10-708: Probabilistic Graphical Models 10-708, Spring 2014

Discrete sequential models and CRFs

Lecturer: Eric P. Xing *Scribes: Pankesh Bamotra, Xuanchong Li*

1 Case Study: Supervised Part-of-Speech Tagging

The supervised part-of-speech tagging is a supervised task that tags each part of speech sequence with predefined labels, such as noun, verb, and so on. As shown in Figure 1, given the data $D = \{x^{(n)}, y^{(n)}\}$, x represent the speech word and y represents the tag.

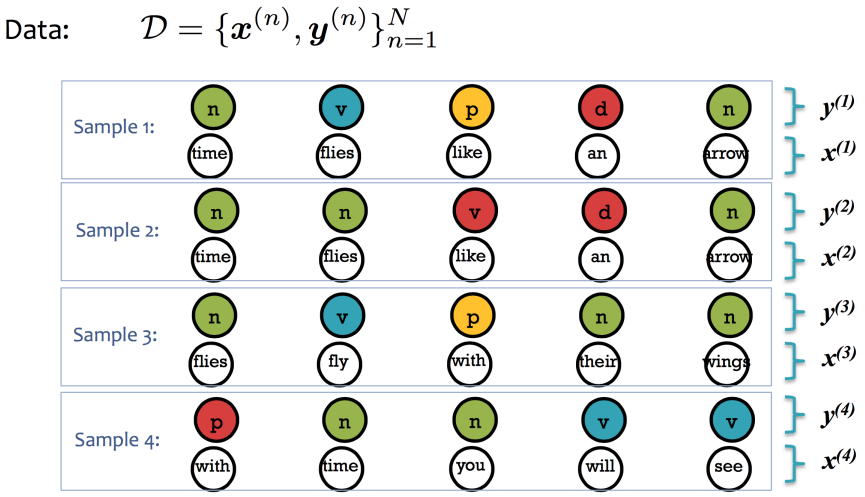

Figure 1: Supervised Part-of-Speech Tagging

This problem can be approached with many different methods. Here we discuss three of them: Markov random field, Bayes network, and conditional random field.

- **Markov Random Field (Figure 2)**: it models the joint distribution over the tags Y_i and words X_i. The individual factors are not probabilities. Thus a normalization Z is needed.
- **Bayes Network (Figure 3)**: it also models the joint distribution over the tags Y_i and words X_i. Note that here the individual factors are probabilities. So $Z = 1$.
- **Conditional Random Field (Figure 4)**: it models conditional distribution over tags Y_i given words X_i. The factors and Z are specific to sentence X.

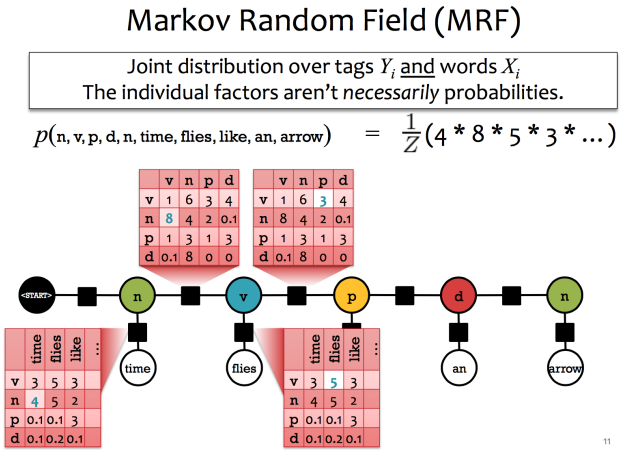

Figure 2: Supervised Part-of-Speech Tagging with Markov Random Field

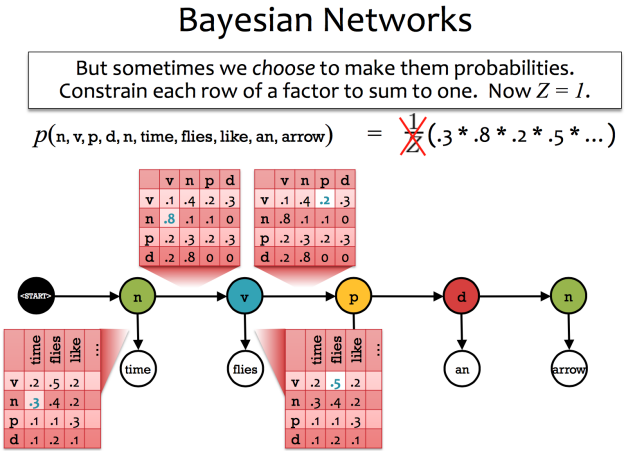

Figure 3: Supervised Part-of-Speech Tagging with Bayes Network

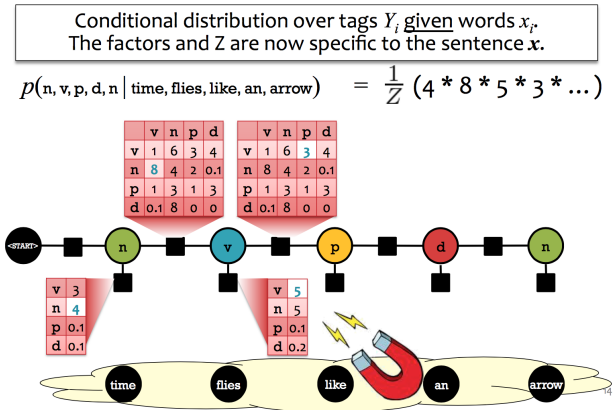

Figure 4: Supervised Part-of-Speech Tagging with Conditional Random Field

2 Review of Inference Algorithm

The forward-backward algorithm (marginal inference) and viterbi algorithm (MAP inference) are two major inference algorithms we have seen so far. It turns out they are all belief propagation algorithms.

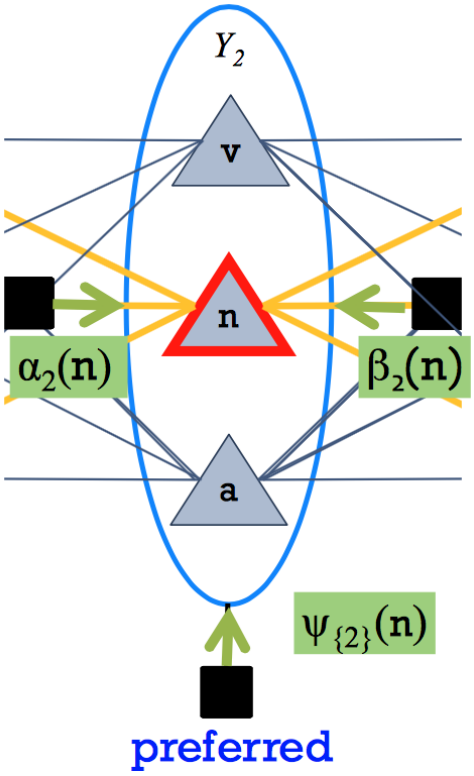

Figure 5: Belief Propagation in Forward-backward Algorithm

For example, in the forward-backward algorithm, as shown in Figure 5, α is the belief from forward pass, the β is the belief from the backward pass, and the ψ is the belief from the x_i. Then the belief of of $Y_2 = n$ is the product of the belief from the three directions: $\alpha(n)\beta(n)\psi(n)$.

3 Hidden Markov Model (HMM) and Maximal Entropy Markov Model (MEMM)

3.1 HMM

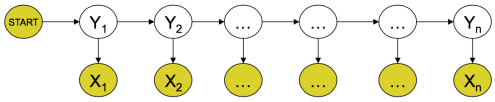

Figure 6: Hidden Markov Model

HMM (Figure 6) is a graphical model with latent variable Y and observed variable X. It is a simple model for sequential data such as language, speech, and so force. But, there are two issues with HMM.

- Locality of feature: HMM models only capture dependencies between each state and its corresponding observation. But in real world problem like NLP, each segmental state may depend not just on a single word (and the adjacent segmental stages), but also on the (non-local) features of the whole line such as line length, indentation, amount of white space, etc.
- Mismatch of objective function: HMM learns a joint distribution of states and observations $P(Y, X)$, but in a prediction task, we need the conditional probability $P(Y|X)$

3.2 MEMM

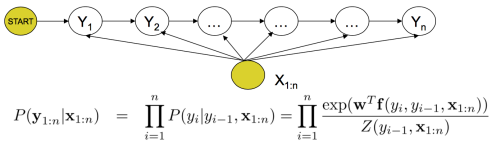

Figure 7: Maximal Entropy Markov Model

MEMM (Figure 7) is for solving the problems of HMM. It gives full observation of sequence to every state, which make the model more experssive than HMM. It is also a discriminative model, since it completely ignores modeling $P(X)$ and the learning objective function consistent with predictive function: $P(Y|X)$. But MEMM has the label bias problem, which means a preference for states with lower number of transitions over others. To avoid this problem, one solution is not normalizing probabilities locally. This improvement gives us the conditional random field (CRF).

4 Comparison between Markov Random Field and Conditional Random Field

4.1 Data

- CRF: $D = \{X^{(n)}, Y^{(n)}\}_{n=1}^{N}$
- MRF: $D = \{Y^{(n)}\}_{n=1}^{N}$

4.2 Model

- CRF: $P(Y|X, \theta) = \frac{1}{Z(X,\theta)} \prod_{c \in C} \psi_c(Y_c, X)$, where $\psi_c(Y_c, X) = \exp(\theta f_c(Y_c, X))$
- MRF: $P(Y|\theta) = \frac{1}{Z(\theta)} \prod_{c \in C} \psi_c(Y_c)$, where $\psi_c(Y_c) = \exp(\theta f_c(Y_c))$

4.3 Log likelihood

- CRF: $l(\theta, D) = \frac{1}{N} \sum_{n=1}^{N} \log(y^n|x^{(n)}, \theta)$
- MRF: $l(\theta, D) = \frac{1}{N} \sum_{n=1}^{N} \log(y^n|\theta)$

4.4 Derivatives

- CRF: $\frac{\partial l}{\partial \theta_k} = \frac{1}{N}\sum_{n=1}^{N}\sum_c f_{c,k}(y_c^{(n)}, x^{(n)}) - \frac{1}{N}\sum_{n=1}^{N}\sum_c \sum_{y_c} p(y_c^{(n)}|x^{(n)}) f_{c,k}(y_c^{(n)}, x^{(n)})$
- MRF: $\frac{\partial l}{\partial \theta_k} = \frac{1}{N}\sum_{n=1}^{N}\sum_c f_{c,k}(y_c^{(n)}, x^{(n)}) - \frac{1}{N}\sum_{n=1}^{N}\sum_c \sum_{y_c} p(y_c^{(n)}) f_{c,k}(y_c^{(n)})$

5 Generative Vs. Discriminative Models - Recap

In simple terms, the difference between generative and discriminative models is the generative models are based on joint distribution $p(y, \mathbf{x})$, while discriminative models are based on the conditional distribution $p(y|\mathbf{x})$. Typical example of generative-discriminative pair [1] is the Naive Bayes classifier and Logistic regression. The principle advantage of using discriminative models is that they can incorporate rich features which can have long range dependencies. For example, in POS tagging we can incorporate features like capitalization of the word, syntactic properties of the neighbour words, and others like location of the word in the sentence. Having such features in generative models generally leads to poor performance of the models.

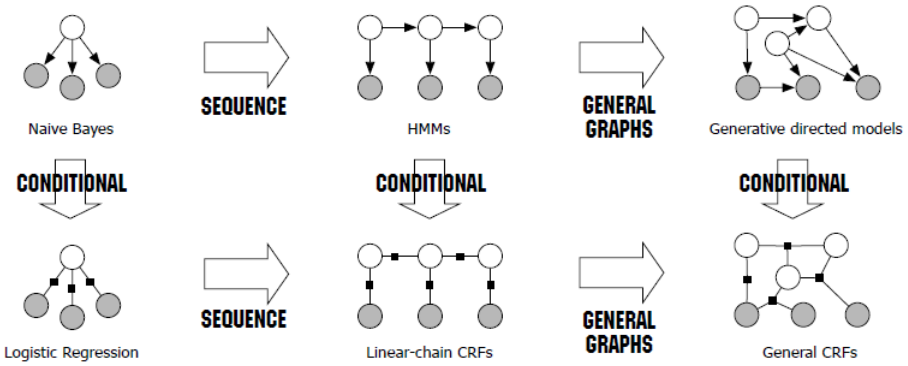

Figure 8: Relationship between some generative-discriminative pairs

6 CRF formulation

Conditional random fields are formulated as below: -

$$P(\mathbf{y}_{1:n}|\mathbf{x}_{1:n}) = \frac{1}{Z(\mathbf{x}_{1:n})}\prod_{i=1}^{n}\phi(y_i, y_{i-1}, x_{1:n}) = \frac{1}{Z(\mathbf{x}_{1:n}, w)}\prod_{i=1}^{n} exp(w^T f(y_i, y_{i-1}, x_{1:n}))$$

An important difference to note here is that CRF formulation looks more or less like the MEMM formulation. However, here the partition function is global and lies outside of the exponential product term.

7 Properties of CRFs

- CRFs are *partially* directed models.

- CRFs are discriminative models like MEMMs.
- CRF formulation has global normalizer Z that helps in overcoming the label bias problem of the MEMMs.
- Being a discriminative model, CRFs can model rich set of features over the entire observation sequence.

8 Linear Chain CRFs

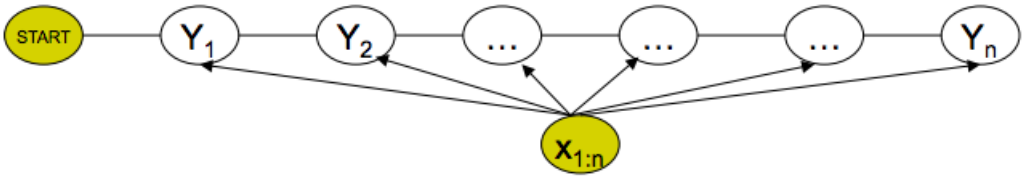

Figure 9: A Linear chain CRF

In a linear chain conditional random field, we model the potentials between adjacent nodes as a Markov Random Field conditioned on input $\mathbf{x}$. Thus, we can formulate linear chain CRF as: -

$$P(\mathbf{y}|x) = \frac{1}{Z(x,\lambda,\mu)} exp(\sum_{i=1}^{n}(\sum_{k}\lambda_k f_k(y_i, y_{i-1}, \mathbf{x}) + \sum_{l}\mu_l g_l(y_i, \mathbf{x})))$$

$$where \;\; Z(x,\lambda,\mu) = \sum_{\mathbf{y}} exp(\sum_{i=1}^{n}(\sum_{k}\lambda_k f_k(y_i, y_{i-1}, \mathbf{x}) + \sum_{l}\mu_l g_l(y_i, \mathbf{x})))$$

f_k and g_l in the above equations are referred to as *feature functions* that can encode rich set of features over the entire observation sequence.

9 CRFs Vs. MRFs

9.1 Model

$$CRF: \quad P(y|x,\theta) = \frac{1}{Z(x,\theta)} \prod_{c} exp(\theta, f(y_c, x))$$

$$MRF: \quad P(y|x,\theta) = \frac{1}{Z(\theta)} \prod_{c} exp(\theta, f(y_c))$$

9.2 Average Likelihood

$$CRF: \quad \tilde{l}(\theta; D) = \frac{1}{N}\sum_{n=1}^{N} log\ p(y_{(n)}|x_{(n)}, \theta)$$

$$MRF: \quad \tilde{l}(\theta; D) = \frac{1}{N}\sum_{n=1}^{N} log\ p(y_{(n)}|\theta)$$

9.3 Derivatives

$$CRF: \quad \frac{d\tilde{l}(\theta;D)}{d\theta_k} = \frac{1}{N}\sum_{n=1}^{N}\sum_{c} f_{c,k}(y_c^{(n)}, x_c^{(n)}) - \frac{1}{N}\sum_{n=1}^{N}\sum_{c}\sum_{y_c} p(y_c|x_c^{(n)}) f_{c,k}(y_c, x_c^{(n)})$$

$$MRF: \quad \frac{d\tilde{l}(\theta;D)}{d\theta_k} = \frac{1}{N}\sum_{n=1}^{N}\sum_{c} f_{c,k}(y_c^{(n)}) - \frac{1}{N}\sum_{n=1}^{N}\sum_{c}\sum_{y_c} p(y_c) f_{c,k}(y_c)$$

10 Parameter estimation

As we saw in the previous sections CRFs have parameters λ_k and μ_k which we estimate from the training data $D = (x^{(n)}, y^{(n)})_{i=1}^{N}$ having empirical distribution $\tilde{p}(x, y)$. We can use iterative scaling algorithms and gradient descent to maximizes the log-likelihood function.

11 Performance

model	*error*	*oov error*
HMM	5.69%	45.99%
MEMM	6.37%	54.61%
CRF	5.55%	48.05%
MEMM$^+$	4.81%	26.99%
CRF$^+$	4.27%	23.76%

Figure 10: CRFs vs. other sequential models for POS tagging on Penn treebank

12 Minimum Bayes Risk Decoding

Decoding in CRFs refers to choosing a particular $\mathbf{y}$ from $p(y|x)$ such that some form of loss function is minimized. A minimum Bayes risk decoder returns an assignment that minimizes expected loss under model distribution. This can be represented as: -

$$h_\theta(x) = argmin_y \sum_{y} p_\theta(y|x) L(\hat{y}, y)$$

Here $L(\hat{y}, y)$ represents a loss function like 0-1 loss or Hamming loss.

13 Applications - Computer Vision

A few applications of CRFs in computer vision are: -

- Image segmentation

- Handwriting recognition
- Pose estimation
- Object recognition

13.1 Image segmentation

Image segmentation can be modelled by conditional random fields. We exploit the fact that foreground and background portions of an image have pixel-wise and local characteristics that can be incorporated as CRF parameters. Thus, image segmentation can be formulated as: -

$$Y^* = argmax_{y \in (0,1)^n} \left[\sum_{i \in S} V_i(y_i, X) + \sum_{i \in S} \sum_{j \in N_i} V_{i,j}(y_i, y_j) \right]$$

Here, Y refers to image label as foreground or background, Xs are data features, S are pixels, and N_i refers to the neighbours of pixel i.