Supplementary Materials to ‘Uncovering the Spatiotemporal Patterns of Collective Social Activity’

Martin Jankowiak[*] Manuel Gomez-Rodriguez[†]

1 Details on Inference Algorithm

In this section we describe how Sequential Monte Carlo can be used to infer latent patterns from the observed spatiotemporal and content data. The posterior distribution $p(s_{1:n}|t_{1:n}, \mathbf{d}_{1:n}, \mathbf{r}_{1:n})$ is sequentially approximated from $n = 1$ to $n = N$ with a set of $|\mathcal{P}|$ particles that are sampled from a proposal distribution that factorizes as

$$\begin{aligned} q_n(s_{\leq n}|t_{\leq n}, \mathbf{d}_{\leq n}, \mathbf{r}_{\leq n}) &= q_n(s_n|s_{<n}, t_{\leq n}, \mathbf{d}_{\leq n}, \mathbf{r}_{\leq n}) \\ &\times q_{n-1}(s_{<n}|t_{<n}, \mathbf{d}_{<n}, \mathbf{r}_{<n}) \end{aligned}$$

where $q_n(s_n|s_{<n}, t_{\leq n}, \mathbf{d}_{\leq n}, \mathbf{r}_{\leq n})$ is given by

(1.1)
$$\frac{p(s_n|s_{<n},t_{\leq n})p(\mathbf{d}_n|s_{\leq n},\mathbf{d}_{<n})p(\mathbf{r}_n|s_{\leq n},\mathbf{r}_{<n})}{\sum_{s_n} p(s_n|s_{<n},t_{\leq n})p(\mathbf{d}_n|s_{\leq n},\mathbf{d}_{<n})p(\mathbf{r}_n|s_{\leq n},\mathbf{r}_{<n})}$$

In the above expression, the distribution $p(s_n|s_{<n}, t_{\leq n})$ is given by

$$p(s_n|s_{<n}, t_{\leq n}) = \frac{\lambda_{s_n}(t_n)}{\lambda_0 + \sum_{i=1}^{n-1} \gamma_{s_i}(t_n, t_i)} \tag{1.2}$$

where the numerator $\lambda_{s_n}(t_n)$ is equal to λ_0 when s_n is a new spatiotemporal pattern.

We can exploit the conjugacy between the multinomial and the Dirichlet distributions as well as the conjugacy between the normal distribution and normal-gamma prior to integrate out the word distributions $\boldsymbol{\theta}_s$ and spatial parameters $\{\sigma_s, \mathbf{R}_s\}$, respectively, and obtain the marginal likelihoods:

$$p(\mathbf{d}_n|s_{\leq n}, \mathbf{d}_{<n}) = \frac{\Gamma(C^{s_n/\mathbf{d}_n}+V\theta_0)\prod_v^V \Gamma(C_v^{s_n/\mathbf{d}_n}+C_v^{\mathbf{d}_n}+\theta_0)}{\Gamma(C^{s_n/\mathbf{d}_n}+C^{\mathbf{d}_n}+V\theta_0)\prod_v^V \Gamma(C_v^{s_n/\mathbf{d}_n}+\theta_0)}$$

where V is the size of the observed vocabulary, $C^{s_n/\mathbf{d}_n}$ is the total number of words in the spatiotemporal pattern s_n seen so far excluding $\mathbf{d}_n$, $C_v^{s_n/\mathbf{d}_n}$ is the total count for word v in spatiotemporal pattern s_n so far excluding $\mathbf{d}_n$, $C_v^{\mathbf{d}_n}$ is the total count for word v in $\mathbf{d}_n$ and $C^{\mathbf{d}_n}$ is the total word count for $\mathbf{d}_n$; and,

$$p(\mathbf{r}_n|s_{\leq n}, \mathbf{r}_{<n}) = \begin{cases} \frac{N_{s_n}^2}{2\pi(1+N_{s_n})} \frac{\xi_{s_n}^{-1}}{[1+\Delta(\mathbf{r}_n)/\xi_{s_n}]^{1+N_{s_n}}} & \text{if } N_{s_n} \geq 1 \\ 1 & \text{if } N_{s_n} = 0 \end{cases}$$

where $N_{s_n} = \sum_{i=1}^{n-1} \mathbb{I}[s_i = s_n]$ is the number of posts assigned to spatiotemporal pattern s_n, ξ_{s_n} is given by[1]

$$\xi_{s_n} = \beta_{\text{space}} + \tfrac{1}{2}\sum_{i=1}^{n-1} \mathbf{r}_i^2 \mathbb{I}[s_i = s_n] - \tfrac{1}{2N_{s_n}} \left(\sum_{i=1}^{n-1} \mathbf{r}_i \mathbb{I}[s_i = s_n]\right)^2$$

and

$$\Delta(\mathbf{r}_n) = \frac{N_{s_n}}{2(N_{s_n}+1)} \left(\mathbf{r}_n - \frac{1}{N_{s_n}} \sum_{i=1}^{n-1} \mathbf{r}_i \mathbb{I}[s_i = s_n]\right)^2$$

This choice of $q_n(\cdot)$ results in the incremental importance weight

(1.3)
$$\alpha_n(s_{<n}) = p(t_n|s_{<n}, t_{<n}) Q_n(s_{<n}, t_{\leq n}, \mathbf{d}_{\leq n}, \mathbf{r}_{\leq n})$$

where $Q_n(s_{<n}, t_{\leq n}, \mathbf{d}_{\leq n}, \mathbf{r}_{\leq n})$ is given by

$$\sum_{s_n} p(s_n|s_{<n}, t_{\leq n}) p(\mathbf{d}_n|s_{\leq n}, \mathbf{d}_{<n}) p(\mathbf{r}_n|s_{\leq n}, \mathbf{r}_{<n}) \tag{1.4}$$

This update is optimal in the sense that it leads to minimum variance among the particle weights. Finally note that in order to mitigate against particle degeneracy systematic resampling is used whenever the particle system satisfies $\|\mathbf{w}_n\|_2^{-2} < \kappa_{\text{thresh}}|\mathcal{P}|$ (throughout we use $\kappa_{\text{thresh}} = 0.9$). For more details on Sequential Importance Resampling see e.g. ref. [2].

1.1 Time kernel inference

If the time kernels parameters, $\{\alpha_s, \tau_s\}$, are fixed the inference procedure described above yields an unbiased estimate of the posterior $p(s_{1:N}|t_{1:N}, \mathbf{d}_{1:N}, \mathbf{r}_{1:N})$. In general, however, these parameters are unknown and need to be estimated. Methods for calculating the full posterior $p(s_{1:N}, \{\alpha_s, \tau_s\}|t_{1:N}, \mathbf{d}_{1:N}, \mathbf{r}_{1:N})$ can be derived; however, they are computationally expensive and do not

[*] Center for Urban Science and Progress, New York University, jankowiak@gmail.com

[†] Max Planck Institute for Software Systems, manuelgr@mpi-sws.org

[1] Up to a factor of $\frac{1}{2}N_{s_n}$ this is the spatial variance of pattern s_n when $\beta_{\text{space}} \to 0$.

Algorithm 1 Inference algorithm for the SDHP

```
Initialize $w_1^{(p)} \to 1/|\mathcal{P}|$ and $S^{(p)} \to 0$ for all $p \in \mathcal{P}$.
for $n = 1, \ldots, N$ do
    for $p \in \mathcal{P}$ do
        Draw $s_n^{(p)}$ from Eqn. 1.1.
        if $s_n^{(p)} = S^{(p)} + 1$ then
            Draw the time kernel parameters $\{\alpha_s, \tau_s\}$ for $s = s_n^{(p)}$ from the prior
            Increase the number of patterns $S^{(p)} \to S^{(p)} + 1$
        Update the particle weight $\mathrm{w}_n^{(p)}$ using Eqn. 1.3
        Update $\{\alpha_s, \tau_s\}$ for all patterns via Eqn. 1.6
    Normalize particle weights.
    if $\|\mathbf{w}_n\|_2^{-2} < \kappa_{\text{thresh}}|\mathcal{P}|$ then
        Resample particles.
Finally return the particle $p \in \mathcal{P}$ with the largest weight as an approximate MAP estimate to Eqn. 1.5.
```

scale to large datasets (since they rely on e.g. expensive MCMC updates). Since our primary interest is not in the posterior itself but rather the MAP estimate, i.e.

$$s_{1:N}^{\text{MAP}} = \arg\max_{s_{1:N}} p(s_{1:N}|t_{1:N}, \mathbf{d}_{1:N}, \mathbf{r}_{1:N}), \tag{1.5}$$

we do not necessarily require SMC to produce unbiased samples from the posterior. Rather, we just need SMC to explore the posterior space efficiently and return an (approximate) MAP estimate. Consequently, we use the following computationally efficient procedure: after each time step, the parameters $\{\alpha_s, \tau_s\}$ are set equal to a (restricted) MLE estimate. More specifically, as part of the model specification we choose a fixed, finite set of allowed time constants, $\Psi_\tau = \{\tau_i\}$. Then at each time step n and for each spatiotemporal pattern s and $\tau_i \in \Psi_\tau$ we compute

$$\alpha_s^{\text{MLE}}(\tau_i) = \arg\max_{\alpha_s} p(\alpha_s|\alpha_{\text{time}}, \beta_{\text{time}}) p(\mathcal{T}_{s;n}|\alpha_s, \tau_i) \tag{1.6}$$

where $\mathcal{T}_{s;n}$ is the sequence of times for the posts assigned to spatiotemporal pattern s through time step n. For each $\tau_i \in \Psi_\tau$ Eqn. 1.6 can be computed in closed form. Finally, we choose the pair $(\alpha_s^{\text{MLE}}(\tau_i), \tau_i)$ that maximizes the likelihood in Eqn. 1.6.[2] In this way the parameters $\{\alpha_s, \tau_s\}$ are updated at each time step for all patterns that contain at least two posts.

2 Setup for synthetic experiments

Unless stated otherwise, the following experimental parameters are common to all four experiments: the vocabulary has length $|\mathcal{V}| = 15$; the hyperparameters for the prior on the self-excitation parameter α_s are given by $\alpha_{\text{time}} = 0.1$ and $\beta_{\text{time}} = 0.2$; the base intensity $\lambda_0 = 10$; the time constants are given by $\Psi_\tau = \{1\}$; the Dirichlet hyperparameter is given by $\theta_0 = 1$; the number of words per tweet is given by $N_{\text{words}} = 7$; and the number of particles used during inference is $|\mathcal{P}| = 4$. The number of tweets in each sample will be denoted as N and the number of trials per value of x will be denoted as N_{trials}.

In order for the spatial part of the generative process to be well-defined, we use a uniform prior on the mean location $\mathbf{R}_s$ of each pattern s, with the prior defined on the unit square.[3] Unless stated otherwise the spatial hyperparameter $\beta_{\text{space}} = 0.01$ and the generative process assigns each spatiotemporal pattern a spatial extent $\sigma_0 = 0.1$.

For the experiment corresponding to Fig. 2a we set $N_{\text{trials}} = 60$ and $\sigma_0 = 0.03$, while for the experiment corresponding to Fig. 2b we set $N = 5500$ and $N_{\text{trials}} = 500$ as well as $N_{\text{words}} = 15$ and $\sigma_0 = 0.02$ (so that even smaller patterns should be readily identifiable). For the experiment corresponding to Fig. 3a we set $N = 500$, $N_{\text{trials}} = 50$, $|\mathcal{P}| = 8$, and $\beta_{\text{space}} = \sigma_0^2$, while for the experiment corresponding to Fig. 3b we set $N = 2000$, $N_{\text{trials}} = 200$, $\sigma_0 = 0.03$, and $|\mathcal{P}| = 1$.

3 Details on Location Prediction Experiment

The two selection criteria used in the paper are defined as follows:

— *Loose selection:* we sort all tweets in ascending order according to the σ_s of the associated pattern, discard any tweet in a pattern with less than 7 tweets, and compute the average root mean square error (RMSE) of the top 4%.

— *Tight selection:* we sort all tweets in ascending order according to the σ_s of the associated pattern, discard any tweet in a pattern with less than 11 tweets, and compute the average root mean square error (RMSE) of the top 4%.

In the above measures, ties are adjudicated by preferring tweets which belong to patterns with more tweets. Any remaining ties are decided randomly.

4 Goodness of Fit Measures

4.1 Spatial measure We use the following spatial goodness of fit measure. At each iteration n of the corresponding inference algorithm (after a burnin period of 500 tweets), we evaluate the marginal likelihood of the next sample $n + 1$ given the parameters and latent

[2] Note that this is not equivalent to simultaneously maximizing over (α_s, τ_s), which cannot be done in closed form.

[3] If a given tweet falls outside the unit square during sampling from the generative process, sampling of the location is repeated until the location falls within the unit square.

variables inferred from the first n samples; e.g. for the SDHP we have:

(4.7)

$$\text{spatial g.o.f.} = \tfrac{1}{2000} \sum_{n=501}^{2500} \log p(\boldsymbol{r}_n | t_{\leq n}, s_{<n}, \boldsymbol{d}_{\leq n}, \boldsymbol{r}_{<n})$$

An analogous expression (i.e. without conditioning on $\boldsymbol{d}_{\leq n}$ and $t_{\leq n}$) holds for the GMM. In order to make a more direct comparison between the two models we setup the GMM as follows: (i) at each iteration n we set the number of gaussian components equal to the number of spatiotemporal patterns inferred by the SDHP at time step $n-1$; and (ii) we consider isotropic gaussians with the minimum covariance set equal to ${\sigma^2}_{\text{min}} = 2\beta_{\text{space}}$.

4.2 Content measure With reference to the expression in Eqn. 4.7, we use a related goodness of fit measure, namely the perplexity $\mathcal{P}$ [1]; e.g. for the DHP we have the following:

$$\mathcal{P} = \exp\left(\tfrac{-1}{N_{\text{words}}} \sum_{n=501}^{2500} \log p(\boldsymbol{d}_n | t_{\leq n}, s_{<n}, \boldsymbol{d}_{<n}) \right)$$

where N_{words} is the total number of words in $\{\boldsymbol{d}_{501}, ..., \boldsymbol{d}_{2500}\}$. An analogous expression (i.e. with additional conditioning on $\boldsymbol{r}_{\leq n}$) holds for the SDHP. In order to make a more direct comparison between the two models we set λ_0 for the DHP such that the number of inferred patterns matches that of the SDHP.

References

[1] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3:993–1022, 2003.

[2] A. Doucet and A. M. Johansen. A tutorial on particle filtering and smoothing: Fifteen years later. *Handbook of Nonlinear Filtering*, 12(656-704):3, 2009.