# A Fast Variational Approach for Learning Markov Random Field Language Models: Appendix

## A. Proof of Lemma 1

Let us start by reviewing some of the notions supporting the model presented in the main paper. All probability distributions defined in this paper correspond to Markov random field sequence models, so we begin by describing these models in detail.

### A.1. Sequence models

**Linear-chain Markov sequence model** We start by considering a sequence $\mathbf{x} = (x_1, \ldots, x_n)$ of $n$ variables with state space $\mathcal{X}$. We define an order $K$ Markov sequence model as a Markov random field where each element of the sequence is connected to its $K$ left and right neighbours. Figure 1 presents such a model for $n = 8$, $K = 2$.

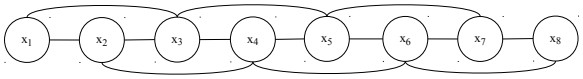


*Figure 1.* Second-order Markov sequence model for $n = 8$.

As mentioned in Section 3, a pairwise MRF with this structure gives the following distribution $p^n_{\text{seq}_K}$ over $\mathcal{X}^n$:

$$\forall \mathbf{x} \in \mathcal{X}^n, \quad \log(p^n_{\text{seq}_K}(\mathbf{x})) = \sum_{i=1}^{n-K} \sum_{l=1}^{K} \theta^{(i,i+l)}_{x_i, x_{i+l}} - A^n_{\text{seq}_K}(\theta) \tag{1}$$

Where:

$$A^n_{\text{seq}_K}(\theta) = \log \left( \sum_{\mathbf{y} \in \mathcal{X}^n} \exp \left( \sum_{i=1}^{n-K} \sum_{l=1}^{K} \theta^{(i,i+l)}_{y_i, y_{i+l}} \right) \right)$$

**Cyclic Markov sequence model** Now consider the cyclic version of the above sequence model, where the last $K$ tokens are connected to the first $K$ (specifically, edges are added between $v_{n-k}$ and $v_l$, $\forall 1 \leq k + l \leq K$), as illustrated in Figure 2.

This gives the following distribution $p^n_{\text{cycl}_K}$ over $\mathcal{X}^n$:

$$\forall \mathbf{x} \in \mathcal{X}^n, \quad \log(p^n_{\text{cycl}_K}(\mathbf{x})) = \sum_{i=1}^{n} \sum_{l=1}^{K} \theta^{(i,i+l)}_{x_i, x_{i+l}} - A^n_{\text{cycl}_K}(\theta) \tag{2}$$

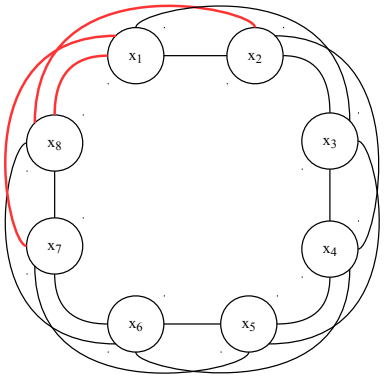


*Figure 2.* Cyclic second order Markov sequence model for $n = 8$.

Where:

$$A^n_{\text{cycl}_K}(\theta) = \log \left( \sum_{\mathbf{y} \in \mathcal{X}^n} \exp \left( \sum_{i=1}^{n} \sum_{l=1}^{K} \theta^{(i,i+l)}_{y_i, y_{i+l}} \right) \right)$$

And $\forall l \in [1, K]$, $x_{n+l} = x_l$, $y_{n+l} = y_l$.

### A.2. Language modelling

To apply a Markov sequence model to language modelling we also need to explicitly handle the boundary cases of a sentence.

Consider a linear-chain Markov sequence model over a sentence of size $M$, let $\mathcal{T}$ denote the vocabulary of our corpus, and define the bidirectional context of a word as its $K$ left and right neighbouring tokens. By adding $K$ "padding" or "separator" tokens $\langle S \rangle \notin \mathcal{T}$ to the left and right boundary of the sentence, this notion of context also allows us to bias the distribution of tokens at the beginning and end of the sentence.

In terms of the sequence model defined above, a sentence $\mathbf{t} \in \mathcal{T}^M$ will then correspond to a sequence $\mathbf{x}(\mathbf{t}) \in \mathcal{X}^{M+2K}$, with $\mathcal{X} = \mathcal{T} \cup \{\langle S \rangle\}$, such that $\mathbf{x}(\mathbf{t})^{K+M}_{K+1} = \mathbf{t}$ and $\mathbf{x}(\mathbf{t})^K_1 = \mathbf{x}(\mathbf{t})^{M+2K}_{M+K+1} = \langle S \rangle^K$.

This allows us to define the following distribution $p^M$ over

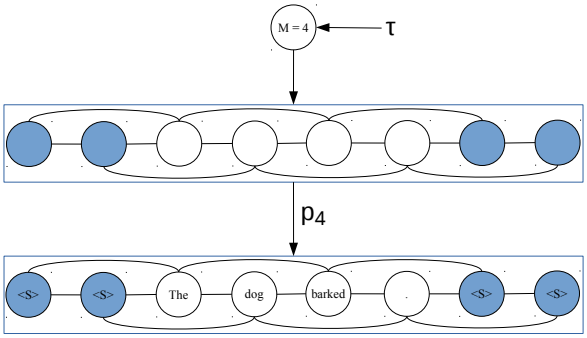


*Figure 3.* The full generative model for a sentence.

sentences of length $M$:

$$\forall \mathbf{t} \in \mathcal{T}^M,\ p^M(\mathbf{t}) = p_{\text{seq}_K}^{M+2K}\Big(\mathbf{x}_{K+1}^{K+M} = \mathbf{t} \Big| \mathbf{x}_1^K = \mathbf{x}_{M+K+1}^{M+2K} = \langle S \rangle^K\Big) \quad (3)$$

We can then define the following generative process for sentences (as illustrated in Figure 3):

- Draw the sentence length $M$ from a distribution over integers $\tau$.
- Draw a sequence of $M$ tokens: $\mathbf{t}^M = (t_1, \ldots, t_M) \sim p^M(\mathbf{t}^M)$ .

Under this model, the likelihood of a corpus $\mathbf{t}^c = (\mathbf{t}^1, \ldots, \mathbf{t}^{n_c})$ is then:

$$\begin{aligned} p(\mathbf{t}^c) &= \prod_{i=1}^{n_c} \tau(M_i) p^{M_i}(\mathbf{t}^i) \\ &= \tau(M_1, \ldots, M_{n_c}) \prod_{i=1}^{n_c} p^{M_i}(\mathbf{t}^i) \end{aligned}$$

Since the maximum likelihood parameters of $\tau$ can easily be estimated, we focus on the second part $\prod_{i=1}^{n_c} p^{M_i}(\mathbf{t}^i)$ in the rest of the proof.

### A.3. Proving the Lemma

Now we consider the lemma of interest relating the linear-chain Markov sequence model to the cyclic model. We restate the lemma here:

**Lemma.** *Let* $\mathcal{S} = \{\mathbf{x}(\mathbf{t^c}) | \mathbf{t^c} \in \mathcal{T}^{M_1} \times \ldots \times \mathcal{T}^{M_{n_c}}\}$. *Then,*

$$\prod_{i=1}^{n_c} p^{M_i}(\mathbf{t}_i) = \frac{p^N_{cycl_K}(\mathbf{x} = \mathbf{x}(\mathbf{t}))}{p^N_{cycl_K}(\mathbf{x} \in \mathcal{S})}.$$

Our proof first shows how to chain together sentences in a corpus, and then applies the cyclic Markov sequence model.

**Concatenating sentences** Consider a corpus of $c$ sentences $\mathbf{x}(\mathbf{t^c}) = (\mathbf{t}^1, \ldots, \mathbf{t}^{n_c})$ (of lengths $(M_1, \ldots, M_c)$) independently drawn from the above model. As above, we can use a mapping $\mathbf{x}$ of $\mathbf{t}$ to $\mathcal{X}^{N+K}$, where: $N = K + M_1 + \ldots + K + M_{n_c}$, by adding $\langle S \rangle$ tokens at the beginning and end of the corpus and between adjacent sentences:

$$\begin{aligned} &\mathbf{x}(\mathbf{t}^1, \ldots, \mathbf{t}^{n_c}) = \quad (4) \\ &\Big(\underbrace{\langle S \rangle, \ldots, \langle S \rangle}_{\times K}, t^1_1, \ldots, t^1_{M_1}, \underbrace{\langle S \rangle, \ldots, \langle S \rangle}_{\times K}, t^2_1, \\ &\qquad \ldots, t^c_{M_c}, \underbrace{\langle S \rangle, \ldots, \langle S \rangle}_{\times K}\Big) \end{aligned}$$

Let us first consider the base case where $c = 2$. From Equations 1 and 3, we get that:

$$\begin{aligned} \forall j \in \{1, 2\} \quad p^{M_j}(\mathbf{t}^j) &\propto p_{\text{seq}_K}^{M_j+2K}(\mathbf{x}(\mathbf{t}^j)) \\ &\propto \exp\Big(\sum_{i=1}^{M_j+K} \sum_{l=1}^{K} \theta^l_{\mathbf{x}(\mathbf{t}^j)_i, \mathbf{x}(\mathbf{t}^j)_{i+l}}\Big) \end{aligned}$$

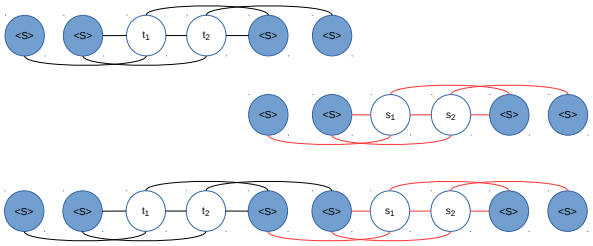


*Figure 4.* Concatenating sentences $\mathbf{t}^1$ and $\mathbf{t}^2$

Additionally, we have by construction:

$$\begin{aligned} \forall l \in [1, K], \quad \mathbf{x}(\mathbf{t^1})_{M_1+K+l} &= \mathbf{x}(\mathbf{t^2})_l \\ &= \mathbf{x}(\mathbf{t^1}, \mathbf{t^2})_{M_1+K+l} \\ &= \langle S \rangle \end{aligned}$$

Hence:

$$\sum_{i=1}^{M_1+M_2+2K} \sum_{l=1^K} \theta^l_{\mathbf{x}(\mathbf{t^1},\mathbf{t^2})_i,\mathbf{x}(\mathbf{t^1},\mathbf{t^2})_{i+l}} = \sum_{i=1}^{M_1+K} \sum_{l=1^K} \theta^l_{\mathbf{x}(\mathbf{t^1})_i,\mathbf{x}(\mathbf{t^1})_{i+l}} + \sum_{j=1}^{M_2+K} \sum_{l=1^K} \theta^l_{\mathbf{x}(\mathbf{t^2})_j,\mathbf{x}(\mathbf{t^2})_{j+l}}$$

In other words:

$$\begin{aligned} p^{M_1}(\mathbf{t}^1)p^{M_2}(\mathbf{t}^2) &\propto \exp\Big(\sum_{i=1}^{M_1+K}\sum_{l=1}^{K}\theta^l_{\mathbf{x}(\mathbf{t^1})_i,\mathbf{x}(\mathbf{t^1})_{i+l}}\Big) \\ &\quad \times \exp\Big(\sum_{i=1}^{M_2+K}\sum_{l=1}^{K}\theta^l_{\mathbf{x}(\mathbf{t^2})_i,\mathbf{x}(\mathbf{t^2})_{i+l}}\Big) \\ &\propto \exp\Big(\sum_{i=1}^{M_1+M_2+2K}\sum_{l=1^K}\theta^l_{\mathbf{x}(\mathbf{t^1},\mathbf{t^2})_i,\mathbf{x}(\mathbf{t^1},\mathbf{t^2})_{i+l}}\Big) \\ &\propto p^{M_1+M_2+3K}_{\text{seq}_K}(\mathbf{x} = \mathbf{x}(\mathbf{t^1},\mathbf{t^2})) \end{aligned}$$

By induction, we get that:

$$\prod_{i=1}^{n_c} p^{M_i}(\mathbf{t^i}) \propto p^{N+K}_{\text{seq}_K}(\mathbf{x} = \mathbf{x}(\mathbf{t}))$$

Now, let $\mathcal{S}_N = \{\mathbf{x}(\mathbf{t})|\mathbf{t} \in \mathcal{T}^{M_1} \times \ldots \times \mathcal{T}^{M_c}\}$. Since the text model is defined for $\mathbf{x} \in \mathcal{S}_N$, by normalization, it then follows that:

$$\begin{aligned} \prod_{i=1}^{n_c} p^{M_i}(\mathbf{t^i}) &= \frac{p^{N+K}_{\text{seq}_K}(\mathbf{x} = \mathbf{x}(\mathbf{t}))}{p^{N+K}_{\text{seq}_K}(\mathbf{x} \in \mathcal{S}_N)} \\ &= p^{N+K}_{\text{seq}_K}(\mathbf{x} = \mathbf{x}(\mathbf{t})|\mathbf{x} \in \mathcal{S}_N) \end{aligned} \quad (5)$$

**Using a cyclic model** Finally, $\forall \mathbf{x} \in \mathcal{S}_N$, we have that $\forall l \in [1, K], x_l = x_{N+l} = \langle S \rangle$. According to Equations 1 and 2, this means that:

$$\forall \mathbf{x} \in \mathcal{S}_N, \;\; p^{N+K}_{\text{seq}_K}(\mathbf{x}) \propto p^{N}_{\text{cycl}_K}(\mathbf{x})$$

Hence:

$$\prod_{i=1}^{n_c} p^{M_i}(\mathbf{t^i}) \propto p^{N}_{\text{cycl}_K}(\mathbf{x})$$

Which by normalization gives us:

$$\begin{aligned} \prod_{i=1}^{n_c} p^{M_i}(\mathbf{t^i}) &= \frac{p^{N}_{\text{cycl}_K}(\mathbf{x} = \mathbf{x}(\mathbf{t}))}{p^{N}_{\text{cycl}_K}(\mathbf{x} \in \mathcal{S}_N)} \\ &= p^{N}_{\text{cycl}_K}(\mathbf{x} = \mathbf{x}(\mathbf{t})|\mathbf{x} \in \mathcal{S}_N) \end{aligned} \quad (6)$$

Which proves the lemma.

## B. Implementation Details

**Synthetic data generation** The synthetic data used to obtain the results presented in Figure 5 consists of a sequence of 12 tokens sampled uniformly at random from $\mathcal{T} = \{$a,b,c,d$\}$. For $K = 2$ this gives a sequence of the form:

$$\langle S \rangle \; \langle S \rangle \text{ a b c d b a b d c b a c } \langle S \rangle \; \langle S \rangle$$

**Language modelling experiments** In our implementation of the inner loop of the algorithm (Algorithm 1 of the main paper), we use LBFGS to find the optimal value of $\delta$. However, as mentioned in Section 4, the inner loop does not need to be run to optimality to find an ascent direction.

The LBL model in Figure 6 was trained using SGD on minibatches of size 64. The learning rate was initialized at 0.1, and halved any time the error validation went up.

The second model presented in Table 1 (MRF SGD) was trained by running SGD on the model pseudo-likelihood, with minibatches of size 100. The learning rate was initialized at 0.025 and decayed as $\frac{1}{t^{0.4}}$.

**Sequence tagger features** For part-of-speech tagging experiments we make use of two feature functions $g(t_i, t_{i+l})$ and $f(t_i, w_{i+m})$. The tag-tag function $g$ simply consists of indicator features for all possible pairs of tags. The feature function $f(t_i, w_{i+m})$ conjoins an indicator of the tag $t_i$ with surface-form features including:

- An indicator for the word $w_{i+m}$ itself.
- Prefixes and suffixes of $w_{i+m}$ up to length 4.

When $m = 0$, i.e. the potential with the tag directly above a word, the tag is further conjoined with a standard set of morphology features including:

- Is $w_i$ completely upper case?
- Is the first letter of $w_i$ upper case?
- Does $w_i$ end with 's'?
- Is the first letter of $w_i$ upper case and it ends with 's'?
- Is $w_i$ completely upper-case and it end with 'S'?
- Does $w_i$ contain a digit?
- Is $w_i$ all digits?
- Does $w_i$ contain a hyphen?