Pairwise HMM

Taikai Takeda

Waseda University, Japan
297.1951@gmail.com

Abstract. .

1 Pairwise Alignment

Pairwise Alignment is to align two sequences of, for example, DNA, by inserting gaps inbetween the elements of the sequences. The goal of this task is to maximize a score of the alignment so that we can choose the best alignment of all the possible ones.

...

2 HMM

HMM (Hidden Markov Model) has been widely used for from gene alignment to speech recognitions. Let us introduce simple HMM before presenting Pairwise HMM.

2.1 Formulation

Let $\mathcal{D} = \boldsymbol{X} = (X_1, ..., X_T)$ and $\boldsymbol{Z} = (Z_1, ..., Z_T)$ be, respectively, observed and hidden random variables. Let $\mathcal{A}$ be a set of simbols and a set of hidden states be $\mathcal{S}$. Input data is a set of sequences, $\boldsymbol{x} = (\boldsymbol{x}^1, ..., \boldsymbol{x}^N)$ where n-th sequence $\boldsymbol{x}^n \in \mathcal{A}^{T_n}$ is T_n is the length of the sequence. Similarly, hidden states are denoted as $\boldsymbol{z} = (\boldsymbol{z}^1, ..., \boldsymbol{z}^N)$ where n-th sequence $\boldsymbol{z}^n \in \mathcal{S}^{T_n}$. Note that we sometimes omit superscript n when concentrating on a single sequence for the sake of notational simplicity. The corresponding joint disribution has the form

$$p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta}) = p(Z_1|\boldsymbol{\alpha}) \prod_{t=2}^{T} p(Z_t|Z_{t-1}, \boldsymbol{\beta}) \prod_{t=1}^{T} p(X_t|Z_t, \boldsymbol{\phi}) \tag{1}$$

where $\boldsymbol{\theta} = \{\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\phi}\}$. $p(Z_1|\boldsymbol{\alpha})$, $p(Z_t|Z_{t-1}, \boldsymbol{\beta})$ and $p(X_t|Z_t, \boldsymbol{\phi})$ are an initial probability, a transition probability, an emission probability, respectively. They are described, respectively, as $p(Z_1 = k, \boldsymbol{\alpha}) = \alpha_k$, $p(Z_t = k|Z_{t-1} = j, \boldsymbol{\beta}) = \beta_{jk}$ and $p(X_t|Z_t = k, \boldsymbol{\phi}) = p(X_t|\phi_k) = \psi_{tk}$ [1]. $\boldsymbol{\alpha} = \{\alpha\}_k$ is $K-$dimensional vector and $\boldsymbol{\beta} = \{\beta\}_{jk}$ is $K \times K$ matrix where K is the number of hidden states

[1] MEMO: define it later (not in this subsection)

2.2 Forward-Backward Algorithm

We here discuss how to compute the smoothed marginal $p(z_t = j|\boldsymbol{x})$ and tghe smoothed two-sliced marginal $p(z_{t-1}, z_t|\boldsymbol{x})$.[2]

Taking a look at the graphical model in Fig(), we can see conditioning on z_t eable to decompose joint distribution into two parts: the past and the future.

$$p(z_t = k|\boldsymbol{x}) \propto p(z_t = k, \boldsymbol{x}_{t+1:T}|\boldsymbol{x}_{1:t}) \propto p(z_t = k|\boldsymbol{x}_{1:t})p(\boldsymbol{x}_{t+1:T}|z_t = k) \tag{2}$$

Let us define forward variables $f_{t,k} = p(z_t = k|\boldsymbol{x}_{1:t})$, the bilief of the state given all the previous sequence. Also, define backward variables $b_{t,k} = p(\boldsymbol{x}_{t+1:T}|z_t = k)$, the conditional likelihood of future evidence give the hidden staetes z_t. Forward variables are efficiently computed using dynamic programming. The base case and the recursive relationship is given as follows:

$$\begin{aligned}
f_{t,k} &= p(z_t = k|\boldsymbol{x}_{1:t}) \\
&= \frac{p(z_t = k, x_t|\boldsymbol{x}_{1:t-1})}{p(x_t|\boldsymbol{x}_{1:t-1})} \\
&= \frac{p(x_t|z_t = k, \cancel{\boldsymbol{x}_{1:t-1}})p(z_t = k|\boldsymbol{x}_{1:t-1})}{p(x_t|\boldsymbol{x}_{1:t-1})} \\
&= \frac{p(x_t|z_t = k)\sum_{j=1}^{K} p(z_t = k|z_{t-1} = j)p(z_{t-1} = j|\boldsymbol{x}_{1:t-1})}{p(x_t|\boldsymbol{x}_{1:t-1})} \\
&= p(x_t|z_t = k)\sum_{j=1}^{K} \beta_{j,k} f_{t-1,k} && (3) \\
f_{1,k} &= p(z_1 = k) = \alpha_k && (4)
\end{aligned}$$

[3] Similarly, backward variables are computed using following equations:

$$\begin{aligned}
b_{t-1,j} &= p(\boldsymbol{x}_{t:T}|z_{t-1} = j) \\
&= \sum_{j=1}^{K} p(\boldsymbol{x}_{t:T}, z_t = j|z_{t-1} = j) \\
&= \sum_{j=1}^{K} p(z_t = k|z_{t-1} = j)p(\boldsymbol{x}_{t:T}|z_t = k, \cancel{z_{t-1} = j}) \\
&= \sum_{j=1}^{K} p(z_t = k|z_{t-1} = j)p(x_t|z_t = k)p(\boldsymbol{x}_{t+1:T}|z_t = k) \\
&= \sum_{j=1}^{K} \beta_{j,k}\psi_{t,k}b_{t,k} && (5)
\end{aligned}$$

[2] Note that in online setting, we can only compute $p(z_t = j|\boldsymbol{x}_{\mathbf{1:t}})$, so called filtered marginal, but we concentrate on the offline setting here.

[3] MEMO: should we define emission notation e.g. $\psi_{t,k}$

Now, we can compute smoothed posterior using forward and backward variables. Let us denote smoothed posterior $\gamma_{t,k} = p(z_t = k|\boldsymbol{x}_{1:T})$ and smoothed two-sliced marginal $\xi_{t,j,k} = p(z_{t-1}, z_t|\boldsymbol{x})$

$$\gamma_{t,k} \propto f_{t,k} b_{t,k} \tag{6}$$

Also, smoothed two-sliced marginal is computed as follows:

$$\begin{aligned}
\xi_{t,j,k} &= p(z_{t-1}, z_t|\boldsymbol{x}_{1:T}) \\
&\propto p(z_t, z_{t-1}, \boldsymbol{x}_{t:T}|\boldsymbol{x}_{1:t-1}) \\
&= p(z_{t-1}|\boldsymbol{x}_{1:t-1}) p(z_t, \boldsymbol{x}_{t:T}|z_{t-1}, \cancel{\boldsymbol{x}_{1:t-1}}) \\
&= p(z_{t-1}|\boldsymbol{x}_{1:t-1}) p(z_t|z_{t-1}) p(\boldsymbol{x}_{t:T}|z_t, \cancel{z_{t-1}}) \\
&= p(z_{t-1}|\boldsymbol{x}_{1:t-1}) p(z_t|z_{t-1}) p(x_t|z_t) p(\boldsymbol{x}_{t+1:T}|z_t, \cancel{x_t}) \\
&= f_{t-1,j} \beta_{jk} \psi_{tk} b_{t,k} && (7)
\end{aligned}$$

2.3 Parameter Optimizations via EM

The paramters can be learned from the dataset using EM (Expectation Maximization), which is also called Baum-Welch specifically for HMM. Likelihood function $l(\boldsymbol{\theta}) = p(\boldsymbol{X}|\boldsymbol{\theta}) = \sum_Z p(\boldsymbol{X}, \boldsymbol{Z})$ is hard to optimize because it includes partition function over all the possible states of the hidden states. EM, however, can optimize parameters through iterative procedure if the joint distribution over the observed and hidden variables is easy to compute. We first explain EM in general case before moving on EM for HMM. EM iterate E-step (Expectation step) and M-step (Maximization step) in order. Define the complete data log likelyhoood to be

$$l_c(\boldsymbol{\theta}) = \sum_{n=1}^{N} \ln(\boldsymbol{x}^n, \boldsymbol{z}^n|\boldsymbol{\theta}) \tag{8}$$

Note that $\boldsymbol{z^n}$ is actually not given. We further define auxiliary function $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old})$

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) = E_{p(\boldsymbol{Z}|\mathcal{D}, \boldsymbol{\theta}^{old})}[l_c(\boldsymbol{\theta})] \tag{9}$$

The E-step computes the expectation of complete log likelihood $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old})$, then the M-step finds $\boldsymbol{\theta}$ that maximize the computed expectation.

$$\boldsymbol{\theta}^{new} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) \tag{10}$$

Now we can apply EM for HMM. The complete data log likelihood $l_c(\boldsymbol{\theta})$ is written down as

$$l_c(\boldsymbol{\theta}) = \sum_{n=1}^{N} \left[\ln p(z_1^n|\boldsymbol{\alpha}) + \sum_{t=2}^{T} \ln p(z_t^n|z_{t-1}^n, \boldsymbol{\beta}) + \sum_{t=1}^{T} \ln p(x_t^n|z_t^n, \boldsymbol{\phi}) \right] \tag{11}$$

The auxilary function $Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old})$, the expected complete log likelihood, is given by

$$\begin{aligned} Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) &= E_{p(\boldsymbol{Z}|\boldsymbol{x},\boldsymbol{\theta}^{old})}[l_c(\boldsymbol{\theta})] \\ &= \sum_{n=1}^{N} \left[\sum_{k=1}^{K} \gamma_{t,k}^{n} \ln \alpha_k + \sum_{t=2}^{T} \sum_{j=1}^{K} \sum_{k=1}^{K} \xi_{t,j,k}^{n} \ln \beta_{j,k} + \sum_{t=1}^{T} \sum_{k=1}^{K} \gamma_{t,k}^{n} \ln \psi_{t,k} \right] \end{aligned} \tag{12}$$

In the M step, we optimize Q w.r.t. $\boldsymbol{\theta} = \{\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\phi}\}$. Firstly, let us consider the optimization of $\boldsymbol{\alpha}$. Using Lagrange Multiplier with the constraint $\sum_k \alpha_k = 1$, we obtain the optimal $\boldsymbol{\alpha}$.

$$L_{\boldsymbol{\alpha}}(\boldsymbol{\theta}, \lambda) = Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) + \lambda(1 - \sum_k \alpha_k) \tag{13}$$

$$\frac{\partial L_{\boldsymbol{\alpha}}(\boldsymbol{\theta}, \lambda)}{\partial \alpha_k} = \frac{\sum_{n=1}^{N} \gamma_{t,k}^{n}}{\alpha_k} - \lambda \tag{14}$$

$$\frac{\partial L_{\boldsymbol{\alpha}}(\boldsymbol{\theta}, \lambda)}{\partial \lambda} = (1 - \sum_k \alpha_k) \tag{15}$$

Seeking stationary point, that is $\partial L_{\boldsymbol{\alpha}}(\boldsymbol{\theta}, \lambda)/\partial \alpha_k = \partial L_{\boldsymbol{\alpha}}(\boldsymbol{\theta}, \lambda)/\partial \lambda = 0$, we obtain the optimal initial probability α_k^*.

$$\alpha_k^* = \frac{\sum_{t=1}^{T} \gamma_{t,k}^{n}}{\sum_{t=1}^{T} \sum_{l=1}^{K} \gamma_{t,l}^{n}} \tag{16}$$

Similary, using appropriate Lagrange Multiplier, we obtain the optimal transition probability β_k^*.

$$\beta_{j,k}^* = \frac{\sum_{T=2}^{T} \xi_{t-1,t,j,k}}{\sum_{t=2}^{T} \sum_{l=1}^{K} \xi_{t-1,t,j,l}} \tag{17}$$

Assume the emission probability is categorical distribution

$$p(x_t = m | z_t = k, \boldsymbol{\phi}) = \mu_{k,m} \tag{18}$$

where $\boldsymbol{\phi} = \{\boldsymbol{\mu}\}$. $\boldsymbol{\mu}$ is $K \times M$ matrix where M is the number of the categories. Lagrange function of the emission parameter with corresponding constraint and its partial derrivatives are given by

$$L_{\boldsymbol{\mu}}(\boldsymbol{\theta}, \boldsymbol{\lambda}) = Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) + \sum_{k=1}^{K} \lambda_k (1 - \sum_{m=1}^{M} \mu_{k,m}) \tag{19}$$

$$\frac{\partial L_{\boldsymbol{\mu}}(\boldsymbol{\theta}, \boldsymbol{\lambda})}{\partial \mu_{k,m}} = \frac{N_{k,m}}{\mu_{k,m}} \tag{20}$$

$$\frac{\partial L_{\boldsymbol{\mu}}(\boldsymbol{\theta}, \boldsymbol{\lambda})}{\partial \lambda_k} = 1 - \sum_{m=1}^{M} \mu_{k,m} \tag{21}$$

where $\boldsymbol{\lambda} = (\lambda_1, ..., \lambda_K)$ is vector of Lagrange Multipliers. $N_{m,k}$ is the weighted counts of output categories given by

$$N_{k,m} = \sum_{n=1}^{N} \sum_{t=1}^{T_n} \gamma_{t,k}^{n} \delta_{x_t^n, m} \tag{22}$$

where $\delta_{i,j}$ is Kronecker delta[4]. Now we have the optimal parameter $\mu_{k,m}^{*}$ by seeking stationary point.

$$\mu_{k,m}^{*} = \frac{N_{k,m}}{\sum_{m=1}^{M} N_{k,m}} \tag{23}$$

2.4 Viterbi decoding

Choose the optimal sequence of hiden states. ...

3 Pairwise HMM

Pairwise Hidden Markov Model (PHMM) is probablistic generative model used for pairwise sequence alignment. Given transition and emission probability distributions, it can compute likelyhood of 'similarity' as well as the most probable alignment. Furthermore, it is possible to optimize parameters by iterative procedure (EM algorithm). This one of discrete HMMs, but different from them as the length of hidden states changes dinamically according to the alighment. We will introduce PHMM using analogy to simple HMM.

3.1 Formulation

The main difference of PHMM from HMM comes from the uncertain alinment. We have to marginalize over all the possible alignment in order to obtain, for example, marginal likelihood.

Let us consider the simpliest configuration of the hidden states, in which the model has three states $\mathcal{S} = \{M, X, Y\}$, Match state M, X insertion state X, and Y insertion state Fig().

Let $\mathcal{D} = \{\boldsymbol{x}, \boldsymbol{y}\}$ be observed data, each of which contains N sequences $\boldsymbol{x} = (\boldsymbol{x}^1, ..., \boldsymbol{x}^N)$, $\boldsymbol{y} = (\boldsymbol{y}^1, ..., \boldsymbol{y}^N)$. The n-th sequences are $\boldsymbol{x}^n = (x_1^n, ..., x_{T_x^n}^n)$ and $\boldsymbol{y}^n = (y_1^n, ..., y_{T_{yn}}^n)$ where T_x^n and T_y^n are the length of the $\boldsymbol{x}^n$ and $\boldsymbol{y}^n$, respectively. Unlike the normal HMM, we here introduce 2d-grid hidden states to keep the notation simple. Let $\boldsymbol{w} = (\boldsymbol{w}^1, ..., \boldsymbol{w}^N)$ be hidden states. $\boldsymbol{w}^n$ is 2d-grid hidden variables of size $T_x^n \times T_y^n$.

3.2 EM

[4] should use Iverson bracket? (because we might use simbol for input and hidden states)

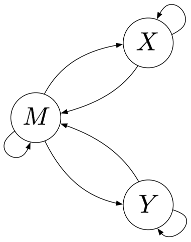

Fig. 1: The simpliest configuration of hidden states