Latent Models for Word Alignments *without Magics*

Hieu Pham
hyhieu@cmu.edu

Fall 2017

Abstract

I derive the formulas for Hidden Markov Model (HMM) that you will be using for Project 4: Word Alignment. The goal of this note is to demystify any potential confusions that you may have with the mathematical foundations needed for this assignment. This means that every single equation written in this note should in self-explicable, and you shouldn't need to consult any other notes to understand what's going on. However, if you want to skip those derivations, you should still be able to do the assignment, if you just know how to translate the pseudo code I provide in Algorithm 1 into Java. Finally, after we wrestle to understand HMM, I will derive IBM Model 1, which you also have to implement, as a simple instance of HMM.

1 Notations

In the problem of word alignment, we want to model the joint conditional probability $p(\mathbf{a}, \mathbf{f}|\mathbf{e})$ where $\mathbf{f}$ is a sentence in French, $\mathbf{e}$ is a sentence in English and $\mathbf{a}$ is called the *alignment vector*. Let $I = |\mathbf{e}|$ and $J = |\mathbf{f}| = |\mathbf{a}|$ be the length of $\mathbf{e}$, $\mathbf{f}$, and $\mathbf{a}$ respectively. We also assume that each value a_j of $\mathbf{a}$ has to take a value in $\{0, 1, ..., I-1, \text{null}\}$. These values correspond to the position that the word f_j in $\mathbf{f}$ comes from, or null, which means that f_j comes from grammatical rules and not from $\mathbf{e}$.

A quick digression. We have discussed in class the *noisy channel model,* which models $p(\mathbf{f}|\mathbf{e})$ if you want to translate $\mathbf{e}$ into $\mathbf{f}$. However, let's leave that for another story, since considering the full translation model in this note would just create more confusions.

In Section 2, we describe how to use a Hidden Markov Model (HMM) to model $p(\mathbf{a}, \mathbf{f}|\mathbf{e})$. We derive all the formulas you will use to train the HMM, and also give the pseudocode. Then, in Section 3, we show that the IBM Model 1 is just an instance of HMM.

2 Hidden Markov Model (HMM)

In an HMM, we have a sequence of hidden states, which is modeled by the *transition probabilities*. Based on this sequence of hidden states, we model the observation by the *emission probabilities*. To see how to fit this into our context, let us reiterate that we are trying to model the *joint conditional probability* $p(\mathbf{a}, \mathbf{f}|\mathbf{e})$, and thus all the computations that follow will be conditioned on the English sentence $\mathbf{e}$. Naturally, the alignment vector $\mathbf{a}|\mathbf{e}$ corresponds to the latent sequence in our HMM.

HMM makes the *independence assumption* and the *Markov assumption*. The independence assumption says that each f_j depends only on a_j and nothing else; the Markov assumption says that each $j > 0$, a_j depends only on a_{j-1} and nothing else. For the special case, a_0 has a prior distribution. Under these assumptions, we can rewrite $p(\mathbf{a}, \mathbf{f}|\mathbf{e})$ as follows

$$p(\mathbf{a}, \mathbf{f}|\mathbf{e}) = p(a_0|\mathbf{e}) \prod_{j=1}^{J-1} p(a_j|a_{j-1}, \mathbf{e}) \cdot \prod_{j=0}^{J-1} p(f_j|a_j, \mathbf{e}) \tag{2.1}$$

2.1 Parametrization

As with any machine learning model, our HMM defines a set of parameters such that based on these parameters, we can compute $p(\mathbf{f}|\mathbf{e})$ from Eqn 2.1. This process is called *parametrization*. To parametrize our HMM, we have to define our parameters so that for any $\mathbf{f}$ and $\mathbf{e}$, we can compute the followings

1. $p(a_0|\mathbf{e})$,
2. $p(a_j|a_{j-1}, \mathbf{e})$ for each $j = 1, 2, ..., J-1$,
3. $p(f_j|a_j, \mathbf{e})$ for each $j = 0, 1, ..., J-1$.

Note that how we define our parameters, and how we compute the terms above are completely left at our decisions. If we make a bad decision, then our translation quality suffers, but we would still have a probabilistically correct model.

Model $p(f_j|a_j, \mathbf{e})$ for each $j = 0, 1, ..., J-1$. We start with (3) since it's the easiest to understand. For each English word e and each French word f, we define a parameter $\theta_{f,e}$. Formally, we do *not* give it any meaning, but intuitively we can *think of* $\theta_{f,e}$ as the probability of f and e are the dictionary translations of each other. Please don't let this intuition blur your maths. With $\theta_{f,e}$ as defined, we resolve (3) by *assigning*

$$p(f_j|a_j, \mathbf{e}) \overset{\text{def}}{=} \theta_{f_j, e_{a_j}} \tag{2.2}$$

Please convince yourself of the following points

- Eqn 2.2 has utterly no mathematical cause. We just made it up by defining $\theta_{f,e}$ for any f, e and then insisting that the term $p(f_j|a_j, \mathbf{e})$ in the left-hand side takes the value in the right-hand side. In fact, if you wish, you can throw a neural network into Eqn 2.2 and it would still work, but that's for another story.
- The right-hand side of Eqn 2.2 requires that you know $\mathbf{e}$. Otherwise, without $\mathbf{e}$, saying e_{a_j} makes completely no sense. That's why all the alignments have to be conditioned on an English sentence $\mathbf{e}$. This usage of a_j also requires that $a_j \in \{0, 1, ..., I-1\}$ for each $j = 0, 1, ..., J-1$, which will be important when we compute other terms in Eqn 2.2.

To sanity check that you understand Eqn 2.2 and HMM so far, please answer the following question (honestly, don't look at the answers before attempting)

- How many parameters $\theta_{f,e}$ are there? *Answer:* #(English words) × (#(French words) + 1)
- Are there any requirements for $\theta_{f,e}$? *Answer:* Yes. In order for $p(f_j|a_j, \mathbf{e})$ Eqn 2.2 to be valid conditional probabilities, we must have $\theta_{f,e} \geq 0$ and for each e

$$\sum_f \theta_{f,e} = 1 \tag{2.3}$$

Model $p(a_j|a_{j-1}, \mathbf{e})$ for each $j = 1, 2, ..., J-1$. We start by assuming that all the English sentences $\mathbf{e}$ that we will ever produce with our translation model have at most $I_{\max}$ words. In practice, if we have a longer sentence, we can break it into chunks using some heuristics and translate these chunks separately. Now, as with $\theta_{f,e}$, we define $I_{\max}$ parameters ψ_0, ψ_1, ..., $\psi_{I_{\max}}$, as well as an extra hyper-parameter ε (in this context, hyper-parameter is just a parameter but we give it a fixed value that never changes). Then, with ψ and ε, which is not necessarily associated to any meanings, we define

$$p(a_j|a_{j-1}, \mathbf{e}) \overset{\text{def}}{=} \begin{cases} \dfrac{(1-\varepsilon) \cdot \psi_{|a_j - a_{j-1}|}}{Z_j} & \text{if } 0 \leq a_j \leq I-1 \\ \varepsilon & \text{if } a_j \text{ is null} \\ 0 & \text{otherwise} \end{cases}, \text{ where } Z_j = \sum_{i=0}^{I-1} \psi_{|i - a_{j-1}|} \tag{2.4}$$

Just like Eqn 2.2, Eqn 2.4 is totally made up by us. Eqn 2.4 is consistent with all our discussions so far, since the alignment vector $\mathbf{a}|\mathbf{e}$ has to have all its components in $\{0, ..., I-1\}$, guaranteeing that the absolute difference $|a_j - a_{j-1}|$ always belongs to $\{0, 1, ..., I-1\}$, so we never index anything in ψ that is out of bound. As with $\theta_{f,e}$, the parameters ψ_d have to satisfy some conditions in order for Eqn 2.4 to be mathematically consistent. In our case, they are that $\varepsilon \in [0, 1]$ and that $\psi_d \geq 0$. This is a nice behavior of ψ, as we no longer need ψ to sum to 1. Note that now we have

$$\sum_{a_j=0}^{I-1} p(a_j|a_{j-1}, \mathbf{e}) = 1 - \varepsilon, \tag{2.5}$$

so we can think of ε as the probability that the word f_j does not come from any e_i. In lectures, we call this the *null alignment*. Nevertheless, this is merely an interpretation, and it does not affect our mathematical reasonings.

Model $p(a_0|\mathbf{e})$. Fortunately, with ψ and ε defined, we decide not to create new parameters and instead, just do

$$p(a_0|\mathbf{e}) \overset{\text{def}}{=} \begin{cases} \varepsilon & \text{if } a_0 \text{ is null} \\ \dfrac{(1-\varepsilon)\cdot\psi_{a_0}}{\sum_{i=0}^{I-1}\psi_i} & \text{otherwise} \end{cases} \tag{2.6}$$

2.2 Learning with EM

We discuss how to estimate each $\theta_{f,e}$ and ψ_d. To faciliate the discussion, we start by assuming that we have N pairs of training sentences $\mathcal{D} = \left\{\left(\mathbf{f}^{(n)}, \mathbf{e}^{(n)}\right)\right\}$ for each $n = 0, 1, ..., N-1$. The training process aims to maximize the likelihood on $\mathcal{D}$

$$\mathcal{L}(\theta, \psi) = \prod_{n=0}^{N-1} p\left(\mathbf{a}^{(n)}, \mathbf{f}^{(n)} \middle| \mathbf{e}^{(n)}; \theta, \psi\right) \tag{2.7}$$

In Eqn 2.7, only $\mathbf{f}^{(n)}$ and $\mathbf{e}^{(n)}$ are known, while the alignment vectors $\mathbf{a}^{(n)}$ are hidden. Depending on how we choose each $\mathbf{a}^{(n)}$, and of course, θ and ψ, $\mathcal{L}(\theta, \psi)$ will take different values.

The EM algorithm maximizes $\mathcal{L}(\theta, \psi)$ via an iterative process. Starting with an initial guess $\psi^{(0)}$, $\theta(0)$, it iteratively computes $\psi^{(t+1)}$, $\theta(t+1)$ based on $\psi^{(t)}$, $\theta(t)$ and the training data $\mathcal{D}$.

Next, in Section 2.2.1, we show the theoretical justification that the EM algorithm converges to a local optimum of the parameters ψ and θ.

2.2.1 Overview and Justification of the EM Algorithm

For simplicity of notations, we will denote by $(\mathbf{F}, \mathbf{E})$ the set $\mathcal{D}$ and by $\mathbf{A}$ the set of all possible alignments $\left\{\mathbf{a}^{(n)}\right\}_{n=0}^{N-1}$. Under this notation, we can write

$$\mathcal{L}(\theta, \psi) = p(\mathbf{F}|\mathbf{E}; \theta, \psi) = \sum_{\mathbf{A}} p(\mathbf{F}, \mathbf{A}|\mathbf{E}; \theta, \psi) \tag{2.8}$$

We introduce a new concept, the *complete log-likelihood*

$$Q\left(\theta, \psi, \tilde{\theta}, \tilde{\psi}\right) = \sum_{\mathbf{A}} p(\mathbf{F}, \mathbf{A}|\mathbf{E}; \theta, \psi) \cdot \log p\left(\mathbf{F}, \mathbf{A}\middle|\mathbf{E}; \tilde{\theta}, \tilde{\psi}\right) \tag{2.9}$$

We have the following crucial result regarding $Q\left(\theta, \psi, \tilde{\theta}, \tilde{\psi}\right)$.

Theorem 1. *If $Q\left(\theta, \psi, \tilde{\theta}, \tilde{\psi}\right) \geq Q\left(\theta, \psi, \theta, \psi\right)$ then $\mathcal{L}\left(\tilde{\theta}, \tilde{\psi}\right) \geq \mathcal{L}\left(\theta, \psi\right)$.*

Proof. We have

$$\log \frac{\mathcal{L}\left(\tilde{\theta}, \tilde{\psi}\right)}{\mathcal{L}\left(\theta, \psi\right)} = \log \left\{\sum_{\mathbf{A}} \frac{p(\mathbf{F}, \mathbf{A}|\mathbf{E}; \tilde{\theta}, \tilde{\psi})}{\mathcal{L}\left(\theta, \psi\right)}\right\} \tag{2.10}$$

$$= \log \left\{\sum_{\mathbf{A}} \frac{p(\mathbf{F}, \mathbf{A}|\mathbf{E}; \theta, \psi)}{\mathcal{L}\left(\theta, \psi\right)} \cdot \frac{p(\mathbf{F}, \mathbf{A}|\mathbf{E}; \tilde{\theta}, \tilde{\psi})}{p(\mathbf{F}, \mathbf{A}|\mathbf{E}; \theta, \psi)}\right\} \tag{2.11}$$

$$\geq \sum_{\mathbf{A}} \left\{\frac{p(\mathbf{F}, \mathbf{A}|\mathbf{E}; \theta, \psi)}{\mathcal{L}\left(\theta, \psi\right)} \cdot \log \frac{p(\mathbf{F}, \mathbf{A}|\mathbf{E}; \tilde{\theta}, \tilde{\psi})}{p(\mathbf{F}, \mathbf{A}|\mathbf{E}; \theta, \psi)}\right\} \tag{2.12}$$

$$= \frac{1}{\mathcal{L}\left(\theta, \psi\right)} \sum_{\mathbf{A}} \left\{p(\mathbf{F}, \mathbf{A}|\mathbf{E}; \theta, \psi) \cdot \log \frac{p(\mathbf{F}, \mathbf{A}|\mathbf{E}; \tilde{\theta}, \tilde{\psi})}{p(\mathbf{F}, \mathbf{A}|\mathbf{E}; \theta, \psi)}\right\} \tag{2.13}$$

$$= \frac{1}{\mathcal{L}\left(\theta, \psi\right)} \left(Q\left(\theta, \psi, \tilde{\theta}, \tilde{\psi}\right) - Q\left(\theta, \psi, \theta, \psi\right)\right) \tag{2.14}$$

In the above derivation, the inequality in Eqn 2.12 follows from Jensen's inequality, the concavity of the function $\log(\cdot)$, and the observation that the terms

$$\frac{p(\mathbf{F}, \mathbf{A}|\mathbf{E}; \theta, \psi)}{\mathcal{L}\left(\theta, \psi\right)} = \frac{p(\mathbf{F}, \mathbf{A}|\mathbf{E}; \theta, \psi)}{\sum_{\tilde{\mathbf{A}}} p(\mathbf{F}, \tilde{\mathbf{A}}|\mathbf{E}; \theta, \psi)} \tag{2.15}$$

constitute a (conditional) probability distribution over $(\mathbf{F}, \mathbf{A}|\mathbf{E})$. Finally, note that from Eqn 2.14, if $Q\left(\theta, \psi, \tilde{\theta}, \tilde{\psi}\right) \geq Q\left(\theta, \psi, \theta, \psi\right)$ then since $\mathcal{L}\left(\theta, \psi\right) \geq 0$, as it is a product of probabilities, we have

$$\log \frac{\mathcal{L}\left(\tilde{\theta}, \tilde{\psi}\right)}{\mathcal{L}\left(\theta, \psi\right)} \geq 0, \tag{2.16}$$

which proves the theorem. □

Under Theorem 1, the EM update at step t^{th} attempts to find the new parameters

$$\theta^{(t+1)}, \psi^{(t+1)} \stackrel{\text{def}}{=} \underset{\theta,\psi}{\operatorname{argmax}}\, Q\left(\theta^{(t)}, \psi^{(t)}, \theta, \psi\right) \tag{2.17}$$

Fortunately, Eqn 2.17 has a closed form solution that can be computed efficiently. This is *not* the case for many other optimization problems. Next, in Section 2.2.2, we derive its solution.

2.2.2 Updates of the EM Algorithm

We compute $\theta^{(t+1)}$ and $\psi^{(t+1)}$ in Eqn 2.17 by setting their corresponding derivatives with respect to the Lagrangian to 0. For simplicity, we will not explicitly write down the Lagrangian, but instead, will mention the Lagrange multipliers whenever they are used. We first expand $p\left(\mathbf{F}, \mathbf{A}|\mathbf{E}; \theta, \psi\right)$ to its parametrized form

$$\log p\left(\mathbf{F}, \mathbf{A}|\mathbf{E}; \theta, \psi\right) = \log\left\{\prod_{n=0}^{N-1} p\left(\mathbf{f}^{(n)}, \mathbf{a}^{(n)}\middle|\mathbf{e}^{(n)}; \theta, \psi\right)\right\} \tag{2.18}$$

$$= \sum_{n=0}^{N-1} \log p\left(\mathbf{f}^{(n)}, \mathbf{a}^{(n)}\middle|\mathbf{e}^{(n)}; \theta, \psi\right) \tag{2.19}$$

$$= \sum_{n=0}^{N-1}\left\{\log p\left(a_0\middle|\mathbf{e}^{(n)}\right) + \sum_{j=1}^{J^{(n)}-1} \log p\left(a_j\middle|a_{j-1}, \mathbf{e}^{(n)}\right) + \sum_{j=0}^{J^{(n)}-1} \log \theta_{f_j, e_{a_j}}\right\} \tag{2.20}$$

Thus, $\log p\left(\mathbf{F}, \mathbf{A}|\mathbf{E}; \theta, \psi\right)$ is summation over N terms. Correspondingly, $p\left(\mathbf{F}, \mathbf{A}|\mathbf{E}; \theta, \psi\right)$ factors over the same N terms. If for each $n = 0, 1, ..., N-1$, we take $\log p\left(\mathbf{f}^{(n)}, \mathbf{a}^{(n)}\middle|\mathbf{e}^{(n)}; \theta, \psi\right)$ and marginalize the terms in $p\left(\mathbf{F}, \mathbf{A}|\mathbf{E}; \theta, \psi\right)$ for which there is no n, then they sum up to 1. Hence

$$Q\left(\theta^{(t)}, \psi^{(t)}, \theta, \psi\right) = \sum_{n=0}^{N-1}\left\{\sum_{\mathbf{a}^{(n)}} p\left(\mathbf{f}^{(n)}, \mathbf{a}^{(n)}\middle|\mathbf{e}^{(n)}; \theta^{(t)}, \psi^{(t)}\right) \cdot \log p\left(\mathbf{f}^{(n)}, \mathbf{a}^{(n)}\middle|\mathbf{e}^{(n)}; \theta, \psi\right)\right\} \tag{2.21}$$

Deriving $\theta_{f,e}^{(t+1)}$. We now compute the partial derivative

$$\frac{\partial Q\left(\theta^{(t)}, \psi^{(t)}, \theta, \psi\right)}{\partial \theta} = \sum_{n=0}^{N-1}\left\{\sum_{\mathbf{a}^{(n)}} p\left(\mathbf{f}^{(n)}, \mathbf{a}^{(n)}\middle|\mathbf{e}^{(n)}; \theta^{(t)}, \psi^{(t)}\right) \cdot \frac{\partial \log p\left(\mathbf{f}^{(n)}, \mathbf{a}^{(n)}\middle|\mathbf{e}^{(n)}; \theta, \psi\right)}{\partial \theta}\right\} \tag{2.22}$$

Comparing Eqn 2.20 and Eqn 2.22, we see that for each French word f and each English word e (including the null word), $\theta_{f,e}$ appears in the terms $\log p\left(\mathbf{f}^{(n)}, \mathbf{a}^{(n)}\middle|\mathbf{e}^{(n)}; \theta, \psi\right)$ for which the followings are all true

1. A word f_j in $\mathbf{f}^{(n)}$ is f,
2. A word e_i in $\mathbf{e}^{(n)}$ is e,
3. The alignment vector $\mathbf{a}^{(n)}$ says that $a_j = i$.

We can express these conditions using the indicator function $\mathbb{1}[\cdot]$

$$\frac{\partial Q\left(\theta^{(t)}, \psi^{(t)}, \theta, \psi\right)}{\partial \theta_{f,e}} = \frac{1}{\theta_{f,e}} \cdot \sum_{n=0}^{N-1}\left\{\sum_{\mathbf{a}^{(n)}} p\left(\mathbf{f}^{(n)}, \mathbf{a}^{(n)}\middle|\mathbf{e}^{(n)}; \theta^{(t)}, \psi^{(t)}\right) \cdot \sum_{i=0}^{I^{(n)}-1} \sum_{j=0}^{J^{(n)}-1} \mathbb{1}\left[f_j = f\right] \cdot \mathbb{1}\left[e_i = e\right] \cdot \mathbb{1}\left[a_j = i\right]\right\} \tag{2.23}$$

$$= \frac{1}{\theta_{f,e}} \cdot \sum_{n=0}^{N-1} \sum_{i=0}^{I^{(n)}-1} \sum_{j=0}^{J^{(n)}-1} \mathbb{1}\left[f_j = f\right] \cdot \mathbb{1}\left[e_i = e\right] \cdot \left\{ \sum_{\mathbf{a}^{(n)}} p\left(\mathbf{f}^{(n)}, \mathbf{a}^{(n)} \middle| \mathbf{e}^{(n)}; \theta^{(t)}, \psi^{(t)}\right) \cdot \mathbb{1}\left[a_j = i\right] \right\} \tag{2.24}$$

$$= \frac{1}{\theta_{f,e}} \cdot \sum_{n=0}^{N-1} \sum_{i=0}^{I^{(n)}-1} \sum_{j=0}^{J^{(n)}-1} \mathbb{1}\left[f_j = f\right] \cdot \mathbb{1}\left[e_i = e\right] \cdot p\left(\mathbf{f}^{(n)}, \mathbf{a}_j^{(n)} = i \middle| \mathbf{e}^{(n)}; \theta^{(t)}, \psi^{(t)}\right) \tag{2.25}$$

Note that for each English word e, we have a condition

$$\sum_f \theta_{f,e} = 1, \tag{2.26}$$

so we introduce a Largrange multiplier λ_e to $Q\left(\theta^{(t)}, \psi^{(t)}, \theta, \psi\right)$, which will lead to the normalization of $\theta_{f,e}$ long each e. We thus have the first update rule

$$\boxed{\theta_{f,e}^{(t+1)} = \frac{1}{\lambda_e} \cdot \sum_{n=0}^{N-1} \sum_{i=0}^{I^{(n)}-1} \sum_{j=0}^{J^{(n)}-1} \mathbb{1}\left[f_j = f\right] \cdot \mathbb{1}\left[e_i = e\right] \cdot p\left(\mathbf{f}^{(n)}, \mathbf{a}_j^{(n)} = i \middle| \mathbf{e}^{(n)}; \theta^{(t)}, \psi^{(t)}\right)} \tag{2.27}$$

To realize Eqn 2.27, we only need to compute $p\left(\mathbf{f}, \mathbf{a}_j = i \middle| \mathbf{e}; \theta^{(t)}, \psi^{(t)}\right)$. For this, we resort to the α and β forward-backward probabilities. Thanks to the independence assumption of HMM, we can write

$$p\left(\mathbf{f}, \mathbf{a}_j = i \middle| \mathbf{e}; \psi^{(t)}\right) = p\left(\mathbf{f}, \mathbf{a}_0, ..., \mathbf{a}_{j-1}, \mathbf{a}_j = i, \mathbf{a}_{j+1}, ..., \mathbf{a}_{J-1} \middle| \mathbf{e}; \theta^{(t)}, \psi^{(t)}\right) \tag{2.28}$$

$$= \underbrace{p\left(\mathbf{f}, \mathbf{a}_0, ..., \mathbf{a}_{j-1}, \mathbf{a}_j = i \middle| \mathbf{e}; \psi^{(t)}\right)}_{\alpha_j^i} \cdot \underbrace{p\left(\mathbf{f}, \mathbf{a}_{j+1}, ..., \mathbf{a}_{J-1} \middle| \mathbf{a}_j = i, \mathbf{a}_{j-1}, ..., \mathbf{a}_0, \mathbf{e}; \theta^{(t)}, \psi^{(t)}\right)}_{\beta_j^i} \tag{2.29}$$

We can compute α_j^i efficiently using dynamic programming

$$\alpha_j^i = \theta_{\mathbf{f}_j, \mathbf{e}_i}^{(t)} \cdot \sum_{i'=0}^{I-1} \alpha_{j-1}^{i'} \cdot p\left(\mathbf{a}_j = i \middle| \mathbf{a}_{j-1} = i', \mathbf{e}; \theta^{(t)}, \psi^{(t)}\right) \qquad p\left(\mathbf{a}_j \middle| \mathbf{a}_{j-1}, \mathbf{e}; \theta^{(t)}, \psi^{(t)}\right) \text{ come from Eqn 2.4} \tag{2.30}$$

$$\alpha_0^i = \theta_{\mathbf{f}_0, \mathbf{e}_i}^{(t)} \cdot p\left(\mathbf{a}_0 = i \middle| \mathbf{e}; \theta^{(t)}, \psi^{(t)}\right) \qquad p\left(\mathbf{a}_0 \middle| \mathbf{e}; \theta^{(t)}, \psi^{(t)}\right) \text{ come from Eqn 2.6} \tag{2.31}$$

Same for β_j^i, but trickier

$$\beta_j^i = p\left(\mathbf{f}, \mathbf{a}_{j+1}, ..., \mathbf{a}_{J-1} \middle| \mathbf{a}_j = i, ..., \mathbf{a}_0, \mathbf{e}; \theta^{(t)}, \psi^{(t)}\right) \tag{2.32}$$

$$= \sum_{i'=0}^{I-1} p\left(\mathbf{f}, \mathbf{a}_{j+1} = i' \middle| \mathbf{a}_j = i, ..., \mathbf{a}_0, \mathbf{e}; \theta^{(t)}, \psi^{(t)}\right) \cdot p\left(\mathbf{f}, \mathbf{a}_{j+2}, ..., \mathbf{a}_{J-1} \middle| \mathbf{a}_{j+1} = i', \mathbf{a}_j = i, ..., \mathbf{a}_0, \mathbf{e}; \theta^{(t)}, \psi^{(t)}\right) \tag{2.33}$$

$$= \sum_{i'=0}^{I-1} p\left(\mathbf{f}, \mathbf{a}_{j+1} = i' \middle| \mathbf{a}_j = i, \mathbf{e}; \theta^{(t)}, \psi^{(t)}\right) \cdot p\left(\mathbf{f}, \mathbf{a}_{j+2}, ..., \mathbf{a}_{J-1} \middle| \mathbf{a}_{j+1} = i', \mathbf{a}_j, ..., \mathbf{a}_0, \mathbf{e}; \theta^{(t)}, \psi^{(t)}\right) \tag{2.34}$$

$$= \sum_{i'=0}^{I-1} \theta_{\mathbf{f}_j, \mathbf{e}_{i'}}^{(t)} \cdot p\left(\mathbf{a}_{j+1} = i' \middle| \mathbf{a}_j = i, \mathbf{e}; \theta^{(t)}, \psi^{(t)}\right) \cdot \beta_{j+1}^{i'} \tag{2.35}$$

$$\beta_{J-1}^i = p(\text{nothing}|\mathbf{a}_{J-1} = i, \mathbf{e}; \theta^{(t)}, \psi^{(t)}) = 1.0 \tag{2.36}$$

Eqn 2.33 is due to Bayes rule. Eqn 2.34 is derived because the HMM assumptions allow us to drop the dependency of everything but intermediate time steps. Lastly, the probabilities required in Eqn 2.35 come from Eqn 2.4.

Deriving $\psi_d^{(t+1)}$. We first modify Eqn 2.21 into its Lagrangian. First, note that Eqn 2.4 and 2.6 essentially say that

$$p\left(\mathbf{a}_j | \mathbf{a}_{j-1}, \mathbf{e}; \theta, \psi\right) \propto \psi_{|\mathbf{a}_j - \mathbf{a}_{j-1}|} \text{ and } p\left(\mathbf{a}_0 | \mathbf{e}; \theta, \psi\right) \propto \psi_{|\mathbf{a}_0|} \tag{2.37}$$

In order to relax the denominators Z_j in Eqn 2.4 and 2.6, for each $n = 0, 1, ..., N-1$, and for each $j = 0, 1, ..., J^{(n)}-1$, we introduce a Lagrange multiplier $Z_j^{(n)}$ that constraints the $p(\cdot|\cdot, \mathbf{e}; \theta, \psi)$ to 1. These Lagrange multipliers allow us to compute $\partial Q/\partial\psi_d$ without having to deal with the denominators, provided that we normalize each corresponding values, as we did to $\theta_{f,e}$. We thus have

$$
\begin{aligned}
\frac{\partial Q\left(\theta^{(t)}, \psi^{(t)}, \theta, \psi\right)}{\partial\psi_d} &= \sum_{n=0}^{N-1}\left\{\sum_{\mathbf{a}^{(n)}} p\left(\mathbf{f}^{(n)}, \mathbf{a}^{(n)}\middle|\mathbf{e}^{(n)}; \theta^{(t)}, \psi^{(t)}\right)\cdot\frac{\partial\log p\left(\mathbf{f}^{(n)}, \mathbf{a}^{(n)}|\mathbf{e}^{(n)}; \theta, \psi\right)}{\partial\psi_d}\right\} \qquad (2.38)\\
&= \frac{1}{\psi_d}\sum_{n=0}^{N-1}\sum_{\mathbf{a}^{(n)}} p\left(\mathbf{f}^{(n)}, \mathbf{a}^{(n)}\middle|\mathbf{e}^{(n)}; \theta^{(t)}, \psi^{(t)}\right)\cdot\left\{\mathbb{1}\left[\mathbf{a}_0^{(n)} = d\right] + \sum_{j=1}^{J^{(n)}-1}\mathbb{1}\left[|\mathbf{a}_j - \mathbf{a}_{j-1}| = d\right]\right\} \qquad (2.39)\\
&= \frac{1}{\psi_d}\sum_{n=0}^{N-1}\left\{p\left(\mathbf{f}^{(n)}, \mathbf{a}_0^{(n)} = d\middle|\mathbf{e}^{(n)}; \theta^{(t)}, \psi^{(t)}\right) + \sum_{j=1}^{J^{(n)}-1} p\left(\mathbf{f}^{(n)}, |\mathbf{a}_j^{(n)} - \mathbf{a}_{j-1}^{(n)}| = d\middle|\mathbf{e}^{(n)}; \theta^{(t)}, \psi^{(t)}\right)\right\} \qquad (2.40)
\end{aligned}
$$

As done with $\theta_{f,e}$, we can compute the probabilities in Eqn 2.40 using the α and β values. Specifically, note that α, and β a specific to each pair of sentences $\left(\mathbf{f}^{(n)}, \mathbf{e}^{(n)}\right)$, we can write

$$
p\left(\mathbf{f}^{(n)}, \mathbf{a}_0^{(n)} = d\middle|\mathbf{e}^{(n)}; \theta^{(t)}, \psi^{(t)}\right) \propto \left\{\alpha_0^d\right\}^{(n)} \qquad (2.41)
$$

$$
\begin{aligned}
p\left(\mathbf{f}^{(n)}, |\mathbf{a}_j^{(n)} - \mathbf{a}_{j-1}^{(n)}| = d\middle|\mathbf{e}^{(n)}; \theta^{(t)}, \psi^{(t)}\right) &\propto \sum_{i=0}^{I-d-1} p\left(\mathbf{f}^{(n)}, \mathbf{a}_j^{(n)} = i, \mathbf{a}_{j-1}^{(n)} = i+d\middle|\mathbf{e}^{(n)}; \theta^{(t)}, \psi^{(t)}\right)\\
&\quad + \sum_{i=0}^{I-d-1} p\left(\mathbf{f}^{(n)}, \mathbf{a}_j^{(n)} = i+d, \mathbf{a}_{j-1}^{(n)} = i\middle|\mathbf{e}^{(n)}; \theta^{(t)}, \psi^{(t)}\right) \qquad (2.42)\\
&= \sum_{i=0}^{I-d-1}\left\{\alpha_{j-1}^{i+d}\right\}^{(n)}\cdot\left\{\beta_j^i\right\}^{(n)}\cdot p\left(\mathbf{f}^{(n)}, \mathbf{a}_j^{(n)} = i+d\middle|\mathbf{a}_{j-1}^{(n)} = i, \mathbf{e}^{(n)}; \theta^{(t)}, \psi^{(t)}\right)\\
&\quad + \left\{\alpha_{j-1}^i\right\}^{(n)}\cdot\left\{\beta_j^{i+d}\right\}^{(n)}\cdot p\left(\mathbf{f}^{(n)}, \mathbf{a}_j^{(n)} = i\middle|\mathbf{a}_{j-1}^{(n)} = i+d, \mathbf{e}^{(n)}; \theta^{(t)}, \psi^{(t)}\right) \qquad (2.43)
\end{aligned}
$$

Finally, the Largrange multipliers $Z_j^{(n)}$ simply normalize $p\left(\mathbf{f}^{(n)}, |\mathbf{a}_j^{(n)} - \mathbf{a}_{j-1}^{(n)}| = d\middle|\cdot\right)$ and $p\left(\mathbf{f}^{(n)}, \mathbf{a}_0^{(n)} = d\middle|\cdot\right)$ accross all d. We thus have the following update rule for ψ_d

$$
\boxed{\psi_d^{(t+1)} = \sum_{n=0}^{N-1}\left\{\sum_{j=0}^{J^{(n)}-1}\frac{\tilde{p}_{j,d}^{(n)}}{Z_j^{(n)}}\right\}}, \qquad (2.44)
$$

where

$$
\begin{aligned}
\tilde{p}_{j,d}^{(n)} &= \sum_{i=0}^{I-d-1}\left\{\alpha_{j-1}^{i+d}\right\}^{(n)}\cdot\left\{\beta_j^i\right\}^{(n)}\cdot p\left(\mathbf{f}^{(n)}, \mathbf{a}_j^{(n)} = i+d\middle|\mathbf{a}_{j-1}^{(n)} = i, \mathbf{e}^{(n)}; \theta^{(t)}, \psi^{(t)}\right)\\
&\quad + \left\{\alpha_{j-1}^i\right\}^{(n)}\cdot\left\{\beta_j^{i+d}\right\}^{(n)}\cdot p\left(\mathbf{f}^{(n)}, \mathbf{a}_j^{(n)} = i\middle|\mathbf{a}_{j-1}^{(n)} = i+d, \mathbf{e}^{(n)}; \theta^{(t)}, \psi^{(t)}\right) \qquad (2.45)\\
Z_j^{(n)} &= \sum_{d=0}^{I^{(n)}-1}\tilde{p}_{j,d}^{(n)} \qquad (2.46)
\end{aligned}
$$

2.3 Put Together and Pseudocode

We are now ready to present the pseudo code for EM algorithm for HMM in Algorithm 1.

Algorithm 1 Training HMM with EM.

```
procedure TrainHMMWithEM($\mathcal{D} = \{(\mathbf{f}^{(n)}, \mathbf{e}^{(n)})_{n=0}^{N-1}\}$, $\varepsilon$)
    Initialize $\theta^{(0)}$, $\psi^{(0)}$
    for $t = 0$ to $T - 1$ do
        Set $\theta_{f,e}^{(t+1)} \leftarrow 0$ for all $f$, $e$
        Set $\psi_d^{(t+1)} \leftarrow 0$ for all $d$
        for $n = 0$ to $N - 1$ do
            Let $\mathbf{f} = \mathbf{f}^{(n)}$, $\mathbf{e} = \mathbf{e}^{(n)}$, $I = |\mathbf{e}|$, $J = |\mathbf{f}|$
            Compute $p\left(i' \middle| i, \mathbf{e}; \psi^{(t)}\right)$ and $p\left(a_0 = i \middle| \mathbf{e}; \psi^{(t)}\right)$ using Eqn 2.4 and Eqn 2.6.
            Compute $\alpha_j^i$ using Eqn 2.30 and Eqn 2.31.
            Compute $\beta_j^i$ using Eqn 2.35 and Eqn 2.36.
            Compute $\tilde{p}_{j,d}$ and $Z_j$ using Eqn 2.45 and Eqn 2.46.
            for $j = 0$ to $J - 1$ do
                Update $\psi_d^{(t+1)} \leftarrow \psi_d^{(t+1)} + \frac{\tilde{p}_{j,d}}{Z_j}$.
            end for
            for $i = 0$ to $I - 1$ do
                for $j = 0$ to $J - 1$ do
                    Update $\theta_{\mathbf{f}_j, \mathbf{e}_i}^{(t+1)} \leftarrow \theta_{\mathbf{f}_j, \mathbf{e}_i}^{(t+1)} + \alpha_j^i \cdot \beta_j^i$.
                end for
            end for
        end for
        Normalize $\theta_{f,e}^{(t+1)} \leftarrow \dfrac{\theta_{f,e}^{(t+1)}}{\sum_{f'} \theta_{f',e}^{(t+1)}}$ for each $e$.
    end for
    return $\theta^{(T)}$, $\psi^{(T)}$
end procedure
```

3 IBM Model 1

All the derivations of HMM still work, if you replace all the transition probabilities with the uniform distribution $\frac{1}{I^{(n)}+1}$ for each $\left(\mathbf{f}^{(n)}, \mathbf{e}^{(n)}\right)$.

References

Leonard E. Baum, Ted Petrie, George Soules, and Norman Weiss. A maximization technique occurring in the statistical analysis of probabilistic. In *The Annals of Mathematical Statistics*, 1980.

A. J. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. In *IEEE Transactions on Information Theory*, 1967.