Extended Abstract Track

Graph Neural Networks for Connectivity Inference in Spatially Patterned Neural Responses

Editors: List of editors' names

Abstract

A continuous attractor network is one of the most common theoretical framework for studying a wide range of neural computations in the brain. Many previous approaches have attempted to identify continuous attractor systems by investigating the state-space structure of population neural activity. However, establishing the patterns of connectivity for relating the structure of attractor networks to their function is still an open problem. In this work, we propose the use of graph neural networks combined with the structure learning for inferring the recurrent connectivity of a ring attractor network and demonstrate that the developed model greatly improves the quality of circuit inference as well as the prediction of neural responses compared to baseline inference algorithms.

Keywords: continuous attractor networks, connectivity inference, graph neural networks

1. Introduction

Continuous attractor networks (CAN) have become some of the most influential circuit models in systems neuroscience. A CAN defined by a symmetric recurrent connectivity leads to spontaneous and stable neural activity patterns and the stable states can be used to represent numerous external variables such as the orientation of visual stimuli (Ben-Yishai et al., 1995), head direction (Zhang, 1996), eye position (Seung, 1996), and spatial location (Fuhs and Touretzky, 2006; Burak and Fiete, 2009). The CAN's wide range of utility has spurred several computational studies to find evidence of attractor dynamics in the brain.

The classic approach for identification of continuous attractors is to probe for the invariance of correlation structures in neural population activity (Taube et al., 1990; Yoganarasimha et al., 2006; Fyhn et al., 2007; Yoon et al., 2013; Gardner et al., 2019; Trettel et al., 2019). The invariance is an essential condition because the states of the system are internally generated by strong recurrent connectivity, which constrains the activity of neurons to form a certain coactivation pattern; thus, pairwise cell-cell relationships should remain invariant across time, environmental conditions, and behavioral states, independent of external sensory inputs (Burak and Fiete, 2009). Another recent attempt to address this challenge has been made to directly characterize the full state-space of the population responses and visualize them by topological data analysis (Low et al., 2018; Chaudhuri et al., 2019; Rybakken et al., 2019; Gardner et al., 2022). The resulting topological structure of the attractor manifold provides the hints of an underlying circuit mechanism, and its invariance across the aforementioned external factors validates the main predictions of CAN models.

Nevertheless, few approaches are more reliable for identifying CAN than directly measuring the wiring diagrams of neural circuits. Hence, estimating the network connectivity

© 2022 .

from large population recordings has also been a long-standing problem (Paninski et al., 2003; Schneidman et al., 2006; Pillow et al., 2008). A fundamental challenge in circuit inference, however, is that many previous statistical models with strong parametric assumptions limit their ability to recover direct synaptic connections; thus, the inferred connectivity could be substantially different from the true connectivity, especially when neurons are highly correlated (Das and Fiete, 2020).

Here, we take a new approach to inferring structural connectivity with no stringent parametric assumptions based on neural spike data generated from a strongly recurrent ring attractor network. We adopt the framework of graph neural networks (GNNs) (Hamilton et al., 2017; Bronstein et al., 2017; Battaglia et al., 2018) combined with the structure learning module, an architectural design strategy that relies on learning the underlying circuitry jointly with predicting multi-neuronal spike responses. We show that this approach can be significantly more expressive than statistically sophisticated inference algorithms. Furthermore, we propose the use of an iterative neural message-passing scheme (Gilmer et al., 2017) that is closely matched to the generative nature of CAN. Taken together, these ideas suggest new possibilities for less biased inference of recurrent neural circuits.

2. Methods

2.1. Generative Model

We consider a ring network of 100 neurons where the process of generating neural spikes is by either a simple threshold-crossing or simulating an inhomogeneous Poisson process (Das and Fiete, 2020). The generative network model is designed to have rotation-invariant recurrent connections with a local Mexican hat weight profile and exhibit spatially periodic activity patterns drifting over time (Figure S1)[1]. We record spikes from the network for 8 minutes with a time step $\Delta t = 0.1$ ms and represent the spike trains via $\mathbf{x} \in \{0,1\}^{N \times L}$, where N is the number of neurons and L is the length of time steps (Figure S1). The data is split into 80%, 10%, and 10% over time for training, validation and testing.

2.2. Inference Model

Structure learning module The goal of this module is to model the pairwise connection strength q_{ij} given every pair of $(\mathbf{x}_i, \mathbf{x}_j)$ where $\mathbf{x}_i = (\mathbf{x}_i^1, \ldots, \mathbf{x}_i^L)$ denotes the spike train of ith neuron. We apply a 1D convolution with 10 kernels of size 200 [2] and a stride of 20 over each input spike train, vectorize along time dimension, and use a fully connected (FC) layer to get a reduced output embedding vector $\mathbf{z}_i$:

$$\mathbf{z}_i = \mathrm{FC}(\mathrm{vec}(\mathrm{CONV}_{1d}(\mathbf{x}_i))) \tag{1}$$

Aside from these layers, we use batch normalization (Ioffe and Szegedy, 2015) placed after `ReLU` activation function. We then concatenate every pair of $(\mathbf{z}_i, \mathbf{z}_j)$ and apply a multilayer perceptron (MLP)[3] that has two hidden layers with 32 units to infer the strength of network

1. This generative model is equivalent to the highly structured ring network in the strongest weight regime by Das and Fiete (2020) when the scalar weight strength parameter $r = 0.025$.
2. The kernel size is set twice as long as the synaptic time constant ($\tau = 10$ ms) of the generative model.
3. The last activation is set to be a `sigmoid`.

Extended Abstract Track

connectivity:

$$\mathbf{w}_{ij} = \mathrm{MLP}(\mathbf{z}_i \parallel \mathbf{z}_j) \tag{2}$$

The resulting inferred weight matrix $\mathbf{w} \in \mathbb{R}^{N\times N}$ is assumed to be symmetric and have no self-loop. This constraint is weaker than the rotation-invariance of target recurrent weight strengths. In simple terms, every neuron in the proposed inference model does not have to share the same outgoing synaptic weights to all of the other neurons around the ring network.

Spike prediction module The task of this module is to predict the spike activity $\mathbf{x}$ for every neuron given the latent circuitry $\mathbf{w}$:

$$p_\theta(\mathbf{x}|\mathbf{w}) = \prod_{t=1}^{T} p_\theta\left(\mathbf{x}^{t+1}|\mathbf{x}^t, \ldots, \mathbf{x}^1, \mathbf{w}\right) \tag{3}$$

where $\mathbf{x}^t = (\mathbf{x}_1^t, \ldots, \mathbf{x}_N^t)$ represents the spike activity of all N neurons at time t. We assume that the attractor dynamics can be implicitly learned by modeling instantaneous neural spike rates $\boldsymbol{\lambda}^{t+1}$ where $\mathrm{Poisson}(\boldsymbol{\lambda}^{t+1}) = p_\theta\left(\mathbf{x}^{t+1}|\mathbf{x}^t, \ldots, \mathbf{x}^1, \mathbf{z}\right)$ through the message-passing operations:

$$\mathbf{h}_{i,t}^0 = f_{\mathrm{emb}}\left(\mathbf{x}_i^t, \ldots, \mathbf{x}_i^{t-2\tau/\Delta t+1}\right) \tag{4}$$

$$\mathbf{h}_{i,t}^{k+1} = \phi\left(\mathbf{h}_{i,t}^k,\ \mathbf{h}_{i,t}^0 \ \middle\|\ \sum_{j\neq i} \mathbf{w}_{ij} \cdot \psi\left(\mathbf{h}_{i,t}^k \parallel \mathbf{h}_{j,t}^k\right)\right) \tag{5}$$

$$\log\left(\lambda_i^{t+1}\right) = f_{\mathrm{out}}\left(\mathbf{h}_{i,t}^K\right) \tag{6}$$

where $\mathbf{h}_{i,t}^0$ is the initial embedding of the ith neuron's spike history for the past $\frac{\tau}{\Delta t}(= 200)$ steps at time t. Here, ψ is a learnable message function, computing the message sent from neuron j to i. At kth step of message passing, a learnable function ϕ combines the sum of incoming messages weighted by connection strength $\mathbf{w}_{ij}$ with the previous neuronal state $\mathbf{h}_{i,t}^k$ to generate the updated state $\mathbf{h}_{i,t}^{k+1}$. After K rounds of message passing, we use f_{out} to read out the log-firing rate from the output of the final propagation step (for our spike trains, K=1 was enough). This firing rate determines the rate of an inhomogeneous Poisson process that generates spikes in neuron i at time t+1. In this experiment, separate two-layer MLPs are used for f_{emb}, ψ, f_{out}, and a gated recurrent unit is applied to ϕ.

2.3. Training

The training loss is the negative log-likelihood of the recorded population activity and the objective is to predict the spike times accurately by minimizing the the likelihood using the Adam optimizer with a learning rate of 0.0005, decayed by a factor of 0.5 every 50 epochs:

$$\boldsymbol{\Theta}^*, \mathbf{w} = \arg\min_{\boldsymbol{\Theta}} \sum_{i=1}^{N}\sum_{t=1}^{L}\left(\lambda_i^t - \mathbf{x}_i^t \log \lambda_i^t\right) \tag{7}$$

where the parameters $\boldsymbol{\Theta}$ are the weights of the structure learning and spike prediction network modules, and $\mathbf{w}$ is the best inferred connectivity weight matrix.

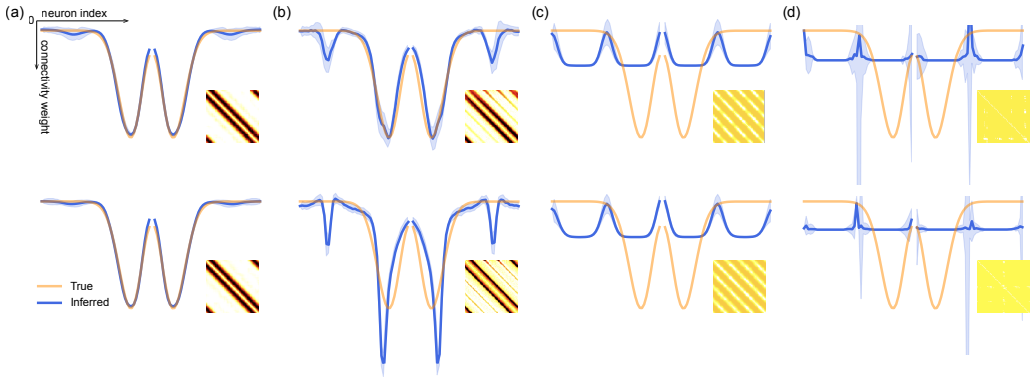

Figure 1: **Quality of circuit inference using 4.8×10^6 spikes from a strongly recurrent ring attractor network. (a)** True Mexican hat-shaped connectivity weight (orange) superimposed by the average inferred weight profile by GNN (blue). The inset at the bottom right presents the inferred weight matrix $\hat{\mathbf{W}}$. The model performs inference on spike data generated by threshold-crossing (top) and linear-nonlinear-poisson (bottom) models. The followings are the quality of inference by **(b)** GLM, **(c)** seqNMF, and **(d)** TCA.

3. Experiments

To evaluate the quality of inferred connectivity, the outgoing weight profiles $\mathbf{w}$ of all neurons are realigned and rescaled to match the scale of the true weights $\mathbf{W}$ because the message function ψ in Equation (5) may compensate for an arbitrary scale of $\mathbf{w}$. We then use the following two metrics to compare the proposed approach to three baselines[4]: i) the normalized inference error, computed as $\Delta = \|\mathbf{W} - \hat{\mathbf{W}}\|_F / \|\mathbf{W}\|_F$ where $\hat{\mathbf{W}}$ is the scaled version of $\mathbf{w}$ and $\|\cdot\|_F$ denotes the Frobenius-norm and ii) the log-likelihood score $\mathcal{L}$ in Equation (7). Every alternative inference method exhibits an error biased toward inferring unconnected neurons, evidenced by side bumps in the weight profile or by multiple off-diagonal stripes in the weight matrix (Figures 1b-1d). This type of systematic inference error is interpreted as overestimated connections resulting from highly correlated activity by strongly recurrent networks (Das and Fiete, 2020). Despite the failure to explain away strong correlations, the proposed inference model alleviates the side bumps (Figure 1a) and thus significantly reduces biased errors (Table. S1). More accurate estimation of spike activity in our model (Table. S1) suggests that the GNN-based spike prediction module is expressive enough to learn and match actual neural dynamics of a ring attractor network.

4. Conclusion

We presented a GNN-based neural inference model to understand the network mechanism of neural circuits by reconstruction of structural connectivity from population neural spike data. We demonstrated that the proposed model is highly effective at inferring direct connections of a strongly recurrent network. It will be interesting to test this model framework by examining real population recording data, and to see whether the inferred circuitry supports the continuous attractor dynamics in the brain.

4. The baselines are tensor component analysis (TCA) (Williams et al., 2018), sequence non-negative matrix factorization (seqNMF) (Mackevicius et al., 2019), and generalized linear model (GLM) (Pillow et al., 2008) (see more details in Appendix A).

References

Peter W Battaglia, Jessica B Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, et al. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, 2018.

Rani Ben-Yishai, R Lev Bar-Or, and Haim Sompolinsky. Theory of orientation tuning in visual cortex. *Proceedings of the National Academy of Sciences*, 92(9):3844–3848, 1995.

Michael M Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: Going beyond euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 2017.

Yoram Burak and Ila R Fiete. Accurate path integration in continuous attractor network models of grid cells. *PLoS Computational Biology*, 5(2):e1000291, 2009.

Rishidev Chaudhuri, Berk Gerçek, Biraj Pandey, Adrien Peyrache, and Ila Fiete. The intrinsic attractor manifold and population dynamics of a canonical cognitive circuit across waking and sleep. *Nature Neuroscience*, 22(9):1512–1520, 2019.

Abhranil Das and Ila R Fiete. Systematic errors in connectivity inferred from activity in strongly recurrent networks. *Nature Neuroscience*, 23(10):1286–1296, 2020.

Mark C Fuhs and David S Touretzky. A spin glass model of path integration in rat medial entorhinal cortex. *Journal of Neuroscience*, 26(16):4266–4276, 2006.

Marianne Fyhn, Torkel Hafting, Alessandro Treves, May-Britt Moser, and Edvard I Moser. Hippocampal remapping and grid realignment in entorhinal cortex. *Nature*, 446(7132): 190–194, 2007.

Richard J Gardner, Li Lu, Tanja Wernle, May-Britt Moser, and Edvard I Moser. Correlation structure of grid cells is preserved during sleep. *Nature Neuroscience*, 22(4):598–608, 2019.

Richard J Gardner, Erik Hermansen, Marius Pachitariu, Yoram Burak, Nils A Baas, Benjamin A Dunn, May-Britt Moser, and Edvard I Moser. Toroidal topology of population activity in grid cells. *Nature*, 602(7895):123–128, 2022.

Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. *arXiv preprint arXiv:1704.01212*, 2017.

William L Hamilton, Rex Ying, and Jure Leskovec. Representation learning on graphs: Methods and applications. *IEEE Data Engineering Bulletin*, 2017.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, pages 448–456. PMLR, 2015.

Ryan J Low, Sam Lewallen, Dmitriy Aronov, Rhino Nevers, and David W Tank. Probing variability in a cognitive map using manifold inference from neural dynamics. *BioRxiv*, page 418939, 2018.

Emily L Mackevicius, Andrew H Bahle, Alex H Williams, Shijie Gu, Natalia I Denisenko, Mark S Goldman, and Michale S Fee. Unsupervised discovery of temporal sequences in high-dimensional datasets, with applications to neuroscience. *Elife*, 8:e38471, 2019.

Liam Paninski, Eero Simoncelli, and Jonathan Pillow. Maximum likelihood estimation of a stochastic integrate-and-fire neural model. *Advances in Neural Information Processing Systems*, 16, 2003.

Jonathan W Pillow, Jonathon Shlens, Liam Paninski, Alexander Sher, Alan M Litke, EJ Chichilnisky, and Eero P Simoncelli. Spatio-temporal correlations and visual signalling in a complete neuronal population. *Nature*, 454(7207):995–999, 2008.

Erik Rybakken, Nils Baas, and Benjamin Dunn. Decoding of neural data using cohomological feature extraction. *Neural computation*, 31(1):68–93, 2019.

Elad Schneidman, Michael J Berry, Ronen Segev, and William Bialek. Weak pairwise correlations imply strongly correlated network states in a neural population. *Nature*, 440 (7087):1007–1012, 2006.

H Sebastian Seung. How the brain keeps the eyes still. *Proceedings of the National Academy of Sciences*, 93(23):13339–13344, 1996.

Jeffrey S Taube, Robert U Muller, and James B Ranck. Head-direction cells recorded from the postsubiculum in freely moving rats. ii. effects of environmental manipulations. *Journal of Neuroscience*, 10(2):436–447, 1990.

Sean G Trettel, John B Trimper, Ernie Hwaun, Ila R Fiete, and Laura Lee Colgin. Grid cell co-activity patterns during sleep reflect spatial overlap of grid fields during active behaviors. *Nature Neuroscience*, 22(4):609–617, 2019.

Alex H Williams, Tony Hyun Kim, Forea Wang, Saurabh Vyas, Stephen I Ryu, Krishna V Shenoy, Mark Schnitzer, Tamara G Kolda, and Surya Ganguli. Unsupervised discovery of demixed, low-dimensional neural dynamics across multiple timescales through tensor component analysis. *Neuron*, 98(6):1099–1115, 2018.

Doreswamy Yoganarasimha, Xintian Yu, and James J Knierim. Head direction cell representations maintain internal coherence during conflicting proximal and distal cue rotations: comparison with hippocampal place cells. *Journal of Neuroscience*, 26(2):622–631, 2006.

KiJung Yoon, Michael A Buice, Caswell Barry, Robin Hayman, Neil Burgess, and Ila R Fiete. Specific evidence of low-dimensional continuous attractor dynamics in grid cells. *Nature Neuroscience*, 16(8):1077–1084, 2013.

Kechen Zhang. Representation of spatial orientation by the intrinsic dynamics of the head-direction cell ensemble: a theory. *Journal of Neuroscience*, 16(6):2112–2126, 1996.

Extended Abstract Track

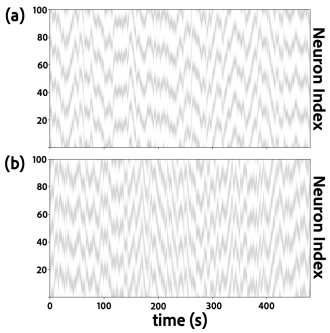

Table. S1: Quality of spike prediction $\mathcal{L}$ and inference error Δ

spiking method	threshold-crossing		linear-nonlinear-poisson	
	$\mathcal{L}$	Δ	$\mathcal{L}$	Δ
TCA	-	0.7619	-	0.7612
seqNMF	-	0.7885	-	0.7963
GLM	0.0409	0.2441	0.0430	0.4841
Ours	**0.0322**	**0.0614**	**0.0334**	**0.0487**

Figure. S1: **Spike raster plots of a ring attractor network binned at** 10 **ms. (a)** Spikes emitted by threshold-crossing. **(b)** Spikes by linear-nonlinear-poisson model.

Appendix A. Extended figures and tables

Appendix B. Algorithm of baselines

For notations, we denote $\cdot$ as matrix multiplication and $*$ as discrete convolution. We express exponentially weighted function as EW. Outer product is $\otimes$ and $\|\cdot\|_1$ denotes the sum of matrix or vector elements.

Algorithm 1: Generalized Linear Model

1. Initialize model coefficients $b, \mathbf{w}, \mathbf{s}$
2. For $i = 1$ to number of neurons N
   (a) For $k = 1$ to maximum function evaluations
      i. Stimuli Filter $\mathbf{a}^{(k)} = \mathbf{A}_{\text{basis}} \cdot \mathbf{s}^{(k)}$
      ii. If $i > 1$
         A. Circularly shift $\mathbf{w}^{(k)}$, i steps
      iii. $\mathbf{x}_i = \mathbf{X}_{i,L+1:end}$
      iv. $\mathbf{X}_{\text{rest}} = \mathbf{X}_{[1:i-1,i+1:N],1:L-1}$
      v. $\mathbf{x}_{\mathbf{w}}^{(k)} = \mathbf{X}_{\text{rest}} \cdot \mathbf{w}^{(k)}$
      vi. $\log \lambda_i^{(k)} = \mathbf{x}_{\mathbf{w}}^{(k)} * \mathbf{a}^{(k)} + b^{(k)}$
      vii. $\mathcal{L} = \sum_t \sum_i \left(\lambda_i^t - \mathbf{x}_i^t \log \lambda_i^t\right)$
      viii. $b^{(k+1)}, \mathbf{w}^{(k+1)}, \mathbf{s}^{(k+1)} = \underset{b,\mathbf{w},\mathbf{s}}{\arg\min}(\mathcal{L})$ by Quasi-Newton Method

In GLM inference model (algorithm 1), we make the basis $\mathbf{A}_{\text{basis}}$ as ten raised cosine filters of window size 200 steps. For initialization of model coefficients, $b = 0, \mathbf{w} = \overrightarrow{0}, \mathbf{s} = \overrightarrow{1}$. Then, we get stimuli filter $\mathbf{a}$ by product of $\mathbf{A}_{\text{basis}}$ with $\mathbf{s}$. Filtered spike train $\mathbf{x}_{\mathbf{w}}$ is obtained

the weighted linear combination of $\mathbf{A}_{\text{basis}}$. We modeled log Poisson rate as the convolution of $\mathbf{a}$ and the product of $\mathbf{x_w}$. Model coefficients are updated by Quasi-Newton method.

Algorithm 2: Tensor Component Analysis (TCA)

---

1. Initialize matrices of factors $\mathbf{N}, \mathbf{K}, \mathbf{G}$
2. For $i = 1$ to number of neurons
   (a) $\mathbf{M}_{i,:} = \text{EW}(\mathbf{X}_{i,:})$
3. Reshape $\mathbf{M} \in \mathbb{R}^{N \times L}$ to $\mathbb{R}^{N \times K \times G}, G = L/K$
4. For $k = 1$ to maximum number of iterations
   (a) For $r = 1$ to the number of factors
      i. Factors $\mathbf{n}^{(r)}, \mathbf{k}^{(r)}, \mathbf{g}^{(r)} = \underset{\mathbf{n}^{(r)}, \mathbf{k}^{(r)}, \mathbf{g}^{(r)}}{\arg\min} \left(\|\mathbf{M} - \sum_r \mathbf{n}^{(r)} \otimes \mathbf{k}^{(r)} \otimes \mathbf{g}^{(r)}\|_F \right)$ by Alternating Least-Square Method
5. For $r = 1$ to the number of factors
   (a) $\mathbf{E}^{(r)} = \|\mathbf{k}^{(r)}\|_1 \|\mathbf{g}^{(r)}\|_1 (\mathbf{n}^{(r)} \otimes \mathbf{n}^{(r)})$
6. $\mathbf{E} = \frac{1}{R} \sum_r \left(\mathbf{E}^{(r)} \right)$

TCA (algorithm 2) is a three-dimensional extended version of principle component analysis, where factorize the original tensor into neuron, time, and trial factors. First, we applied EM to spike train $\mathbf{X}$ along the time axis with smoothing factor $\alpha = 0.01$. Then exponentially weighted spike $\mathbf{M}$ is reshaped to have 10 trial components. Afterward, we applied TCA and took outer product of each neuron factor $\mathbf{n}$ with itself, trying to find coupling strength with other neurons. Each factor is optimized with alternating least-square method. Finally, we multiply $\mathbf{E}^{(r)}$ with the sum of trial and time elements along each factor. Overall predicted connectivity strength $\mathbf{E}$ is obtained by averaging $\mathbf{E}^{(r)}$.

SeqNMF (algorithm 3) is a dimension reduction method with regularized convolutional non-negative matrix factorization to extract sequential patterns. First, we defined smoothing matrix $\mathbf{S}$, with the length of sequence $L = 2$ and regularization parameter $\beta = 0.001$. Then, we get neuron and time factors by the method denoted in Algorithm 3, and took outer product of columns of learned neuron factors to get edge features. Similar to TCA, we multiplied the sum of time elements and averaged edge matrices $\mathbf{E}^{(r)}$. We use $R = 100$ factors for both dimension reduction based method.

Extended Abstract Track

Algorithm 3: Sequence Non-negative Matrix Factorization (seqNMF)

1. Initialize tensor and matrix of factors $\mathbf{W}, \mathbf{H}$
2. Define Smoothing Matrix $\mathbf{S}$, $\mathbf{S}_{i,j} = 1$ when $|i - j| < L$ otherwise $\mathbf{S}_{i,j} = 0$
3. Define regularization parameter β
4. For $k = 1$ to maximum iteration number
   (a) $\tilde{\mathbf{X}}^{(k)}_{i,j} = \sum_r \sum_l \mathbf{W}^{(k)}_{i,r,l} \mathbf{H}^{(k)}_{r,(j-l)}$
   (b) $\mathbf{C}^{(k)}_{i,j} = \sum_l \sum_n \mathbf{W}^{(k)}_{n,i,l} \mathbf{X}^{(k)}_{n,j+l}$
   (c) $\mathbf{W}^{(k+1)}, \mathbf{H}^{(k+1)} = \underset{\mathbf{W},\mathbf{H}}{\arg\min}\big(\|\tilde{\mathbf{X}}^{(k)} - \mathbf{X}\|_F + \beta \|\mathbf{C} \cdot \mathbf{S} \cdot \mathbf{H}^\top\|_{1,i \neq j}\big)$, by standard multiplicative update rules used for NMF and convNMF
5. For $r = 1$ to the number of factors R
   (a) $\mathbf{E}^{(r)} = \|\mathbf{H}_{:,r}\|_1 (\mathbf{W}_{:,r,0} \otimes \mathbf{W}_{:,r,1})$
6. $\mathbf{E} = \frac{1}{R} \sum_r \big(\mathbf{E}^{(r)}\big)$