# Learning Hidden Markov Models from Pairwise Co-occurrences with Application to Topic Modeling

*Supplementary Material*

Kejun Huang
University of Minnesota
Minneapolis, MN 55455
huang663@umn.edu

Xiao Fu
Oregon State University
Corvallis, OR 97331
xiao.fu@oregonstate.edu

Nicholas D. Sidiropoulos
University of Virginia
Charlottesville, VA 22904
nikos@virginia.edu

## 1 Proof of Proposition 1

For categorical probabilities $\boldsymbol{p}$ and $\boldsymbol{q}$, their Kullback-Leiber divergence is defined as

$$D_{\text{KL}}(\boldsymbol{p}\|\boldsymbol{q}) = \sum_{n=1}^{N} p_n \log \frac{p_n}{q_n},$$

and their total variation distance is defined as

$$D_{\text{TV}}(\boldsymbol{p}\|\boldsymbol{q}) = \frac{1}{2}\sum_{n=1}^{N} |p_n - q_n|.$$

The key to prove Proposition 1 is the fact that the cooccurrence probability $\boldsymbol{\Omega}$ can be obtained by marginalizing $X_{t-1}$ in the triple-occurrence probability $\underline{\boldsymbol{\Omega}_3}$, i.e.,

$$\boldsymbol{\Omega}(i,j) = \sum_{n=1}^{N} \underline{\boldsymbol{\Omega}_3}(n,i,j).$$

Similarly, this holds for the cumulative estimates described in §2 of the main paper as well,

$$\widehat{\boldsymbol{\Omega}}(i,j) = \sum_{n=1}^{N} \underline{\widehat{\boldsymbol{\Omega}_3}}(n,i,j).$$

Using the log sum inequality, we have that

$$\boldsymbol{\Omega}(i,j) \log \frac{\boldsymbol{\Omega}(i,j)}{\widehat{\boldsymbol{\Omega}}(i,j)} \le \sum_{n=1}^{N} \underline{\boldsymbol{\Omega}_3}(n,i,j) \log \frac{\underline{\boldsymbol{\Omega}_3}(n,i,j)}{\underline{\widehat{\boldsymbol{\Omega}_3}}(n,i,j)}.$$

Summing both sides over $i$ and $j$, we result in

$$D_{\text{KL}}(\widehat{\boldsymbol{\Omega}}\|\boldsymbol{\Omega}) \le D_{\text{KL}}(\underline{\widehat{\boldsymbol{\Omega}_3}}\|\underline{\boldsymbol{\Omega}_3})$$

Using Hölder's inequality with $L_1$-norm and $L_\infty$-norm, we have that

$$|\boldsymbol{\Omega}(i,j) - \widehat{\boldsymbol{\Omega}}(i,j)| \le \sum_{n=1}^{N} |\underline{\boldsymbol{\Omega}_3}(n,i,j) - \underline{\widehat{\boldsymbol{\Omega}_3}}(n,i,j)|.$$

Summing both sides over $i$ and $j$ and then dividing by 2, we obtain

$$D_{\text{TV}}(\widehat{\boldsymbol{\Omega}}\|\boldsymbol{\Omega}) \le D_{\text{TV}}(\underline{\widehat{\boldsymbol{\Omega}_3}}\|\underline{\boldsymbol{\Omega}_3})$$

**Q.E.D.**

# 2 Proof of Proposition 2

The volume of a hyper-ball in $\mathbb{R}^n$ with radius $R$ is

$$\frac{\pi^{\frac{n}{2}}}{\Gamma(\frac{n}{2}+1)} R^n.$$

The elliptical cone $\mathcal{C}^* = \{\boldsymbol{x} : \|\boldsymbol{x}\| \leq \boldsymbol{1}^\top \boldsymbol{x}/\sqrt{K-1}\}$ intersecting with the hyperplane $\boldsymbol{1}^\top \boldsymbol{x} = 1$ is a hyperball in $\mathbb{R}^{K-1}$ with radius $\sqrt{\frac{1}{K(K-1)}}$. Therefore, the volume of the inner-ball is

$$V_b = \frac{\pi^{\frac{K-1}{2}}}{\Gamma(\frac{K+1}{2})} (K(K-1))^{-\frac{K-1}{2}}.$$

The nonnegative orthan intersecting with $\boldsymbol{1}^\top \boldsymbol{x} = 1$ is a regular simplex in $\mathbb{R}^{K-1}$ with side length $\sqrt{2}$. Its volume is

$$V_s = \frac{\sqrt{K}}{(K-1)!} = \frac{\sqrt{K}}{\Gamma(K)}.$$

Their ratio is

$$\begin{aligned}
\frac{V_b}{V_s} &= \frac{\frac{\pi^{\frac{K-1}{2}}}{\Gamma(\frac{K+1}{2})} (K(K-1))^{-\frac{K-1}{2}}}{\frac{\sqrt{K}}{\Gamma(K)}} \\
&= \frac{1}{\sqrt{K}} \left(\frac{\pi}{K(K-1)}\right)^{\frac{K-1}{2}} \frac{\Gamma(K)}{\Gamma(\frac{K+1}{2})} \\
&= \frac{1}{\sqrt{K}} \left(\frac{\pi}{K(K-1)}\right)^{\frac{K-1}{2}} \frac{\Gamma(\frac{K}{2})}{2^{1-K}\sqrt{\pi}} \\
&= \frac{1}{\sqrt{\pi K}} \left(\frac{4\pi}{K(K-1)}\right)^{\frac{K-1}{2}} \Gamma\left(\frac{K}{2}\right)
\end{aligned}$$

**Q.E.D.**

This function of volume ratio is plotted in Figure 1. As we can see, as $K$ increases, the volume ratio indeed goes to zero at a super-exponential rate.

# 3 Derivation of THETAUPDATE

It is described in [Boyd and Vandenberghe, 2004, §10.2] that for solving a convex equality constrained problem

$$\begin{aligned}
&\underset{x}{\text{minimize}} \;\; f(x) \\
&\text{subject to } Ax = b
\end{aligned}$$

using the Newton's method, we start at a feasible point $x$, and the iterative update takes the form $x \leftarrow x - \alpha \Delta_{\text{nt}} x$, where the Newton direction is calculated from solving the KKT system

$$\begin{bmatrix} \nabla^2 f(x) & A^\top \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta_{\text{nt}} x \\ d \end{bmatrix} = \begin{bmatrix} -\nabla f(x) \\ 0 \end{bmatrix}.$$

Assuming $\nabla^2 f(x) \succ 0$ and $A$ has full row rank, then the KKT system can be solved via elimination, as described in [Boyd and Vandenberghe, 2004, Algorithm 10.3]. Suppose $A \in \mathbb{R}^{m \times n}$, if $\nabla^2 f(x)$ is diagonal, the cost of calculating $\Delta_{\text{nt}} x$ is dominated by forming and inverting the matrix $ADA^\top$ with $D$ being diagonal.

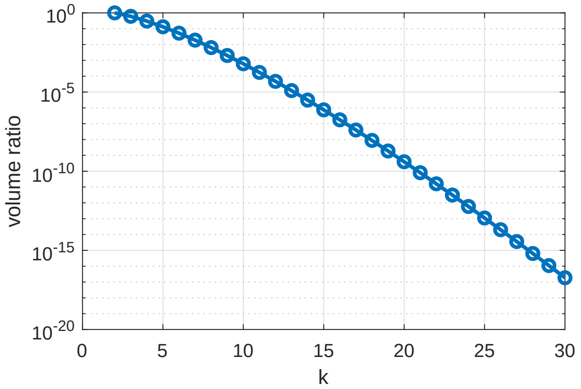

Figure 1: The volume ratio between the hyperball obtained by intersecting $\mathcal{C}$ and the hyperplane $\boldsymbol{1}^\top \boldsymbol{x} = 1$ and the probability simplex, as $K$ increases.

Now we follow the steps of [Boyd and Vandenberghe, 2004, Algorithm 10.3] to derive explicit Newton iterates for solving (11). First, we re-write the part of (11) that involve $\boldsymbol{\Theta}$ here:

$$\begin{aligned} \underset{\boldsymbol{\Theta} > 0}{\text{minimize}} \quad & \sum_{n,\ell=1}^{N} \sum_{k,j=1}^{K} -\Omega_{n\ell} \Pi^r_{n\ell kj} \log \Theta_{kj} + \lambda \sum_{k,j=1}^{K} \Xi^r_{kj} \Theta_{kj} \\ \text{subject to} \quad & \boldsymbol{1}^\top \boldsymbol{\Theta} \boldsymbol{1} = 1, \boldsymbol{\Theta} \boldsymbol{1} = \boldsymbol{\Theta}^\top \boldsymbol{1}. \end{aligned}$$

Let $\boldsymbol{\theta} = \operatorname{vec}(\boldsymbol{\Theta})$, then equality constraint has the form $\boldsymbol{A}\boldsymbol{\theta} = \boldsymbol{b}$ where

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{1}^\top \otimes \boldsymbol{1}^\top \\ \boldsymbol{1}^\top \otimes \boldsymbol{I} - \boldsymbol{I} \otimes \boldsymbol{1}^\top \end{bmatrix}.$$

Matrix $\boldsymbol{A}$ does not have full row rank, because the last row of $\boldsymbol{A}$ is implied by the rest. Therefore, we can discard the last equality constraint. We will keep it when calculating matrix multiplications for simpler expression, and discard the corresponding entry or column/row for other operations.

Obviously $\boldsymbol{A}\boldsymbol{\theta}$ has the form

$$\boldsymbol{A}\boldsymbol{\theta} = \begin{bmatrix} \boldsymbol{1}^\top \boldsymbol{\Theta} \boldsymbol{1} \\ \boldsymbol{\Theta}\boldsymbol{1} - \boldsymbol{\Theta}^\top \boldsymbol{1} \end{bmatrix},$$

which costs $\mathcal{O}(K^2)$ flops. For a slightly more complicated multiplication

$$\boldsymbol{A} \operatorname{Diag}(\boldsymbol{\theta}) \boldsymbol{A}^\top = \begin{bmatrix} \boldsymbol{1}^\top \boldsymbol{\Theta} \boldsymbol{1} & \boldsymbol{1}^\top \boldsymbol{R}^\top - \boldsymbol{1}^\top \boldsymbol{\Theta} \\ \boldsymbol{\Theta}\boldsymbol{1} - \boldsymbol{\Theta}^\top \boldsymbol{1} & \operatorname{Diag}(\boldsymbol{\Theta}\boldsymbol{1} + \boldsymbol{\Theta}^\top \boldsymbol{1}) - \boldsymbol{\Theta} - \boldsymbol{\Theta}^\top \end{bmatrix},$$

which also costs $\mathcal{O}(K^2)$ flops to compute. For $[\, d_0 \ \boldsymbol{d}^\top \,]^\top \in \mathbb{R}^{K+1}$,

$$\boldsymbol{A}^\top [\, d_0 \ \boldsymbol{d}^\top \,]^\top = \operatorname{vec}\left( d_0 \boldsymbol{1}\boldsymbol{1}^\top + \boldsymbol{d}\boldsymbol{1}^\top - \boldsymbol{1}\boldsymbol{d}^\top \right).$$

At point $\boldsymbol{\theta}$, the negative gradient is $-\nabla f(\boldsymbol{\theta}) = \operatorname{vec}(\boldsymbol{G})$ where

$$G_{kj} = \frac{\sum_{n,\ell=1}^{N} \Omega_{n\ell} \Pi^r_{n\ell kj}}{\Theta_{kj}} - \lambda \Xi^r_{kj},$$

and the inverse of the Hessian $\left(\nabla^2 f(\boldsymbol{\theta})\right)^{-1} = \operatorname{Diag}(\operatorname{vec}(\boldsymbol{R}))$ where

$$R_{kj} = \frac{\Theta^2_{kj}}{\sum_{n,\ell=1}^{N} \Omega_{n\ell} \Pi^r_{n\ell kj}}.$$

Let

$$\boldsymbol{H} = \begin{bmatrix} \boldsymbol{1}^\top \boldsymbol{R}\boldsymbol{1} & \boldsymbol{1}^\top \boldsymbol{R}^\top - \boldsymbol{1}^\top \boldsymbol{R} \\ \boldsymbol{R}\boldsymbol{1} - \boldsymbol{R}^\top \boldsymbol{1} & \operatorname{Diag}(\boldsymbol{R}\boldsymbol{1} + \boldsymbol{R}^\top \boldsymbol{1}) - \boldsymbol{R} - \boldsymbol{R}^\top \end{bmatrix}$$

and then delete the last column and row of $\boldsymbol{H}$, and

$$S_{kj} = R_{kj} G_{kj}$$

$$\boldsymbol{g} = \begin{bmatrix} \boldsymbol{1}^\top \boldsymbol{S}\boldsymbol{1} \\ \boldsymbol{S}\boldsymbol{1} - \boldsymbol{S}^\top \boldsymbol{1} \end{bmatrix}$$

and then delete the last entry of $\boldsymbol{g}$. We can first solve for $\boldsymbol{d}$ by

$$\boldsymbol{d} = \boldsymbol{H}^{-1}\boldsymbol{g} = [\, d_0 \; \widetilde{\boldsymbol{d}}^\top \,]^\top.$$

Then we append a zero at the end of $\boldsymbol{d}$ and define

$$[\, \boldsymbol{d}^\top \; 0 \,]^\top \rightarrow \boldsymbol{d} = [\, d_0 \; \widetilde{\boldsymbol{d}}^\top \,]^\top.$$

The Newton direction $\Delta_{\text{nt}}\boldsymbol{\theta}$ can then be obtained via

$$\Delta_{\text{nt}}\boldsymbol{\theta} = \left(\nabla^2 f(\boldsymbol{\theta})\right)^{-1} \left(\boldsymbol{A}^\top \boldsymbol{d} + \nabla f(\boldsymbol{\theta})\right).$$

In matrix form, it is equivalent to

$$\Delta_{\text{nt}}\boldsymbol{\Theta} = \boldsymbol{R} * \left(d_0 \boldsymbol{1}\boldsymbol{1}^\top + \widetilde{\boldsymbol{d}}\boldsymbol{1}^\top - \boldsymbol{1}\widetilde{\boldsymbol{d}}^\top - \boldsymbol{G}\right).$$

The in-line implementation of ThetaUpdate is given here.

**Algorithm 1** ThetaUpdate

```
Require: $\boldsymbol{\Theta}, \widetilde{\boldsymbol{\Theta}}, \lambda, \rho$
$\boldsymbol{\Xi} \leftarrow |\det \boldsymbol{\Theta}| \boldsymbol{\Theta}^{-\top}$
repeat
    $\boldsymbol{G} \leftarrow \widetilde{\boldsymbol{\Theta}} / \boldsymbol{\Theta} - \lambda \boldsymbol{\Xi}$
    $\boldsymbol{R} \leftarrow \boldsymbol{\Theta} * \boldsymbol{\Theta} / \widetilde{\boldsymbol{\Theta}}$
    $\boldsymbol{H} \leftarrow \begin{bmatrix} \boldsymbol{1}^\top \boldsymbol{R}\boldsymbol{1} & \boldsymbol{1}^\top \boldsymbol{R}^\top - \boldsymbol{1}^\top \boldsymbol{R} \\ \boldsymbol{R}\boldsymbol{1} - \boldsymbol{R}^\top \boldsymbol{1} & \operatorname{Diag}(\boldsymbol{R}\boldsymbol{1} + \boldsymbol{R}^\top \boldsymbol{1}) - \boldsymbol{R} - \boldsymbol{R}^\top \end{bmatrix}$
    delete the last column and row of $\boldsymbol{H}$
    $\boldsymbol{g} \leftarrow \begin{bmatrix} \boldsymbol{1}^\top (\boldsymbol{R} * \boldsymbol{G})\boldsymbol{1} \\ (\boldsymbol{R} * \boldsymbol{G})\boldsymbol{1} - (\boldsymbol{R} * \boldsymbol{G})^\top \boldsymbol{1} \end{bmatrix}$
    delete the last entry of $\boldsymbol{g}$
    $\boldsymbol{d} \leftarrow \boldsymbol{H}^{-1}\boldsymbol{g}$
    $[\, d_0 \; \widetilde{\boldsymbol{d}}^\top \,]^\top \leftarrow [\, \boldsymbol{d}^\top \; 0 \,]^\top$
    $\Delta_{\text{nt}}\boldsymbol{\Theta} = \boldsymbol{R} * \left(d_0 \boldsymbol{1}\boldsymbol{1}^\top + \widetilde{\boldsymbol{d}}\boldsymbol{1}^\top - \boldsymbol{1}\widetilde{\boldsymbol{d}}^\top - \boldsymbol{G}\right)$
    $\boldsymbol{\Theta} \leftarrow \boldsymbol{\Theta} - \Delta_{\text{nt}}\boldsymbol{\Theta}$
until convergence
return $\boldsymbol{\Theta}$
```

# 4 Proof of Proposition 3

The form of Algorithm 1 falls exactly into the framework of block successive convex approximation (BSCA) algorithm proposed by Razaviyayn et al. [2013] with only one block of variables. Invoking [Razaviyayn et al., 2013, Theorem 4], we have that every limit point of Algorithm 1 is a stationary point of Problem (7). Additionally, since the constraint set of Problem (7) is compact, *any* sub-sequence has a limit point, which is also a stationary point. This proves that Algorithm 1 converges to a stationary point of Problem (7). **Q.E.D.**

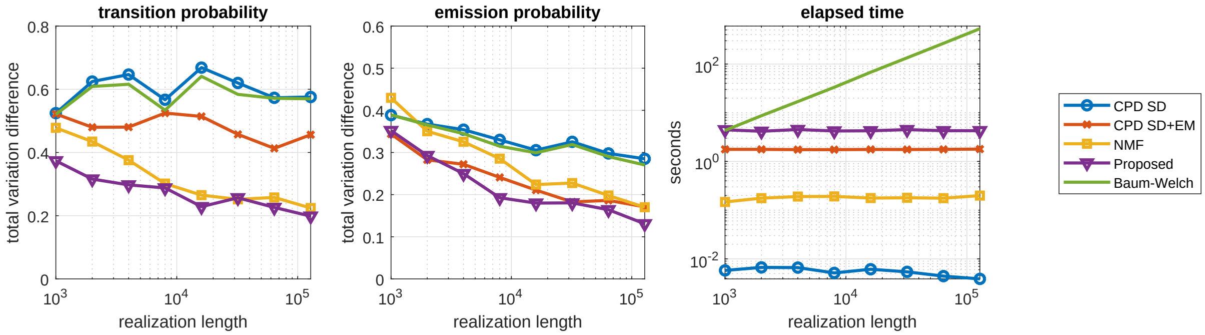

Figure 2: The total variation difference between the ground truth and estimated transition probability (left) and emission probability (middle), and the elapsed time (right) for $N = 16$ and $K = 4$. The total variation difference of the emission probabilities is calculated as $\frac{1}{2K}\|\boldsymbol{M}_\natural - \boldsymbol{M}_\star\|_1$, since each column of the matrices indicates a (conditional) probability, and the total variation difference is equal to one half of the $L_1$-norm; and similarly for that of the transition probabilities after rescaling the rows of $\boldsymbol{\Omega}_\natural$ and $\boldsymbol{\Omega}_\star$ to sum up to one. The result is averaged over 10 random problem instances.

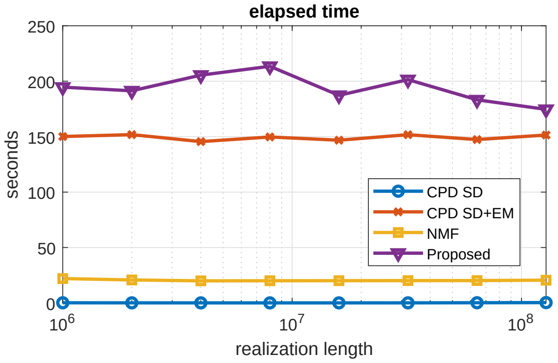

Figure 3: The elapsed time for the synthetic experiment with $N = 100$ and $K = 20$ as in the main paper.

# 5 Additional Synthetic Experiments

In this section we conduct a similar synthetic experiment to identify HMM parameters, but with a much smaller problem size, so that we can include the classical Baum-Welch algorithm [Baum et al., 1970] as another baseline. Fixing $N = 16$ and $K = 4$, the transition probabilities are synthetically generated from a random exponential matrix of size $K \times K$ followed by row-normalization; for the emission probabilities, the top $K \times K$ part of the $N \times K$ random exponential matrices are set to be the identity matrix before column normalization, so that it is guaranteed to be sufficiently scattered. We let the number of HMM realizations go from $10^3$ to $10^5$, and compare the estimation error for the transition matrix and emission matrix by the aforementioned methods. We show the total variation distance between the ground truth probabilities $\Pr[X_{t+1}|X_t]$ and $\Pr[Y_t|X_t]$ and their estimations $\widehat{\Pr}[X_{t+1}|X_t]$ and $\widehat{\Pr}[Y_t|X_t]$ using various methods. The result is shown in Figure 2.

Similar to the experiment shown in the main paper, the proposed method works the best in terms of estimating the HMM parameters, without sacrificing too much computational times. Much to one's surprise, the Baum-Welch algorithm is not working very well in terms of estimation error. This is possibly because we limit the maximum number of EM iterations to be 500 (default setting of the MATLAB implementation), which may not be enough for convergence. What is expected is that the computational time of Baum-Welch grows linearly with respect to the length of the HMM observations, while other methods are independent from it.

An interesting remark is that when $T = 12,800$, the per-iteration elapsed time of Baum-Welch is approximately 1 second. Recall that each iteration of Baum-Welch calls for the forward-backward algorithm, with complexity $\mathcal{O}(K^2T)$.

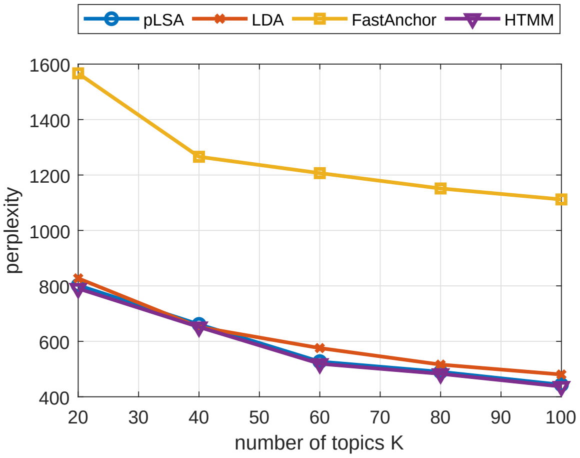

Figure 4: The perplexity of different models as number of topics $K$ increases.

This means for the problem size considered in the main paper, each iteration of Baum-Welch takes approximately 4 minutes to 7 hours, depending on the realization length. This is clearly not feasible in practice.

We also present the elapsed time of the four algorithms excluding the Baum-Welch algorithm for the case considered in the main paper, i.e., $N = 100$ and $K = 20$. Similar to the timing result shown in Figure 2, the proposed method takes the longest time compared to the other three, but not significantly; also recall that the propose method works considerably better in terms of estimation accuracy.

# 6 Additional HTMM Evaluations

In the main body of the paper we showed that HTMM is able to learn topics with higher quality using pairwise word cooccurrences. The quality of topics is evaluated using coherence, which is defined as follows. For each topic, a set of words $\mathcal{V}$ is picked (here we pick the top 20 words with the highest probability of appearing). We calculate the number of documents each word $v_1$ appears freq$(v_1)$ and the number of documents two words $v_1$ and $v_2$ both appear freq$(v_1, v_2)$. The coherence of that topic is calculated as

$$\sum_{v_1,v_2\in\mathcal{V}} \log\left(\frac{\text{freq}(v_1,v_2)+\epsilon}{\text{freq}(v_1)}\right).$$

The intuition is that if both $v_1$ and $v_2$ both have high probability of appearing in a topic, then they have high probability of co-occurring in a document as well; hence a higher value of coherence indicates a more indicative topic. The coherence of the individual topics are then averaged to get the coherence for the entire topic matrix.

Here we show some more evaluation results. Using the learned topic matrix, we can see how it fits the data directly from *perplexity*, defined as Blei et al. [2003]

$$\exp\left(-\frac{\sum_d \log p(\text{doc}_d)}{\sum_d L_d}\right).$$

A smaller perplexity means the probability model fits the data better. As seen in Figure 4, HTMM gives the smallest perplexity. Notice that since HTMM takes word ordering into account, it is not fair for the other methods to take the bag-of-words representation of the documents. The bag-of-words model is essentially multinomial, whose pdf includes a scaling factor $\frac{n!}{n_1!\ldots n_K!}$ for different combinations of observation orderings. In our case this factor is not included since we *do* know the word ordering in each document. For HTMM the log-likelihood is calculated efficiently using the forward algorithm.

This result is not surprising. Even using the same topic matrix, a bag-of-words model tries to find a $K$-dimensional representation for each document, whereas HTMM looks for a $K^2$-dimension representation. One may wonder if it is causing over-fitting, but we argue that it is not. First of all, we have see that in terms of coherence, HTMM learns a topic matrix with higher quality. For learning feature representations for each document, we showcase the following result. Once we have the topic-word probabilities and topic weights or topic transition probability, we can infer the underlying topic for each word. For bag-of-words models, each word only has one most probable topic in a document, no matter where it appears. For HTMM, once we learn the transition and emission probability, the topic of each word can be optimally estimated using the Viterbi algorithm. For one specific news article from the Reuters21578 data set, the topic inference given by pLSA is:

china daily vermin eat pct grain stocks survey provinces and cities showed vermin consume and pct china grain stocks china daily that each year mln tonnes pct china fruit output left rot and mln tonnes pct vegetables paper blamed waste inadequate storage and bad preservation methods government had launched national programme reduce waste calling for improved technology storage and preservation and greater production additives paper gave details

The word topic inference given by HTMM is:

china daily vermin eat pct grain stocks survey provinces and cities showed vermin consume and pct china grain stocks china daily that each year mln tonnes pct china fruit output left rot and mln tonnes pct vegetables paper blamed waste inadequate storage and bad preservation methods government had launched national programme reduce waste calling for improved technology storage and preservation and greater production additives paper gave details

As we can see, HTMM gets much more consistent and smooth inferred topics, which agrees with human understandings.

## 7 Learning HMMs from Triple-occurrences

Finally, we show a stronger identifiability result for learning HMMs using triple-occurrence probabilities.

**Theorem 1.** *Consider a HMM with $K$ hidden states and $N$ observable states. Suppose the emission probability $\Pr[Y_t|X_t]$ is generic (meaning probabilities not satisfying this condition form a set with Lebesgue measure zero), the transition probabilities $\Pr[X_{t+1}|X_t]$ are linearly independent from each other, and each conditional probability $\Pr[X_{t+1}|X_t]$ contains no more than $N/2$ nonzeros. Then this HMM can be uniquely identified from its triple-occurrence probability $\Pr[Y_{t-1}, Y_t, Y_{t+1}]$, up to permutation of the hidden states, for $K \leq \frac{N^2}{16}$.*

*Proof.* It is clear that identifiability holds when $K \leq N$, so we focus on the case that $N < K \leq \frac{N^2}{16}$.

As we explained in §1.1, the triple-occurrence probability can be factored into

$$\Pr[Y_{t-1}, Y_t, Y_{t+1}] = \sum_{k=1}^{K} \Pr[X_t = x_k] \Pr[Y_{t-1}|X_t = x_k] \Pr[Y_t|X_t = x_k] \Pr[Y_{t+1}|X_t = x_k].$$

Using tensor notations, this is equivalent to

$$\underline{\boldsymbol{\Omega}_3} = [\![\boldsymbol{p}; \boldsymbol{L}, \boldsymbol{M}, \boldsymbol{N}]\!],$$

where

$$\begin{aligned} p_k &= \Pr[X_t = x_k], \\ L_{nk} &= \Pr[Y_{t-1} = y_n|X_t = x_k], \\ N_{nk} &= \Pr[Y_{t+1} = y_n|X_t = x_k]. \end{aligned}$$

Let $\overline{\boldsymbol{\Theta}}$ denote the row scaled version of $\boldsymbol{\Theta}$ so that each row sums to one, then $\overline{\boldsymbol{\Theta}}$ denotes the transition probability. Then we have

$$\boldsymbol{L} = \boldsymbol{M}\overline{\boldsymbol{\Theta}}^{\top}. \tag{1}$$

Since $\boldsymbol{M}$ is generic and $\overline{\boldsymbol{\Theta}}$ is full rank, both $\boldsymbol{L}$ and $\boldsymbol{N}$ are generic as well. The latest tensor identifiability result by Chiantini and Ottaviani [2012, Theorem 1.1] shows that for a $N \times N \times N$ tensor with generic factors, the CPD $\underline{\boldsymbol{\Omega}_3} = [\![\boldsymbol{p}; \boldsymbol{L}, \boldsymbol{M}, \boldsymbol{N}]\!]$ is essentially unique if

$$K \leq 2^{2\lfloor \log_2 N \rfloor - 2},$$

or with a slightly worse bound

$$K \leq \frac{N^2}{16}.$$

This does not mean that any non-singular $\overline{\boldsymbol{\Theta}}$ can be uniquely recovered in this case. Equation (1) is underdetermined. A natural assumption to achieve identifiability is that each row of $\overline{\boldsymbol{\Theta}}$, i.e., each conditional transition probability $\Pr[X_{t+1}|X_t]$, can take at most $N/2$ non-zeros. In the context of HMM, this means that at a particular hidden state, there are only a few possible states for the next step, which is very reasonable. For a generic $\boldsymbol{M}$,

$$\mathrm{spark}(\boldsymbol{M}) = \mathrm{krank}(\boldsymbol{M}) + 1 = N + 1.$$

Donoho and Elad [2003] showed that for such a $\boldsymbol{M}$, and a vector $\boldsymbol{\theta}$ with at most $N/2$ nonzeros, $\boldsymbol{\theta}$ is the unique solution with at most $N/2$ nonzeros that satisfies $\boldsymbol{M\theta} = \boldsymbol{\ell}$. Therefore, if we seek for the sparsest solution to the linear equation (1), we can uniquely recover $\boldsymbol{\Theta}$ as well.

□

# References

Leonard E Baum, Ted Petrie, George Soules, and Norman Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains. *The Annals of Mathematical Statistics*, 41(1):164–171, 1970.

David M Blei, Andrew Y Ng, and Michael I Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3(Jan):993–1022, 2003.

Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

Luca Chiantini and Giorgio Ottaviani. On generic identifiability of 3-tensors of small rank. *SIAM Journal on Matrix Analysis and Applications*, 33(3):1018–1037, 2012.

David L. Donoho and Michael Elad. Optimally sparse representation in general (nonorthogonal) dictionaries via ell-one minimization. In *Proceedings of the National Academy of Sciences of the United States of America*, volume 100, pages 2197–202, 2003.

Meisam Razaviyayn, Mingyi Hong, and Zhi-Quan Luo. A unified convergence analysis of block successive minimization methods for nonsmooth optimization. *SIAM Journal on Optimization*, 23(2):1126–1153, 2013.