unrestricted Text,", *Proceedings of Applied Natural Language Processing*, Austin, Texas, pp. 136-143, 1988.

Darroch, J. N. and D. Ratcliff, "Generalized Iterative Scaling for Long-Linear Models..", *The Annals of Mathematical Statistics*, Volume 43, pages 1470-1480, 1972.

Derouault, A. M. and B. Merialdo, "Natural Language Modeling for Phoneme-to-Text Transcription,", *IEEE Tr. on Pattern Anaysis and Machine Intelligence*, vol. PAMI-8, no.6, Nov. 1986.

Curtis, F. G. and Patric O. Wheatley, "Applied Numerical Analysis", forth edition, ADDISON WESLEY, 1989.

Forney, G. D., "The Viterbi Algorithm", *Proc. of the IEEE*, vol. 61, pp. 268-278, Mar. 1973.

Gamble, E. B., Geiger D. and Possio T., "Integration of Vision Modules and labeling of Surface Discontinuities", *IEEE Transactions on systems, man and cybernetics*, vol. 19, no. 6, November/decemver 1989.

Geman, S. and Geman D., "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images", *IEEE transactions on pattern analysis and machine intelligence*, Vol.PAMI-6, NO. 6, NOVEMBER 1984.

Geiger, D. and Girosi F., "Parallel and Deterministic Algorithms from MRF's: Surface Reconstruction", *IEEE Transactions on pattern analysis and machine intelligence*, VOL 13, NO. 5, MAY 1991.

HUANF, X.D., Y. ARIKI and M.A. JACK, "Hidden Markov Models for Speech Recognition", 1990.

Jaynes, E. T., Information Theory and Statistical Mechanics, *Physics Reviews106*, pages 620-630, 1957.

Jelinek, F. , "Self-Organized Language Modeling for Speech Recognition." , in *Readings in Speech Recognition*, Alex Waibel and Kai-Fu Lee(Editors). Morgan Kaufmann, 1989.

Kupiec, J., Robust Part-of-Speeh Tagging Using a Hidden Markov Model, *Computer Speech and Language*, 1992.

Marcus, M. P., Beatrice Santorini and Mary Ann Marcinkiewiez, "Building a large annotated corpus of English: the Penn Treebank", Computational Linguistics, Vol. 19, No. 2, pp 313-330, June, 1993.

Merialdo, B., "Tagging English Text with a Probabilistic Model" , *Computational Linguistics*, Volume 20, no 2, June 1994.

Nakamura, M., K. Maruyama, T. Kawanata and K. Shikano, "Neural Network Approach of Word Category Prediction for English Texts," *Int'l Conf on Computational Linguistics(Coling-90)*, pp. 213-218, 1990.

Pietra, S. D., V. D. Pietra and J. Lafferty, "Inducing features of random fields", Carnegie Mellon University, Technical report CMU-CS-95-144, MAY, 1995.

Rosenfeld, R., "Adaptive Statistical language Modeling: A Maximum Entropy Approach" , Carnegie Mellon University,technical report CMU-CS-94-138, April 19, 1994.

Weischedel, R., R. Scewartz, J. Ralmucci, M. Meteer, and L. Rawshaw. "Coping with Ambiguity and Unknown Words through Probabilistic Models", *Computational Linguistics*, 19(2):359-382, 1993.

Zhang, J. and J.W. Modestino, "A Markov Random Field model-based approach to image interpretation", *Visual Communications and image Processing IV*, Vol 1199, 1989.