TR-IT-0103

A study of Random Markov Fields for Speech recognition

Helmut Lucke

1995.3

In this report we argue that speech recognition is a two dimension pattern recognition process and that a speech recognizer could possibly benefit from using a two dimensional stochastic process such as a Random Markov Field (MRF) instead of an HMM. We address the problem of training such fields and present an algorithm based on the EM-algorithm. This algorithm is experimentally evaluated for a unconstrained and a constrained MRF. We find that while the algorithm works well in the unconstrained case there are some problems in the constrained version.

1 Introduction – the problem

Stochastic models have become widely used in pattern recognition problems. In speech recognition for example, stochastic language models, such as n-gram or stochastic context-free grammars (SCFGs), or Hidden Markov models (HMMs) as pattern matchers are widely used.

We argued previously [5] that stochastic models can often be represented in the form of a graph and that powerful numeric algorithms for decoding and model-parameter estimation exist if and only if the graph has a specific graph-theoretical property known as the chordality.

Many popular models, such as HMMs, n-grams, fall into this category. However, in more complicated models the chordality condition may not be satisfied. Such cases arise for example when one relaxes the Markov assumption in HMMs, considers stochastic dependencies among different vector components of the of the observation sequence, or considers a more complicated interaction of different knowledge sources.

As we noted before [5], non-chordal stochastic models do not allow the efficient decoding and parameter estimation algorithms associated with HMMs. Such models are generally known as Markov Random Fields (MRFs).

Markov Random Fields arise most naturally in two dimensional pattern recognition problems such as computer vision. Here they have been used by several authors for problems such as image restoration or boundary detection. In most such problems the MRF was parameterized with only a handful parameters and plausible values for these parameters were chosen by the experimenter.

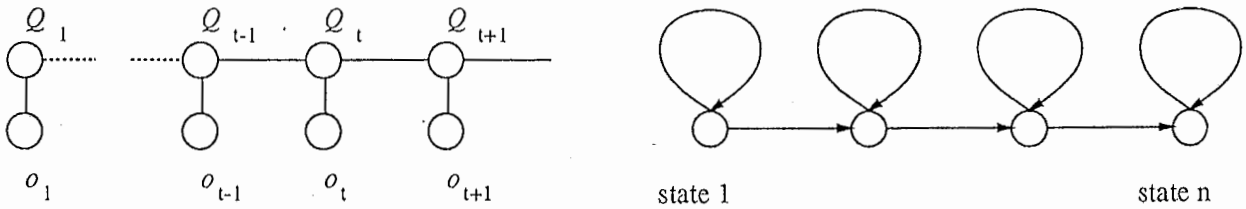
There is currently no coherent theory for the estimation of parameters of RMFs. This report will describe several possible methods for parameter estimation for special kinds of MRFs.

Before discussing the mathematical details of MRFs we will describe an example in speech processing where the application of RMFs would be useful if a workable theory existed.

1.1 RMFs as acoustic models

We regard speech recognition as a two dimensional pattern recognition process. All commonly used low level representations of a speech signal (filter bank, FFT, LPC, autocorrelation, cepstrum) are two dimensional; one dimension in the time the other in the 'frequency' domain. (For simplicity we will refer to the non-temporal domain as the 'frequency' domain, even though, depending on the representation, it may not really be frequency. Similarly the components of one observation vector will be referred to as frequency bins even though they may actually have arisen from some other transform such as cepstrum.)

On the other hand Hidden Markov models, which are commonly used as pattern recognizers, are essentially one-dimensional models. Fig. 1(a) makes this clear. Here the nodes labeled $\mathbf{o}_i$ are observation vectors and the nodes labeled Q_t are random variables that range over the set of all possible states. The horizontal and vertical lines express the statistical dependencies that are implied by the Markov assumption. This representation of a Markov model is to be clearly distinguished from the much more common one shown in Fig. 1(b) in which the states are thought of as sites to be visited by a stochastic finite state machine. We will not use Fig. 1(b) and only included it here to avoid confusion.



(a) (b)

図 1:

Going back to Fig. 1(a) we can therefore regard the HMM as a one dimensional sequence of random variables that is matched to a two dimensional array of pixels (frequency bins). To perform this matching, a 2D to 1D mapping has to be performed at some level. In the early days this was achieved by vector quantization (VQ), nowadays this mapping is usually performed in a more subtle way using continuous HMMs. Nevertheless a number of modeling errors occur as a direct result of this dimensionality mis-match:

Relationship of neighboring components in the frequency domain: A vector with high energy in bin i should be similar to a vector with high energy in bin $i+1$, all other bins being equal. However the VQ or continuous HMM formulation does not take the ordering of frequency bins into account. Any similarity of neighboring components thus has to be learned from examples during training. This greatly increases the amount of training data required to obtain robust models.

Relationship of neighboring components in the time domain: Even if the neighborhood relationship in the frequency domain can be learned by presenting many examples, we loose modeling accuracy in the temporal domain. Suppose we have learned that for a given phoneme, say /a/, high energy occurs either in bin i or $i+1$. Now, if we observe an /a/ with high energy in bin i in frame t we ought to be able to predict that in frame $t+1$ the high energy also occurs in bin i rather than $i+1$. However with current HMMs this is impossible, for at the state level of the HMM bins i and $i+1$ are essentially 'mapped together' so discrimination between the two is no longer possible.

Clearly, a two-dimensional stochastic model that took the stochastic correlations between neighbouring bins in both the time and the frequency domain into account would form a better stochastic model of this process.

2 Random Markov Fields

Let G be a graph with vertex set V and edge set E. Two vertices a, b are adjacent if $(a,b) \in E$. Let $N_a = \{b|(a,b) \in E\}$ be the neighbourhood set of a.

A random field is a set of random variables $\{X_a|a \in V\}$ indexed by the vertex set of G. We say that the random field $\{X_a|a \in V\}$ is a a Markov random field if

$$P(X_a = i|X_b; b \neq a) = P(X_a = i|X_b; b \in N_a) \tag{1}$$

As was mentioned earlier, a general theory for RMFs is still missing. We will therefore narrow our attention to a certain type of graph such as the one shown in figure 2. Here we

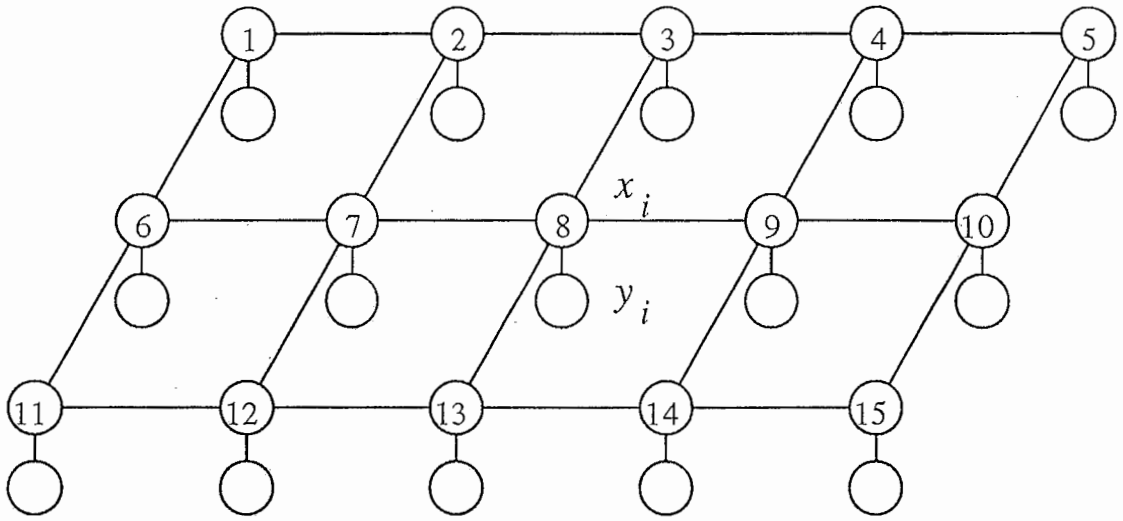


図 2:

have two types of variables (indicated by x and y). The y's stand for an observed field of values (such as the observed time-frequency bins).

3 Parametrization of RMFs

It follows from the Markov assumption (equation 1) that the probability distribution of the stochastic process is a Gibb's distribution (e.g. [1]):

$$P(X_1, \ldots, X_n) = \frac{1}{Z_T} \exp(-U(X)/T) \tag{2}$$

where

$$U(X) = \sum_C u_C(X) \tag{3}$$

Here u_C is a function for each clique C of the graph which is dependent only on the members of C. T is a real positive number known as the temperature from a physical analogy and Z_T is a normalizing constant known as the partition function.

In our case the cliques of the graph come in three different kinds: Horizontal connections between two unobserved nodes, vertical connections between two unobserved nodes and connections between the observed and corresponding unobserved nodes. It follows that we have three types of clique functions. Since we assume translational invariance, the parametrization of the probability distribution is reduced to three clique functions:

$$\begin{array}{ll} U_H(X_i, X_j): & \text{fortwonodesconnectedhorizontally} \\ U_V(X_i, X_j): & \text{fortwonodesconnectedvertically} \\ U_O(X_i, Y_i): & \text{foroapairofobserved/unobservednodesatasite.} \end{array} \tag{4}$$

For our case, equation 2 can therefore be written as:

$$P(X_1, \ldots, X_n, Y_1, \ldots, Y_n) = \frac{1}{Z_T} \exp\left(\frac{-1}{T}\left(\sum_{(i,j)\in S_H} U_H(X_i, X_j) + \sum_{(i,j)\in S_V} U_V(X_i, X_j) + \sum_i U_O(X_i, Y_i)\right)\right) \tag{5}$$

where S_H (resp. S_V) is a set of pairs of indices that are horizontally (resp. vertically) adjacent.

4 Decoding Algorithms

So far we have constructed an algebraic framework for a stochastic model that exhibits the kind of conditional independence assumptions that we are prepared to make. We are now faced with two problems:

1. Given the parameters and an input pattern, what is the most likely state allocation (instantiation of the unobserved nodes)? This is called the decoding problem.

2. Given a set of training patterns, how can we find the best model parameters. This is called the training problem.

In the theory of Hidden Markov Models, identical problems exist and the Baum-Welch algorithm is capable of solving both in way which is optimal in a certain sense. Unfortunately the efficient Baum-Welch algorithm cannot be applied to MRFs. Thus we have to look for alternative solutions. We will first discuss the decoding problem.

4.1 Simulated annealing

Geman and Geman [4] describe an iterative algorithm for this problem. It is based on simulated annealing and is guaranteed to converge to the global optimum (i.e. the maximum of $P(X_1, \ldots, X_n | Y_1, \ldots, Y_n)$ provided that the cooling schedule is chosen sufficiently slow. We will outline the method here.

Initially a state (perhaps random) is assigned to each unobserved variable. Next a temperature T is chosen and each unobserved varible X_i is visited. The order in which sites are visited is not important. For each variable visited the conditional probability distribution $P(X_i | X_j; j \neq i, Y_j) = P(X_i | X_j; j \in N_i, Y_i)$ is calculated. This is easy because only the neighbours of X_i need to be considered. Then a new value for X_i is chosen at random from this distribution. X_i is set to this value and the next site is visited. After all sites have been visited in this way, the temperature is slightly reduced according to a 'cooling schedule' and all sites are visited again. This is repeated until the temperature falls below a certain value.

The cooling schedule proposed by Geman and Geman is $T = \frac{C}{\log(k+1)}$, where C is a constant (about 3) and k is the number of iterations so far.

4.2 Mean Field Theory

Zhang [7] applies the mean field theory from statistical physics to obtain a faster (but perhaps less accurate) algorithm. The idea is, that instead of chosing a state from the probability distribution the probability distribution itself is stored at each site and used in the iterative process.

To summarize the theory (details are described by Chandler [2]), let $U(X)$ be the 'energy' of a configuration (state allocation) of the unobserved variables. In our case

$$U(X) = \sum_{(i,j) \in S_H} U_H(X_i, X_j) + \sum_{(i,j) \in S_V} U_V(X_i, X_j) + \sum_i U_O(X_i, Y_i)$$

We define a new energy at site i as

$$U_i'(X_i) = U(X)|_{u_j = <u_j>, i \neq i}$$

and assume it is of the form

$$U_i'(X_i) = U_i''(X_i) + R_i$$

where

$$U_i''(X_i) = U_H(X_i, < X_E >) + U_H(< X_W >, X_i) + U_V(X_i, < X_N >) + U_V(< X_S >, X_i) + V_O(X_i, Y_i)$$

and X_E etc. indicates the unobserved variable immediatly east of X_i Furthermore the remainder term R_i is assumed to be independent of X_i. Then

$$< X_i > \approx \frac{\sum_{X_i} X_i \exp\left(\frac{-1}{T} U_i'(X_i)\right)}{\sum_{X_i} \exp\left(\frac{-1}{T} U_i'(X_i)\right)} \tag{6}$$

$$= \frac{\sum_{X_i} X_i \exp\left(\frac{-1}{T} U_i''(X_i)\right) \exp\left(\frac{-R_i}{T}\right)}{\sum_{X_i} \exp\left(\frac{-1}{T} U_i''(X_i)\right) \exp\left(\frac{-R_i}{T}\right)} \tag{7}$$

$$= \frac{\sum_{X_i} X_i \exp\left(\frac{-1}{T} U_i''(X_i)\right)}{\sum_{X_i} \exp\left(\frac{-1}{T} U_i''(X_i)\right)} \tag{8}$$

$$\tag{9}$$

This approximation suggests an efficient algorithm. It is similar to the annealing algorithm described above, but at each site the probability distribution $P(X_i|X_j; j \neq i)$ (equivalently $< X_i >$ is stored and (re-)calculated. Experiments show that far fewer iterations are necessary to achieve a reasonable state allocation even if the temperature is kept fixed [7]. However optimality of the solution cannot be guranteed.

5 The parameter training problem

The second important problem is that of parameter estimation. In many previous applications of MRFs (mainly in problems in vision and image restauration) the parameters were kept to a small number which were chosen by hand, thus avoiding the parameter estimation problem.

Previous work in speech recognition has indicated that trainable models such as HMMs perform far superiour to non-trainable ones, so that a training algorithm is required if MRFs are to be applied to speech recognition problems.

A complete theory for the parameter estimation problem is still missing. However in certain cases of symmetry or restriction in complexity solutions do exist. These will be discussed in this section.

Before turning to individual cases we will briefly describe the EM algorithm and point out where the difficulty lie in applying it to MRFs.

5.1 The EM algorithm

Out of possible parameter estimation algorithms, the EM algorithm [3] is the most promissing method for learning the model parameters from training data, because it explicitely distinguishes between the observed and unobserved data. In short the algorithm consists of two steps:

Expectation step (E-step): Given the current parameter estimates Θ find the expectation of the unobserved nodes, i.e. calculate

$$< X|Y >_{\Theta}$$

where X represent the observed variables, Y the unobserved variables and Θ is the parametrization of the Expectation operator.

Maximization step (M-step): Treating the expectation of the unobseved variables calculated previously as if it was observed, find a new parametrization $\bar{\Theta}$ which maximizes the likelihood of the data.

Thus if all variables could be observed, the E-step would not be necessary. Its function is to convert a problem with partially observed data into one with fully observed data.

Since the E-step is based on the previous parameter estimate, the calculation of the expectation of the unobserved variables becomes inaccurate as soon as Θ is replaced by $\bar{\Theta}$. For this reason the algorithm needs to be iterated. However it has been shown that the total likelihood of the training data always increases and that the algorithm converges to a (local) optimum.

As far as MRFs are concerned, the E-step poses no problem and can in fact be solved by the decoding algorithm described in the previous section. The difficulty lies in the M-step because the partition function Z_T is so difficult to calculate. In the E-step the partition function could be ignored because it is a constant in the maximization process. However in the M-step the partition function depends crucialy on the independent variabe (Θ) so that a optimization of Θ is difficult.

5.2 The Baum-Welch algorithm

If the connectivity of the graph is low enough such that the graph is chordal the Baum-Welch algorithm may be applied. This is the case in the case of the HMM. A study of this problem can be found elsewhere [5]. We consider here non-chordal graphs which form a square lattice, i.e. each site is surrounded by four neighbours much like a checker-board.

5.3 Square lattices: Renormalization group theory

This section describes a method for estimating the parameters, which does not require the potential matrices to be symmetric. It is based on ideas taken from the renormalization group theory [6], although it does not in itself use this theory.

Suppose the sites are numbered as in Fig. 2, i.e. such that even and odd numbers form a checkerboard pattern. Then by conditioning on the even numbered sites we can use the

Markov assumption to obtain

$$\begin{aligned} P(X) &= P(X_1, X_2, \ldots) \\ &= P(X_2, X_4, \ldots)P(X_1, X_3, \ldots | X_2, X_4, \ldots) \\ &= P(X_2, X_4, \ldots) \times P(X_1|X_2, X_6) \\ &\quad \times P(X_7|X_6, X_2, X_8, X_{12}) \\ &\quad \times P(X_9|X_8, X_4, X_{10}, X_{14}) \ldots \end{aligned} \tag{10}$$

All the even numbered sites form a rectangular lattice (inclined at 45°) of a coarser resolution. We assume that it is Markov, translationally invariant and can be parameterized by the Gibbs distribution

$$P(X_{\text{even}}) = \frac{1}{Z'} \exp(\sum_{\text{up}} U_U(X_i, X_j) + \sum_{\text{down}} U_D(X_i, X_j))$$

where the first sum is over all pairs of sites in which X_j is to the north-east of X_i and the second sum over all pairs where X_j is to the south-east of X_i.

Suppose that we have already managed to obtain parameter estimates for the U_U and U_D matrices at the coarser resolution. To obtain estimates for U_V and U_H we proceed as follows: Using the most likely state allocation obtained by the stochastic relaxation algorithm we find the statistic $S(k_0, k_W, k_N, k_E, k_S)$ describing the total number of times state k_0 occured surrounded by the states k_W, k_N, k_E, k_S at the nearest neighbor sites. From this after normalization and suitable smoothing we obtain the quantity $Q(k_0|k_W, k_N, k_E, k_S)$ describing the estimated conditional probability of state k_0 occuring in the given context. We now choose U_H and U_V such that

$$\begin{aligned} &\exp\big(U_H(k_W, k_0) + U_H(k_0, k_E) + U_V(k_S, k_0) + U_V(k_0, k_N) \\ &- \tfrac{1}{2}(U_U(k_W, k_N) + U_U(k_S, k_E) + U_D(k_N, k_E) + U_D(k_W, k_S))\big) \\ &\approx Q(k_0|k_W, k_N, k_E, k_S) \end{aligned} \tag{11}$$

We now have approximated the conditional probability $P(X_0|X_W, X_N, X_E, X_S)$ in terms of U_H, U_V, U_U and U_D. In equation 10 $P(X)$ is expressed as the product of such conditional propabilities. Since we made the assumption of shift invariance, we can substitute equation 11 for each of conditional propabiltiy terms in equation 10. Let us first rewrite equation 10 in the form

$$P(X) = P(X_1, X_2, \ldots) \tag{12}$$

$$= P(X_2, X_4, \ldots) \prod_{X_0 = X_1, X_3, \ldots} P(X_0|X_W, X_N, X_E, X_S) \tag{13}$$

where the product runs over all odd-numbered sites and X_W, X_N, X_E, X_S are the four nearest neighbours of X_0.

If we substitute the expression obtained in equation 11 into equation 13 together with the Gibbs distribution for the even numbered sites, we will see that all terms involving U_U and U_D will cancell and we are left with

$$P(X) = \exp(\sum_{\text{hor}} U_H(X_i, X_j) + \sum_{\text{ver}} U_V(X_i, X_j))$$

i.e. we succeeded in finding suitable U_H and U_V parameters. Moreover these are chosen such that the partition function is unity.

The main problem then is to solve equation 11. This is difficult to solve in closed form. However a gradient search method may be used to minimize the square of the error.

We have thus reduced the problem of finding U_V and U_H to that of finding U_U and U_D at a coarser level. We can repeat this procedure and reduce to even coarser grids. At the coarsest level that we wish to consider, we have two possibilities to initiate this inductive procedure:

1. If the lattice is so coarse that it constitutes a singly connected line of sites, we may use the Baum-Welch algorithm, as this is a one dimensional model.

2. Alternatively we may assume that there is no longer any statistical dependence between neighboring sites. Under this assumption all sites are to be treated as independent and the parameters of the potential function can easily be found.

In the experimental work we used the second approach.
We thus have the following algorithm:

For all training epochs
  For all training samples
    Calculate state allocation of training data (decoding)
    update statistic $Q(k_0|k_W, k_N, k_E, k_S)$ at all resolutions.
  For all resolutions
    calculate U_V and U_H (U_U and U_D) based on the estimates at next higher resolution

6 Restricted transitions

The distributions of states over the input array represents in a way a two dimensional labelling of input array. This labelling is to be used in the subsequent recognition process. It makes sense to impose restrictions on the transitions between neighouring states. Suppose for example that there are N states, i.e. each random variable X_i ranges over the integers $\{1, \ldots, N\}$. Then one could impose the following restrictions:

- For X_i is a random variable just south of X_j then $P(X_i = x_i, X_j = x_j) = 0$ if $x_j < x_i$ or $x_j > x_i + 1$.

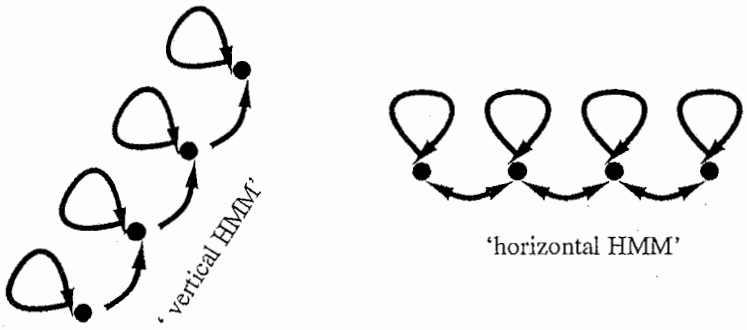
- For X_i is a random variable just west of X_j then $P(X_i = x_i, X_j = x_j) = 0$ if $x_j < x_i - 1$ or $x_j > x_i + 1$.

Such restrictions would force the final assignment of states to look something like

5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
5	5	5	4	4	4	5	5	5	5	5	5	4	4	4	4	5	5	5	5
5	5	4	4	4	4	4	5	5	5	5	4	4	3	4	4	5	5	4	4
5	4	4	3	3	3	3	4	4	4	4	4	3	2	3	3	4	4	4	4
4	3	3	3	2	2	2	3	3	3	3	4	3	2	2	3	4	3	3	3
3	3	2	2	2	2	1	2	3	3	2	3	2	1	1	2	3	3	2	2
2	2	2	2	1	1	1	1	2	2	2	2	1	1	1	1	2	2	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

For example if the underlying representation is a Fourier transform and even states correspond to regions of high energy while odd states correspond to regions of low energy then the resulting system can be regarded as a formant tracker, which is based on a maximum likelihood principle.

Another way to look at restricted transitions is to cut the Markov Random field either horizontally or vertically to obtain just one (horizontal or vertical) chain of random variables. This can then be considered as a traditional Hidden Markov Model. The restricted transitions that were mentioned above would give these models the following model topologies:



Unfortunately we can not directly enforce restricted transitions in the same way we do it in hidden Markov Models because the probabilities are not directly parameters of the model. But we can effectivly enforce them fixing the potential contributions for the horizontal and vertical links to very high values for state combinations that we wish to rule out.

7 Experimental work

A number of experiments were carried out to test the convergence of the training procedure.

For the experiments we used 16 cepstral and delta cepstral coefficients of speech data at a frame rate of 100Hz as the observed data Y. The 'frequency' domain hence consisted of 16 vector valued components, each component being a cepstrum and corresponding delta cepstrum coefficient. $\exp(V)$ was modeled by a 2-variate Gaussians for each state with full covariance matrix.

7.1 Accuracy of equation 11

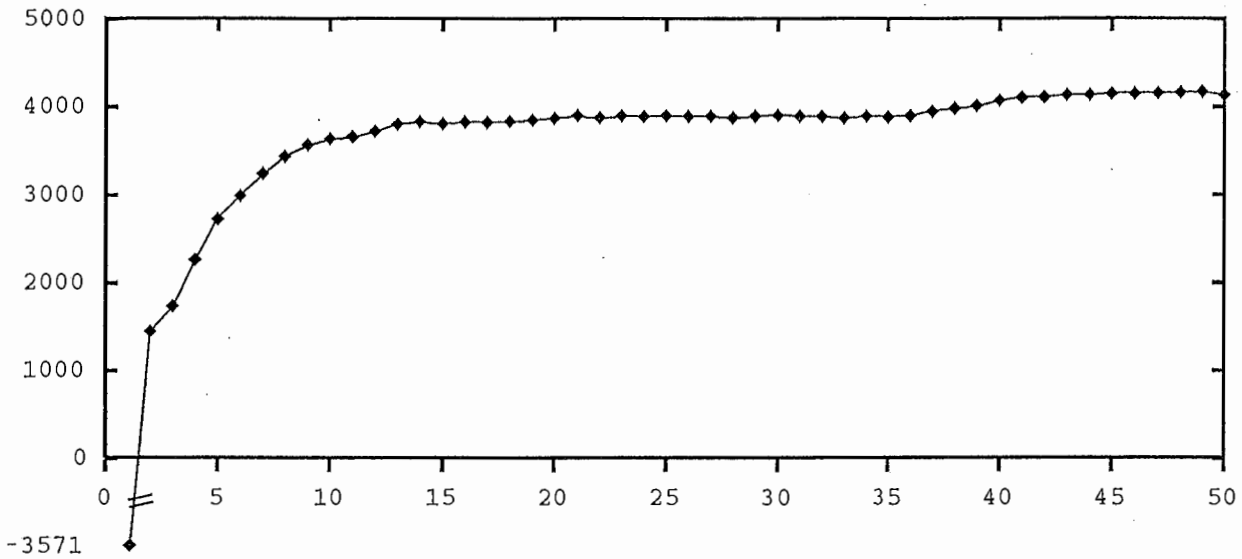
(7.1.1) Unconstrained transitions

We first investigated how accurately Q could be approximated using the gradient search estimation procedure. For this purpose we trained models of the vowel /a/ with 2,3,4,5,6 states. The average estimation errors for these approximation at various levels of coarseness are given below (the coarseness-level is indicated by the Euclidean distance between the nearest neighbor sites):

	2	3	4	5	6
4	15%	9.2%	4.3%	2.2%	1.2%
2.82	17%	11%	5.0%	2.7%	1.5%
2	15%	12%	7.1%	4.1%	2.4%
1.41	17%	14%	7.7%	4.2%	2.7%
1	16%	12%	9.4%	7.1%	4.6%

The numbers suggest that the Q tensor can indeed be accurately approximated using the gradient search, if a sufficient number of states is used. Since in real applications more than 10 states are likely to be used the estimation error should not be significant.

Convergence of the parameter estimation procedure The EM algorithm is guaranteed to converge monotonically. However we are approximating the algorithm in three places: (1) use of stochastic relaxation with a finite cooling schedule, (2) estimation of Q tensor, (3) boundary effects of MRF. Thus the convergence of the algorithm needs to be checked empirically. The following graph shows a plot of the total Gibbs potential of the training data (equivalent to log likelihood) during a training run with 6 states and 50 parameter updates.



Even though convergence is not strictly monotonic, due to the approximations mentioned above, we observe a 'fairly monotonic' behavior.

(7.1.2) Constrained transitions

The same experiments were carried out for constrained transitions. To implement the constraints we fixed the transtions that were not allowed a weight of -10. Since the unconstrained weights usually tend to values around 0 after training, this means that the disallowed transitions occur only with probability e^{-10}.

In the experiments 6 states were used. Some typical state allocations that were achieved after training on real speech patterns are shown in Figs. 3 and 4.

As can be seen one state dominates most of the random variables. This problem also occurs often when training HMM's when the initial conditions are not chosen carefully. In our case we tried different initial consitions and also a state splitting mechanism that would split the state that are used to frequently. However in each case a state dominance was observed. The second problem observable from Figs. 3 is that the state transition constraints were not always observed. (We can see little bubbles in the upper right corner and the lower center. Even varying the penalty for disallowed transitons did not not lead to better results.

A further problem became apparent when we studied the estimation error that is incurred in equation 11. These are shown in the following table:

4	4	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
3	3	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
2	2	2	2	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
1	1	1	1	1	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
2	2	2	2	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
2	2	2	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	4	4	4	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	4	5	5	5	4	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	4	5	5	5	4	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	4	4	4	3	3	3	3	3	3	3	3
2	2	2	2	2	2	2	2	2	2	3	3	3	2	2	2	2	2	2	2	2
1	1	1	1	1	1	1	1	1	1	2	2	2	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

図 3:

5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4
3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
2	2	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
2	2	2	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
1	1	1	1	2	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
0	0	0	0	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

図 4:

5.64	0.1831
4	0.4333
2.82	0.3712
2	0.6336
1.41	0.6396
1	0.7185

Such large estimation errors are really not acceptable and they show that equation 11 can not be justified when using constraints of the form described above.

8 Conclusion

This report described the theory of a two dimensional stochastic model and how it could be applied to speech recognition. Such a model has the potential of modelling the variations in speech more accurately as it allows a 'warped' match in both the time and the frequency domain.

We presented a training method based on ideas from the renormalization group theory for Markov Random fields. Our experiments confirm that if the transitions are unconstrained then the proposed method converges. and produces reasonable model parameters.

However to perform speech recognition some state transitions may have to be constrained. Although such constraints can in principle be enforced by keeping certain parameters at fixed high values, we found that in practise this does not always lead to solutions that observe these constraints and more over we appear to be no longer ableof providing a reasonable approximation in equation 11. Thus reliable parameter estimation may not be possible using this method in this case.

参考文献

[1] Julian E. Besag. Spatial interaction and the statistical analysis of lattice systems. *Royal Statistical Society*, 36(2):192–236, 1974.

[2] D. Chandler. *Introduction to modern statistical Mechanics.* Oxford University Press, 1987.

[3] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Royal Statistical Society Ser. B*, 39:1–38, 1977.

[4] Stuart Geman and Donald Geman. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:721–741, 1984.

[5] Helmut Lucke. Which stocahstic models allow Baum-Welch training? submitted to IEEE Transactions on Signal Processing, June 1994.

[6] Kenneth G. Wilson. The renormalization group and critical phenomena. *Reviews of Modern Physics*, 55(3), July 1983.

[7] Jun Zhang. The mean field theory in EM prodedures for Markov random fields. *IEEE Transactions on Signal Processing*, 40(10), October 1992.