Prior-informed soft label propagation for partially supervised hidden Markov models

Filip Wichrowski, Katarzyna Kaczmarek-Majer

Systems Research Institute, Polish Academy of Sciences, Newelska 6, 01-447 Warsaw

8-12 June 2025

Abstract

Hidden Markov Models (HMMs) are widely used latent-variable models that offer flexible approaches for learning from sequential data. However, standard HMM training typically relies on either fully supervised or fully unsupervised methods, limiting their applicability in real-world scenarios where partially labeled data is available. To address this, HMMs can be adapted by modifying the forward-backward recursions to effectively constrain certain paths in the hidden state lattice when some of the states are known. Unfortunately, this method does not allow for soft or uncertain labels. In this work, we propose a novel semi-supervised HMM training strategy that, instead of altering the forwardâbackward variables, introduces a weight matrix to scale the emission probabilities based on prior beliefs. We propose a method for label propagation that extends a known label to its neighbourhood by convolving an indicator function with a Gaussian kernel. Finally, we demonstrate the effectiveness of the proposed approach through a brief experiment on simulated data.

Introduction

A Tied-Mixture Hidden Markov Model (TM-HMM) is a latent variable model with an underlying Markov chain $\mathbf{X} = \{X_t\}_{t=1}^{T}$ where $T \in \mathbb{N}_+$ with space $\mathcal{S}_\mathbf{X} = \{1, .., N\}$ of hidden states. The initial state is drawn from a distribution $\pi = \{\pi_i\} = \mathbb{P}(X_1 = i)$ and the process evolves according to the transition probability matrix $\mathbf{A} = \{a_{ij}\} = \{\mathbb{P}(X_{t+1} = j \mid X_t = i)\}$, $i, j \in \mathcal{S}_\mathbf{X}$. Given X_t, an observation Y_t is emitted according to a tied Gaussian mixture

$$p_j(Y_t) = p(Y_t \mid X_t = j) = \sum_{k=1}^{M} w_{jk} \mathcal{N}(Y_t; \mu_k, \sigma_k), \tag{1}$$

where $M \in \mathbb{N}_+$ and $\mathcal{N}$ denotes the probability density function of a normal distribution. To infer about $\mathbf{X}$, dynamic programming approach is used in a form of a forward recursion α

$$\alpha_t(j) = \sum_{i=1}^{N} \alpha_{t-1}(i) a_{ij} p_j(Y_t) \tag{2}$$

Typically, HMMs rely on unsupervised learning. To incorporate partial labeling, the forward recursion is modified. If the state X_t is observed to be $i \in \mathcal{S}_\mathbf{X}$, then $\alpha_t(i)$ remains unchanged but $\alpha_t(j) = 0 \ \forall_{j \neq i}$. If X_t is unobserved, $\alpha_t(\cdot)$ is updated in the usual way.

The method: soft labels

To allow for soft labelling, we introduce a weight matrix

$$\varphi = \{\varphi_{it}\}^{N \times T} \tag{3}$$

where $\varphi_{jt} \in [0, 1]$ and $\sum_{j=1}^{N} \varphi_{jt} = 1 \ \forall_t$. Each column of φ defines a discrete probability distribution $\varphi_{\cdot t}$ on the state space $\mathcal{S}_\mathbf{X}$, that is

$$\varphi_{\cdot t} : \mathcal{S}_\mathbf{X} \mapsto [0, 1] \tag{4}$$

which is used to reweight the emission probabilities as

$$\tilde{b}_j(Y_t) = \varphi_{\cdot t}(j) b_j(Y_t) \quad \forall t, j \tag{5}$$

Soft labels allow for incorporating prior beliefs about the state structure. If no information is available at time t, then $\varphi_{\cdot t}(j) = 1/N$, $\forall j \in \mathcal{S}_\mathbf{X}$.

The method: prior knowledge

In numerous real-world scenarios, transitions between system states occur rarely. Hence, we assume that if $X_t = i$, then there exists a neighbourhood $N(t, \varepsilon) = [t - \varepsilon, t + \varepsilon]$, $\varepsilon \in \mathbb{N}_+$ of t in which the state i is much more likely than any other state. To express that belief mathematically, consider a binary vector

$$v_j = (v_1^{(j)}, ..., v_T^{(j)}), \quad v_t^{(j)} = \begin{cases} 1 \text{ iff } X_t = j, \\ 0 \text{ iff } X_t \neq j, \end{cases} \quad j \in \mathcal{S}_\mathbf{X}, \tag{6}$$

of known states. Denote by $\tau_j = \{t_1^{(j)}, ..., t_m^{(j)}\}, t_i^{(j)} \in \{1, ..., T\}$, $m \in \mathbb{N}_+$ the set of time points at which the known label is j. To get a vector $\tilde{v}_j = (\tilde{v}_1^{(j)}, ..., \tilde{v}_T^{(j)})$ of propagated labels j in the neighbourhood of each $t_i^{(j)} \in \tau_j$ a Gaussian kernel g is used

$$g_k = \frac{\exp\{-k^2/(2s^2)\}}{\sum_{|k| \leq \varepsilon} \exp\{-k^2/(2s^2)\}}, k = -\varepsilon, ..., \varepsilon \tag{7}$$

that is convolved with v_j:

$$\tilde{v}^{(j)} = (v^{(j)} * g)_t = \sum_{|k| \leq \varepsilon} v_t^{(j)} g_k, \quad t = 1, ..., T, \tag{8}$$

and adjust the distribution $\varphi_{\cdot t}$ by putting

$$\tilde{\varphi}_{\cdot t}(j) = \frac{\varphi_{\cdot t}(j) \cdot \tilde{v}^{(j)}}{\sum_{i=1}^{N} \varphi_{\cdot t}(i) \cdot \tilde{v}^{(i)}} \tag{9}$$

Results

Assume a Markov process: $\mathcal{S}_\mathbf{X} = \{s_0, s_1, s_2\}$, transition matrix $\mathbf{A}$, initial distribution π

$$\mathbf{A} = \begin{bmatrix} 0.90 \ 0.05 \ 0.05 \\ 0.05 \ 0.90 \ 0.05 \\ 0.05 \ 0.05 \ 0.90 \end{bmatrix}, \quad \pi = \begin{bmatrix} 1/3 \\ 1/3 \\ 1/3 \end{bmatrix}, \tag{10}$$

that is used to generate a vector $\mathbf{X} = (X_1, \ldots, X_L)$ of labels. A vector $\mathbf{Y} = (Y_1, ..., Y_L)$ of observations is drawn such that each observation Y_t is conditionally distributed according to a three-component Gaussian mixture given the state $X_t = s_i$

$$p_{s_i}(Y_t) = p(Y_t \mid X_t = s_i) = \sum\nolimits_{k=1,2,3} \mathbb{1}\{k = i\} \cdot \mathcal{N}(Y_t; \mu \cdot (i - 1), 1) \tag{11}$$

Example data is shown in the Figure 1. The experimental setup, repeated 20 times, is:

1. draw training data $(\mathbf{X}, \mathbf{Y})$; randomly discard $k\%$ labels to obtain $\mathbf{X}^*$,
2. using $\mathbf{Y}$, fit four different hidden Markov models
   - US: unsupervised (i.e. weight matrix $\varphi_{jt} = 1/|\mathcal{S}_\mathbf{X}| \ \forall_{t,j}$)
   - SS: semi-supervised (partially observed labels),
   - $\mathrm{SS}{:}\,\mathrm{Conv}(0.5)$: semi-supervised; partial labels are convolved with Gaussian kernel with $\sigma^2 = 0.5$,
   - $\mathrm{SS}{:}\,\mathrm{Conv}(1.0)$: semi-supervised; partial labels are convolved with Gaussian kernel with $\sigma^2 = 1.0$,
3. draw testing data $(\mathbf{X_T}, \mathbf{Y_T})$, predict labels, calculate F1 score, repeat 20 times.

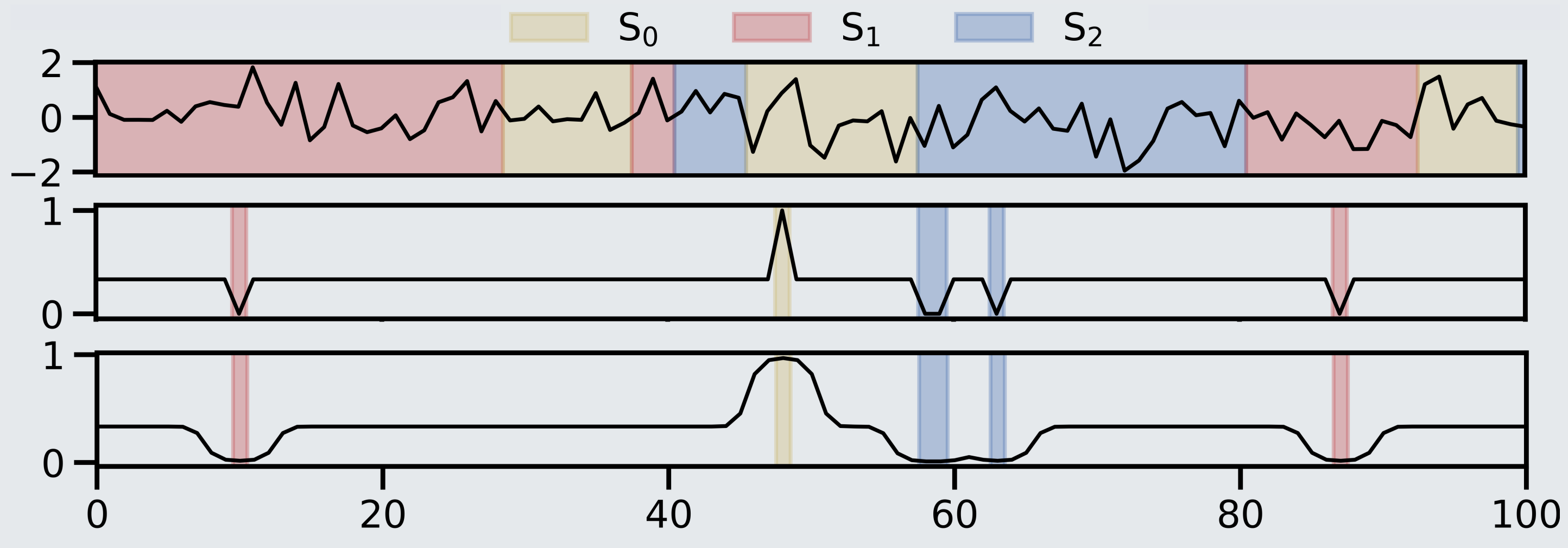

Figure 1: Top panel: an example sequence $\mathbf{X}$ and observations $\mathbf{Y}$. Middle panel: partially observed states (95% missing) and corresponding hard labels for state s_0. Bottom panel: partially observed states and corresponding soft labels (for state s_0) as a product of convolving hard labels with Gaussian kernel ($\sigma^2 = 1$).

Figure 2 presents the results for various percentages of missing labels and different mean values μ. Partial supervision notably refines unsupervised learning, and (propagated) soft labels can further improve performance in situations of extreme label scarcity.

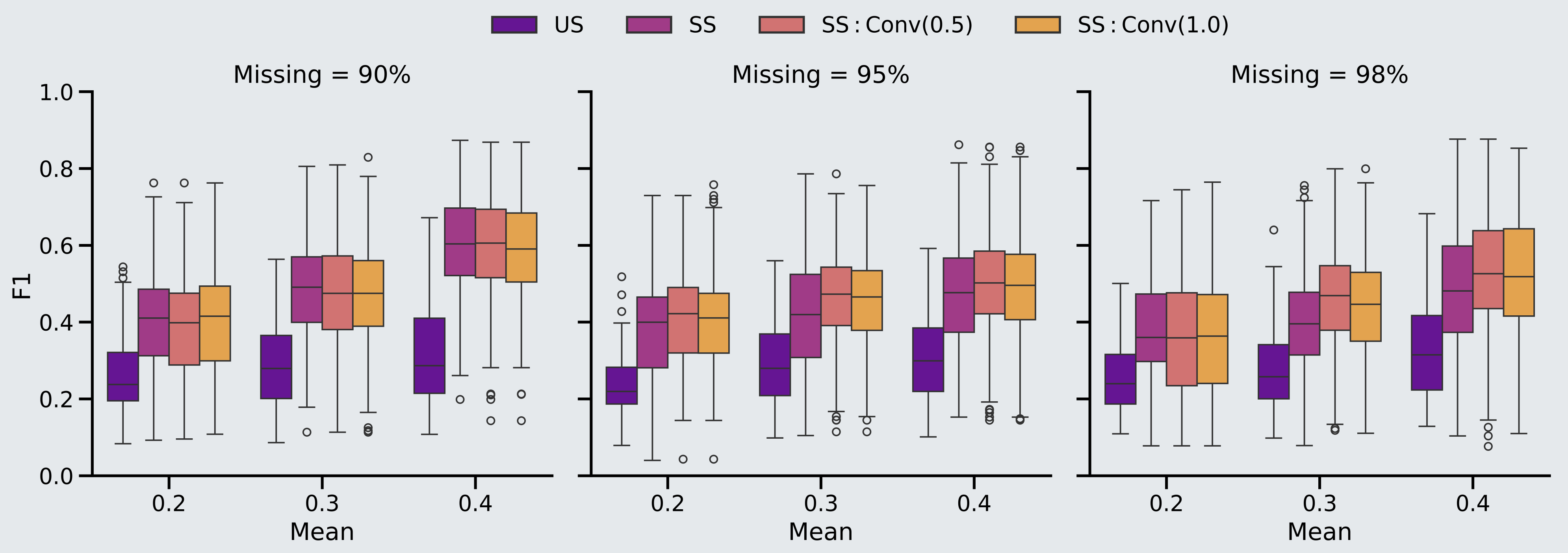

Figure 2: F1 values for different percentages of missing labels and means of a normal distribution.

Conclusions

Using soft labels and incorporating prior knowledge of the labels' temporal structure appears to be a natural and promising enhancement to partial supervision. However, further experiments are needed to evaluate the measurable benefits of this approach.

References

1. Huang, X. D., Jack, M. A., *Semi-continuous hidden Markov models for speech signals*, Computer Speech & Language, 1989,
2. Li, J., Lee, JY. & Liao, L. *A new algorithm to train hidden Markov models for biological sequences with partial labels.* BMC Bioinformatics 22, 2021.

The project *ExplainMe: Explainable Artificial Intelligence for Monitoring Acoustic Features extracted from Speech* (FENG.02.02-IP.05-0302/23) is carried out within the First Team programme of the Foundation for Polish Science co-financed by the European Union under the European Funds for Smart Economy 2021-2027 (FENG).