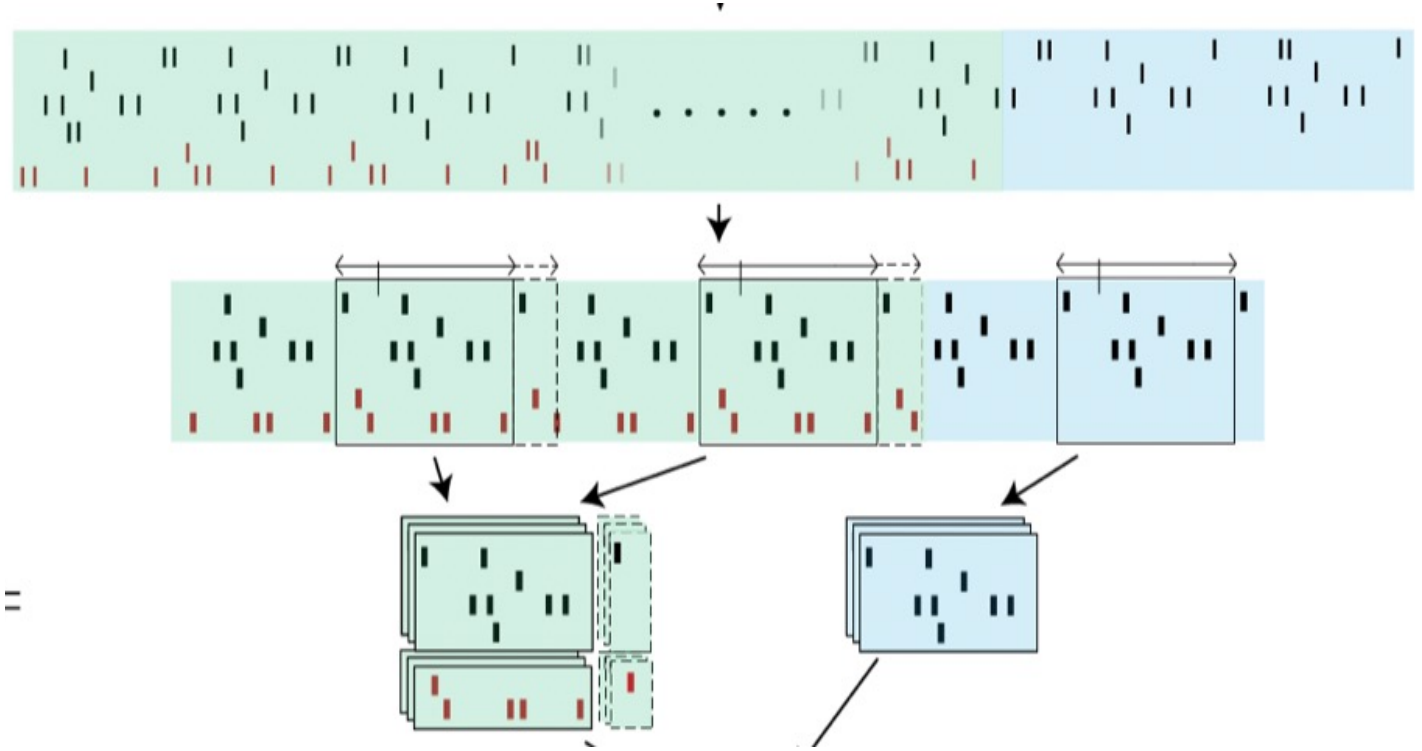
ECE695GM Project
Variational RNN for Joint Neural Latent Generation and Prediction

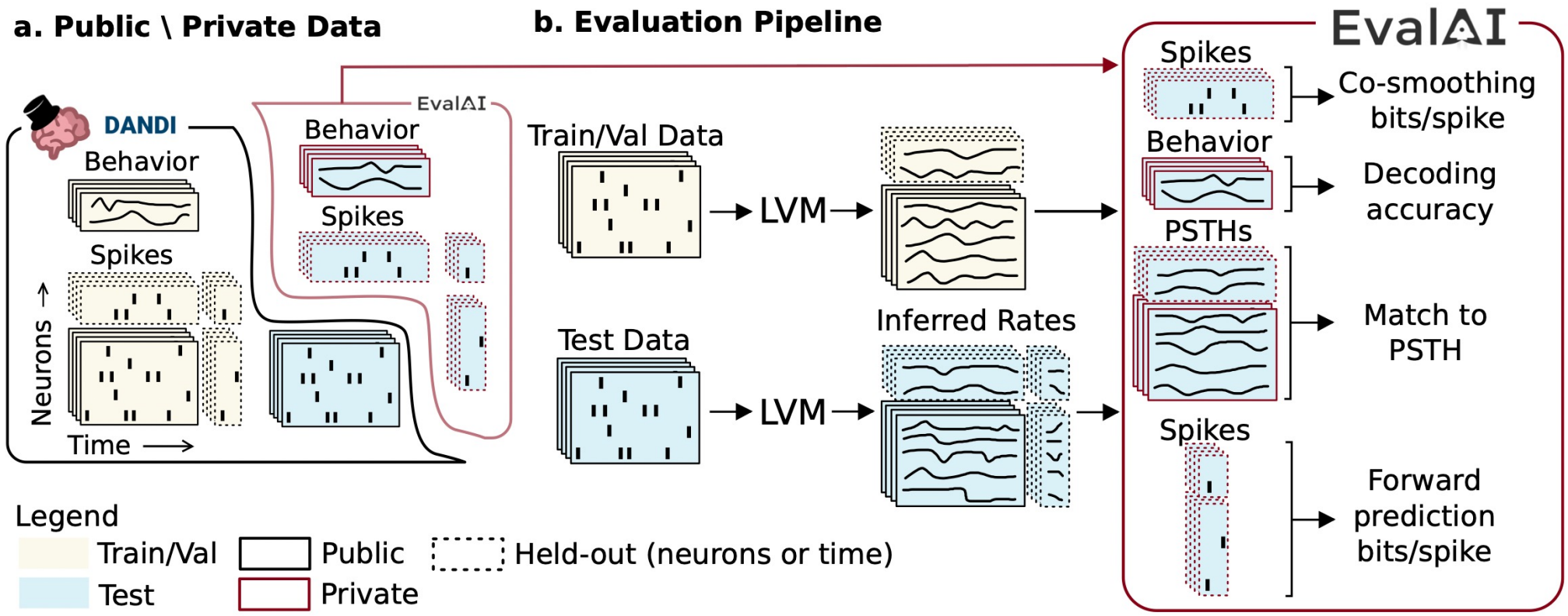
Akshita Kamsali

Data

- Data is initially in spikes binned at 1ms
- Preprocessing:
  - Resample at 5ms
  - Collect trials, separate held-out, and future predictions, and stack them

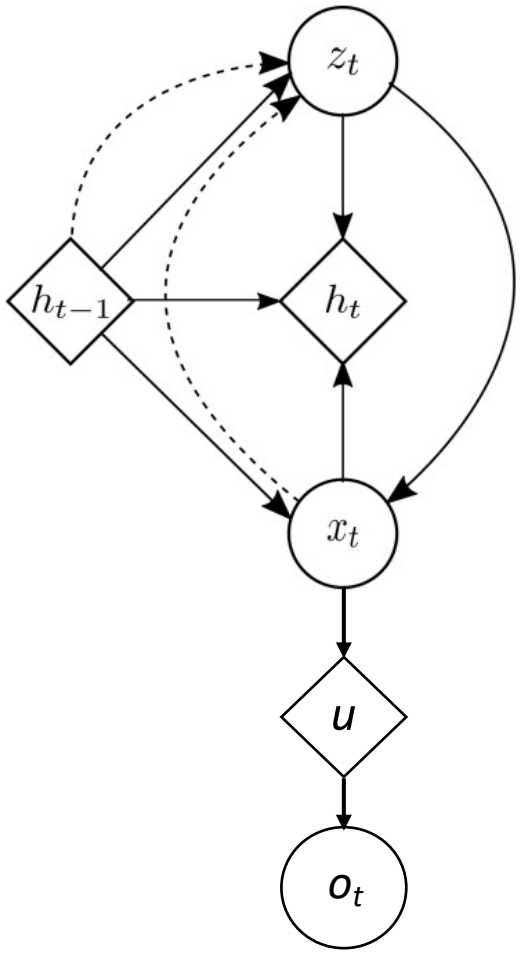


Overview


a. Public \ Private Data
b. Evaluation Pipeline
DANDI
Behavior
Spikes
Neurons
Time
EvalAI
Behavior
Spikes
Train/Val Data
LVM
Test Data
LVM
Inferred Rates
EvalAI
Spikes
Co-smoothing bits/spike
Behavior
Decoding accuracy
PSTHs
Match to PSTH
Spikes
Forward prediction bits/spike
Legend
Train/Val
Test
Public
Private
Held-out (neurons or time)

Proposed Model

Add transform layer and modify loss function objective to VRNN(Chung et. al. 2016)



Prior

$z_t \sim N(\mu_{0,t}, \text{diag}(\sigma^2_{0,t}))$

$x_t \mid z_t \sim N(\mu_{x,t}, \text{diag}(\sigma^2_{x,t}))$

Recurrence

$h_t = f_\theta(\varphi^x_\tau(x_t), \varphi^z_\tau(z_t), h_{t-1})$,

Posterior Inference

$z_t \mid x_t \sim N(\mu_{z,t}, \text{diag}(\sigma^2_{z,t}))$

Transform

$u_t = u * x_t$

Output

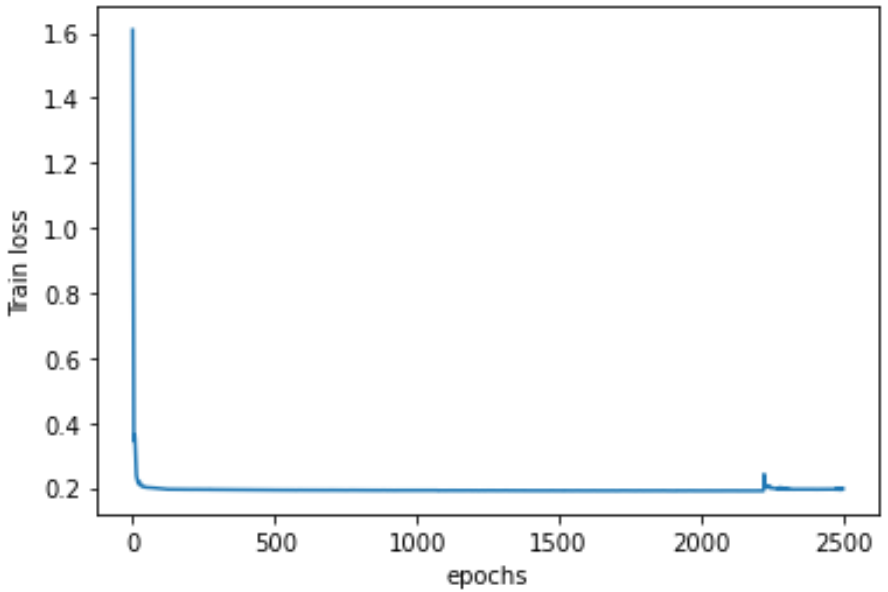
$o_t = \exp(u_t)$

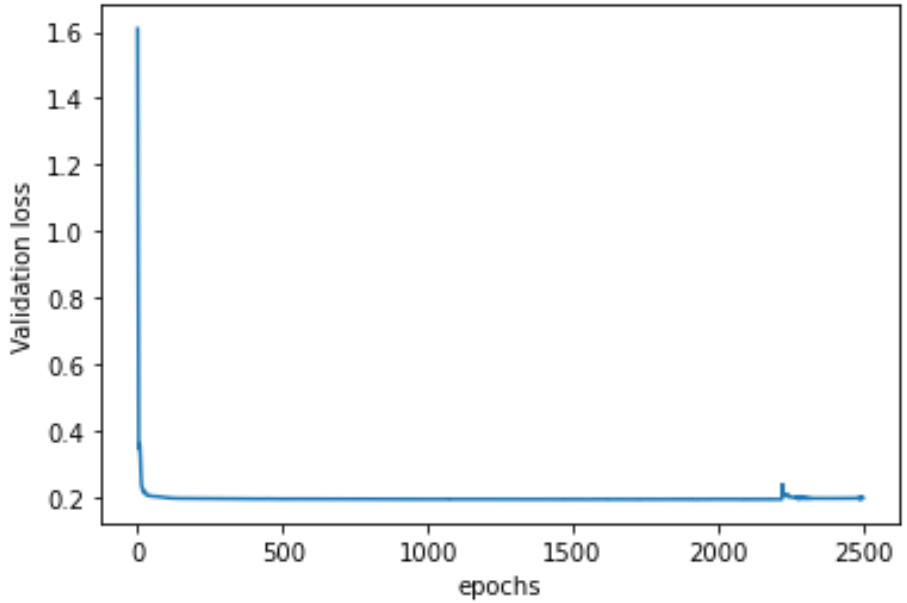
Loss

$\text{poisson\_nll}(o_t, y_t) + E_q[\sum_t - \text{KLD}(q(z_t \mid x_{\le t}, z_{<t}) \,||\, p(z_t \mid x_{<t}, z_{<t})) + \log p(x_t \mid x_{\le t}, z_{<t})]$

Variational ELBO

Loss plots


Train loss
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0
500
1000
1500
2000
2500
epochs


Validation loss
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0
500
1000
1500
2000
2500
epochs

Evaluation criteria

- Co-smoothing bits/spike -
- Decoding accuracy
- Match to PSTH
- Forward prediction bits/spike

$$\text{bits/spike} = \frac{1}{n_{sp} \log 2} (\mathcal{L}(\boldsymbol{\lambda}; \hat{\boldsymbol{y}}_{n,t}) - \mathcal{L}(\bar{\boldsymbol{\lambda}}_{n,:}; \hat{\boldsymbol{y}}_{n,t}))$$

Results

All results are reported on test set of MC_Maze Large dataset

Metric	Epochs = 250	Epochs = 2500
co-bps	0.0176	0.0751
vel-R2	0.0140	0.0258
psth-R2	0.0341	0.1576
fp-bps	-0.0260	-0.0304

Conclusion and Future Work

- Need models to capture the inherent measurement noise
- Having a Variational RNN helps model both the inherent noise and sequential information
- Using more complex encoders and decoders (hierarchical), priors and posteriors
- Train for longer