Efficient Inference for Dynamic Topic Modeling with Large Vocabularies (Supplementary Material)

Federico Tomasi[1] **Mounia Lalmas**[1] **Zhenwen Dai**[1]

[1]Spotify Research

1 IMPORTANCE SAMPLING

We can write the probability of words in a document conditioned on the parameter $\boldsymbol{\eta}_d$ and $\boldsymbol{\beta}$ as:

$$p(W_d|\boldsymbol{\eta}_d, \boldsymbol{\beta}) = \prod_{n=1}^{N_d} \text{Multi}(1, \sigma(\boldsymbol{\xi}_d)) = \prod_{n=1}^{N_d} \text{Cat}(\sigma(\boldsymbol{\xi}_d)) = \tilde{\mathcal{L}}_W. \tag{1}$$

Its derivative can be derived as:

$$\nabla\tilde{\mathcal{L}}_W = \mathbb{E}_{q(\boldsymbol{\beta}|H)q(\boldsymbol{\eta}_d)} \sum_{n=1}^{N_d} [\nabla \log \text{Cat}(\sigma(\boldsymbol{\xi}_d))] \tag{2}$$

$$= \mathbb{E}_{q(\boldsymbol{\beta}|H)q(\boldsymbol{\eta}_d)} \sum_{n=1}^{N_d} \left[\nabla\boldsymbol{\xi}_{d,n} - \nabla \log \sum_{j=1}^{P} \exp(\boldsymbol{\xi}_d)_j\right] \tag{3}$$

$$= \mathbb{E}_{q(\boldsymbol{\beta}|H)q(\boldsymbol{\eta}_d)} \sum_{n=1}^{N_d} \left[\nabla\boldsymbol{\xi}_{d,n} - \frac{1}{\sum_{j=1}^{P} \exp(\boldsymbol{\xi}_{dj})} \nabla \sum_{i=1}^{P} \exp(\boldsymbol{\xi}_d)_i\right] \tag{4}$$

$$= \mathbb{E}_{q(\boldsymbol{\beta}|H)q(\boldsymbol{\eta}_d)} \sum_{n=1}^{N_d} \left[\nabla\boldsymbol{\xi}_{d,n} - \sum_{i=1}^{P} \frac{\exp(\boldsymbol{\xi}_d)_i}{\sum_{j=1}^{P} \exp(\boldsymbol{\xi}_d)_j} \nabla\boldsymbol{\xi}_{d,i}\right]. \tag{5}$$

To approximate this derivative, we consider a random sample of M words from the vocabulary and use those to approximate the normalisation constant. Consider a sample vector $\boldsymbol{s} \in \{1, ..., P\}^{M+N_d}$, which represents a sample of words in the vocabulary and stores the index of the N_d positive (words appearing in document d) and the index of the M sampled words. Let $\xi'_{d,i} := \xi_{d,i} - \ln(Q_{di}/P)$ if $y_i = 0$ (*i.e.*, word i does not appear in document d), $\xi'_{d,i} := \xi_{d,i} - \ln(Q_{di})$ otherwise, with Q_{di} proposal distribution. We shift the true logits by the expected number of occurrences of a word i, ensuring that the sampled softmax is asymptotically unbiased. In our experiment we choose Q to be a uniform distribution over the subset of words considered, so $Q_{di} = 1/(N_d + M)$ [Jean et al., 2014]. Then:

$$\nabla\tilde{\mathcal{L}}_W \approx \mathbb{E}_{q(\boldsymbol{\beta}|H)q(\boldsymbol{\eta}_d)} \sum_{n=1}^{N_d} \left[\nabla\boldsymbol{\xi}_{d,n} - \sum_{i=1}^{M+N_d} \frac{\exp(\xi'_i)}{\sum_{j=1}^{M+N_d} \exp(\xi'_j)} \nabla\xi_{d,i}\right]. \tag{6}$$

2 FITC

The FITC approximation for the multi-output Gaussian process results into the follow formulation:

$$p_{\text{FITC}}(\beta|U, Z_X, Z_H, X, H)$$
$$= \mathcal{N}(\beta|K_{fu}K_{uu}^{-1}(U^\top)_:, \text{diag}\left(K_{ff} - K_{fu}K_{uu}^{-1}K_{uf}\right)),$$

Accepted for the 38th Conference on Uncertainty in Artificial Intelligence (UAI 2022).

where diag $(\cdot)$ returns a diagonal matrix while keeping the diagonal entries, and $A_{:}$ denotes $vec(A)$, the column-wise vectorisation of the matrix A. Since K_{fu}, K_{ff} and K_{uu} have a Kronecker structure, we can rewrite mean and covariance to compute them efficiently as follows

$$K_{fu}K_{uu}^{-1}(U^\top)_{:} = (K_{fu}^X {K_{uu}^X}^{-1} U^\top {K_{uu}^H}^{-\top} {K_{fu}^H}^\top)_{:}$$

$$\begin{aligned}
&\text{diag}\left(K_{ff} - K_{fu}K_{uu}^{-1}K_{uf}\right) \\
&= \text{diag}\left(K_{ff}\right) \\
&- \left(\text{diag}\left(K_{fu}^H {K_{uu}^H}^{-1} {K_{fu}^H}^\top\right) \otimes \text{diag}\left(K_{fu}^X {K_{uu}^X}^{-1} {K_{fu}^X}^\top\right)\right)_{:}
\end{aligned}$$

Note that the last line becomes a vectorised outer product between vectors and solved efficiently. We can use the same trick for diag (K_{ff}).

The full derivation is the following:

$$\begin{aligned}
&K_{fu}K_{uu}^{-1}(U^\top)_{:} \\
&= (K_{fu}^H \otimes K_{fu}^X)(K_{uu}^H \otimes K_{uu}^X)^{-1}(U^\top)_{:} \\
&= (K_{fu}^H \otimes K_{fu}^X)({K_{uu}^H}^{-1} \otimes {K_{uu}^X}^{-1})(U^\top)_{:} \\
&= (K_{fu}^H {K_{uu}^H}^{-1} \otimes K_{fu}^X {K_{uu}^X}^{-1})(U^\top)_{:} \quad \text{(matrix eq)} \\
&= (K_{fu}^X {K_{uu}^X}^{-1} U^\top((K_{fu}^H {K_{uu}^H}^{-1})^\top)_{:} \\
&= (K_{fu}^X {K_{uu}^X}^{-1} U^\top {K_{uu}^H}^{-\top} {K_{fu}^H}^\top)_{:}
\end{aligned}$$

$$\begin{aligned}
&\text{diag}\left(K_{ff} - K_{fu}K_{uu}^{-1}K_{uf}\right) \\
&= \text{diag}\left(K_{ff}\right) - \text{diag}\left(K_{fu}K_{uu}^{-1}K_{uf}\right) \\
&= \text{diag}\left(K_{ff}\right) - \text{diag}\left((K_{fu}^H {K_{uu}^H}^{-1} \otimes K_{fu}^X {K_{uu}^X}^{-1})(K_{fu}^H \otimes K_{fu}^X)^\top\right) \\
&= \text{diag}\left(K_{ff}\right) - \text{diag}\left(K_{fu}^H {K_{uu}^H}^{-1} {K_{fu}^H}^\top \otimes K_{fu}^X {K_{uu}^X}^{-1} {K_{fu}^X}^\top\right) \\
&= \text{diag}\left(K_{ff}\right) - \left(\text{diag}\left(K_{fu}^H {K_{uu}^H}^{-1} {K_{fu}^H}^\top\right) \otimes \text{diag}\left(K_{fu}^X {K_{uu}^X}^{-1} {K_{fu}^X}^\top\right)\right)_{:}
\end{aligned}$$

3 MATRIX NORMAL DISTRIBUTION

The matrix normal is related to the multivariate normal distribution in the following way:

$$\mathbf{X} \sim \mathcal{MN}_{n\times p}(\mathbf{M}, \mathbf{U}, \mathbf{V}), \tag{7}$$

if and only if

$$\text{vec}(\mathbf{X}) \sim \mathcal{N}_{np}(\text{vec}(\mathbf{M}), \mathbf{V} \otimes \mathbf{U}) \tag{8}$$

where $\otimes$ denotes the Kronecker product and $\text{vec}(\mathbf{M})$ denotes the vectorization of $\mathbf{M}$.

Sampling from the distribution and the KL divergence can be computed efficiently. $U_{\boldsymbol{\beta}_k}$ can be sampled efficiently following the procedure: *(i)* sample $C \sim \mathcal{MN}_{h\times x}(\mathbf{0}, I, I)$, $C \in \mathbb{R}^{h\times x}$, a collection of independent samples from a standard normal distribution; then *(ii)* let $U_{\boldsymbol{\beta}_k} = (M + ACB)_{:}$, where $\Sigma^H = AA^\top$ and $\Sigma^{\boldsymbol{x}} = B^\top B$. The KL divergence between $q(U_{\boldsymbol{\beta}_k})$ and $p(U_{\boldsymbol{\beta}_k})$ can also be computed efficiently (see Supplementary Material).

Sampling from the matrix normal distribution is a special case of the sampling procedure for the multivariate normal distribution. Let $\mathbf{X}$ be an n by p matrix of np independent samples from the standard normal distribution, so that

$$\mathbf{X} \sim \mathcal{MN}_{n\times p}(\mathbf{0}, \mathbf{I}, \mathbf{I}). \tag{9}$$

Then let

$\mathbf{Y} = \mathbf{M} + \mathbf{AXB}$, so that

$$\mathbf{Y} \sim \mathcal{MN}_{n\times p}(\mathbf{M}, \mathbf{AA}^T, \mathbf{B}^T\mathbf{B}), \tag{10}$$

where A and B can be chosen by Cholesky decomposition or a similar matrix square root operation.

3.1 KL DIVERGENCE

The KL divergence between two matrix-variate normal distributions, *e.g.*, $q(U_{\boldsymbol{\beta}_k})$ and $p(U_{\boldsymbol{\beta}_k})$, can be analytically computed as:

$$\begin{aligned}\mathrm{KL}(q(U_{\boldsymbol{\beta}_k})||p(U_{\boldsymbol{\beta}_k})) = \frac{1}{2}\Bigg(& M_x \log\frac{|K^H_{uu}|}{|\Sigma^H|} + M_H \log\frac{|K^{\boldsymbol{x}}_{uu}|}{|\Sigma^{\boldsymbol{x}}|} \\ & + \mathrm{tr}(M^\top (K^{\boldsymbol{x}}_{uu})^{-1} M (K^H_{uu})^{-1}) + \mathrm{tr}((K^H_{uu})^{-1}\Sigma^H)\,\mathrm{tr}((K^{\boldsymbol{x}}_{uu})^{-1}\Sigma^{\boldsymbol{x}}) - M_H M_x\Bigg).\end{aligned}$$

To implement $\mathrm{tr}[M^\top (K^{\boldsymbol{x}})^{-1} M (K^H)^{-1}]$, we use $K^{\boldsymbol{x}} = L_{\boldsymbol{x}} L_{\boldsymbol{x}}^\top$, $K^H = L_H L_H^\top$, $A = L_{\boldsymbol{x}}^{-1} M L_H^{-\top}$, then $\mathrm{tr}[M^\top (K^{\boldsymbol{x}})^{-1} M (K^H)^{-1}] = \mathrm{tr}(A^\top A)$.

First, recall that

$$KL(q||p) = \int q(x)(\log q(x) - \log p(x))dx. \tag{11}$$

which in the case of two multivariate Gaussian distributions, say $p(x) = \mathcal{N}(m_1, S_1)$, $q(x) = \mathcal{N}(m_2, S_2)$ is equal to

$$\int [\frac{1}{2}\log\frac{|S_2|}{|S_1|} - \frac{1}{2}(x-m_1)^\top S_1^{-1}(x-m_1) + \frac{1}{2}(x-m_2)^\top S_2^{-1}(x-m_2)]q(x)dx \tag{12}$$

$$= \frac{1}{2}\log\frac{|S_2|}{|S_1|} - \frac{1}{2}tr\{\mathbb{E}[(x-m_1)(x-m_1)^\top]S_1^{-1}\} + \frac{1}{2}\mathbb{E}[(x-m_2)^\top S_2^{-1}(x-m_2)] \tag{13}$$

$$= \frac{1}{2}\log\frac{|S_2|}{|S_1|} - \frac{1}{2}tr\{\mathbb{E}[(x-m_1)(x-m_1)^\top]S_1^{-1}\} + \frac{1}{2}\mathbb{E}[(x-m_2)^\top S_2^{-1}(x-m_2)] \tag{14}$$

$$= \frac{1}{2}\log\frac{|S_2|}{|S_1|} - \frac{1}{2}tr\{I_d\} + \frac{1}{2}(m_1-m_2)^\top S_2^{-1}(m_1-m_2) + \frac{1}{2}tr\{S_2^{-1}S_1\} \tag{15}$$

$$= \frac{1}{2}[\log\frac{|S_2|}{|S_1|} - d + tr\{S_2^{-1}S_1\} + (m_1-m_2)^\top S_2^{-1}(m_1-m_2)] \tag{16}$$

Now, we can use a Kronecker representation of S_1 and S_2 as $S_1 = S_h \otimes S_x$ and $S_2 = K_h \otimes K_x$. Let $M = m_1 - m_2$. Also, we consider a vectorised version of M, and we indicate it as $M_:$. Then the KL divergence becomes: (using $|V \otimes U| = |V|^n |U|^p$, and mixed product property of Kron)

$$\frac{1}{2}[\log\frac{|K_h \otimes K_x|}{|S_h \otimes S_x|} - d + tr\{(K_h \otimes K_x)^{-1}(S_h \otimes S_x)\} + M_:^\top (K_h \otimes K_x)^{-1} M_:] \tag{17}$$

$$= \frac{1}{2}[n\log\frac{|K_h|}{|S_h|} + p\log\frac{|K_x|}{|S_x|} - np + tr\{(K_h^{-1} \otimes K_x^{-1})(S_h \otimes S_x)\} + M_:^\top (K_h^{-1} \otimes K_x^{-1}) M_:] \tag{18}$$

$$= \frac{1}{2}[n\log\frac{|K_h|}{|S_h|} + p\log\frac{|K_x|}{|S_x|} - np + tr\{(K_h^{-1}S_h) \otimes (K_x^{-1}S_x)\} + M_:^\top ((K_h^{-1} \otimes K_x^{-1}) M_:)] \quad (associative) \tag{19}$$

$$= \frac{1}{2}[n\log\frac{|K_h|}{|S_h|} + p\log\frac{|K_x|}{|S_x|} - np + tr(K_h^{-1}S_h)tr(K_x^{-1}S_x) + M_:^\top (K_x^{-1} M K_h^{-1})_:] \quad (kronmatrixequations) \tag{20}$$

$$= \frac{1}{2}[n\log\frac{|K_h|}{|S_h|} + p\log\frac{|K_x|}{|S_x|} - np + tr(K_h^{-1}S_h)tr(K_x^{-1}S_x) + tr[M^\top K_x^{-1} M K_h^{-1}]] \tag{21}$$

4 VARIATIONAL INFERENCE FOR GAUSSIAN AND WISHART PROCESS

Inference for $\boldsymbol{\mu}$. We first augment the Gaussian process with a set of auxiliary variables with a set of corresponding time stamps, *i.e.*,

$$p(\boldsymbol{\mu}|\boldsymbol{x}) = \int p(\boldsymbol{\mu}|U_\mu, \boldsymbol{x}, \boldsymbol{z}_\mu)p(U_\mu|\boldsymbol{z}_\mu)\mathrm{d}U_\mu, \tag{22}$$

where U_μ is the auxiliary variable for $\boldsymbol{\mu}$ and $\boldsymbol{z}_\mu$ is the corresponding index. Both $p(\boldsymbol{\mu}|U_\mu, \boldsymbol{x}, \boldsymbol{z}_\mu)$ and $p(U_\mu|\boldsymbol{z}_\mu)$ follow the same Gaussian processes as the one for $p(\boldsymbol{\mu}|\boldsymbol{x})$, *i.e.*, these Gaussian processes have the same mean and kernel functions. As shown in Equation (22), the above augmentation does not change the prior distributions for $\boldsymbol{\mu}$.

The variational posterior of $\boldsymbol{\mu}$ is constructed in a special form to enable efficient inference [Titsias, 2009]: $q(\boldsymbol{\mu}, U_\mu) = p(\boldsymbol{\mu}|U_\mu)q(U_\mu)$. $q(U_\mu) = \mathcal{N}(M_\mu, S_\mu)$ is a multivariate normal distribution, in which the mean and covariance are variational parameters. $p(\boldsymbol{\mu}|U_\mu)$ is a conditional Gaussian process [Hensman et al., 2013]. When $\boldsymbol{\mu}$ is used in the down-stream distributions, a lower bound can be derived,

$$\log p(\cdot|\boldsymbol{\mu}) \geq \mathbb{E}_{q(\boldsymbol{\mu})}[p(\cdot|\boldsymbol{\mu})] - \mathrm{KL}\left(q(U_\mu)||p(U_\mu)\right), \tag{23}$$

where $q(\boldsymbol{\mu}) = \int p(\boldsymbol{\mu}|U_\mu)q(U_\mu)\mathrm{d}U_\mu$.

Inference for Σ. We derive a similar stochastic variational inference method for the Wishart Process. We augment each GP $p(\boldsymbol{f}_{ij}|\boldsymbol{x})$ in the Wishart process with a set of auxiliary variables and a set of the corresponding inputs,

$$p(\boldsymbol{f}_{ij}|\boldsymbol{x}) = \int p(\boldsymbol{f}_{ij}|\boldsymbol{u}_{ij}, \boldsymbol{x}, \boldsymbol{z}_{ij})p(\boldsymbol{u}_{ij}|\boldsymbol{z}_{ij})\mathrm{d}\boldsymbol{u}_{ij}, \tag{24}$$

where $\boldsymbol{u}_{ij}$ is the auxiliary variable, $\boldsymbol{z}_{ij}$ is the corresponding inputs and $p(\boldsymbol{f}_{ij}|\boldsymbol{u}_{ij})$ is a conditional Gaussian process [Hensman et al., 2013]. We define the variational posterior of $\boldsymbol{f}_{ij}$ to be $q(\boldsymbol{f}_{ij}, \boldsymbol{u}_{ij}) = p(\boldsymbol{f}_{ij}|\boldsymbol{u}_{ij})q(\boldsymbol{u}_{ij})$, where $q(\boldsymbol{u}_{ij}) = \mathcal{N}(\boldsymbol{m}_{ij}, \boldsymbol{s}_{ij})$. We also define the variational posterior of $\boldsymbol{\ell}$ to be $q(\boldsymbol{\ell}) = \mathcal{N}(\boldsymbol{m}_\ell, S_\ell)$, where S_ℓ is a diagonal matrix. As the diagonal elements of L needs to be positive, we apply a change of variable to the variational posterior of the diagonal elements, *i.e.*, $\ell_m = \log(1 + \exp(\hat{\ell}_m))$, $q(\hat{\ell}_m) = \mathcal{N}(\boldsymbol{m}_{\ell_m}, S_{\ell_m})$. Note that $\boldsymbol{z}_\mu$ and $\boldsymbol{z}_{ij}$ are variational parameters instead of random variables. For this reason, we will omit them from the notation for convenience.

We can derive a variational lower bound with such a set of variational posterior for all the entries $\{\boldsymbol{f}_{ij}\}$ and $\boldsymbol{\ell}$, when Σ is used for some down-stream distributions,

$$\log p(\cdot|\Sigma) \geq \mathbb{E}_{q(F)q(\boldsymbol{\ell})}[p(\cdot|\Sigma)] - \sum_{i,j} \mathrm{KL}\left(q(\boldsymbol{u}_{ij})||p(\boldsymbol{u}_{ij})\right) - \mathrm{KL}\left(q(\boldsymbol{\ell})||p(\boldsymbol{\ell})\right), \tag{25}$$

where $q(F) = \prod_{ij} \int p(\boldsymbol{f}_{ij}|\boldsymbol{u}_{ij})q(\boldsymbol{u}_{ij})\mathrm{d}\boldsymbol{u}_{ij}$.

4.1 LOWER BOUND FOR MIST

After deriving the variational lower bounds for the individual components of MIST, we assemble these components together to form the final variational lower bound. The word distributions for individual topics are used in defining the distribution of individual words for each document d, $p(W_d|\boldsymbol{\eta}_d, \boldsymbol{\beta}^{(\boldsymbol{x}_d)})$. Combining the lower bounds (10), (4), (8), (23) and (25), we can derive the complete variational lower bound $\mathcal{L}$ of MIST.

$$\begin{aligned}\log\ p(W) \geq\ & \mathbb{E}_{q(\boldsymbol{\mu})q(\boldsymbol{\ell})q(F)q(\boldsymbol{\beta})}\left[\mathcal{L}_W\right] - \mathrm{KL}\left(q(U_\beta)||p(U_\beta)\right) - \mathrm{KL}\left(q(H)||p(H)\right) \\ & - \mathrm{KL}\left(q(U_\mu)||p(U_\mu)\right) - \mathrm{KL}\left(q(\boldsymbol{\ell})||p(\boldsymbol{\ell})\right) - \sum_{i,j} \mathrm{KL}\left(q(\boldsymbol{u}_{ij})||p(\boldsymbol{u}_{ij})\right) = \mathcal{L}.\end{aligned}$$

The first term of $\mathcal{L}$ can be further decomposed by plugging in (3),

$$\begin{aligned}&\mathbb{E}_{q(\boldsymbol{\mu})q(\boldsymbol{\ell})q(F)q(\boldsymbol{\beta})}\left[\mathcal{L}_W\right] = \\ &\sum_{d=1}^{D}\left(\mathbb{E}_{q(\boldsymbol{\eta}_d)q(\boldsymbol{\beta})}\left[\log p(W_d|\boldsymbol{\eta}_d, \boldsymbol{\beta}^{(\boldsymbol{x}_d)})\right] - \mathbb{E}_{q(\boldsymbol{\eta}_d)q(\boldsymbol{\mu}_{\boldsymbol{x}_d})q(\Sigma_{\boldsymbol{x}_d})}\left[\mathrm{KL}\left(q(\boldsymbol{\eta}_d)||p(\boldsymbol{\eta}_d|\boldsymbol{\mu}_{\boldsymbol{x}_d}, \Sigma_{\boldsymbol{x}_d})\right)\right]\right).\end{aligned}$$

Note that all variational parameters of $q(\boldsymbol{\mu})$, $q(\boldsymbol{\ell})$, $q(F)$, $q(\boldsymbol{\beta})$, $q(\boldsymbol{\eta})$ are optimised.

5 DATASETS

We include a complete list of details for the dataset we used in our analysis.

We considered the following datasets: State of the Union corpus (*SotU*), department of justice press releases (*DoJ*), Elsevier corpus (*Abstracts*) [Kershaw and Koeling, 2020], Blog Authorship Corpus (*Blogs*) [Schler et al., 2006], NeurIPS conference papers (*NeurIPS*) [Perrone et al., 2017], A Million News Headlines (*News*), Twitter sentiment classification (*Twitter*) [Go et al., 2009].

For each dataset, we consider the total indicated number of samples (if not otherwise specified), and divide the dataset into 75% for training and rest for test.

Blog Authorship Corpus [Schler et al., 2006]. The corpus[1] consists of the posts of 19k bloggers gathered from `blogger.com` from June 1999 to August 2004. The corpus incorporates a total of 681k posts, from which we draw a random sample of 5649 for training and 5650 for testing. After our preprocessing, we considered 3000 words in our vocabulary. License: free use for non-commercial research purposes.

State of the Union corpus (1790-2018). The dataset[2] includes a yearly address of the US president, from 1790 to 2018 (229 years). Our vocabulary includes 4583 words after preprocessing. We split the data into 170 documents as training and 57 documents as test data. License: CC BY-SA 4.0

NeurIPS conference papers (1987-2015) [Perrone et al., 2017]. The dataset[3] includes 5804 conference papers from 1987 to 2015 including an average of 34 papers per year. We preprocessed the dataset leading to 4799 words. In both cases we used 4237 documents as training data and 1567 as test data.

Department of justice press releases (2009-2018). The dataset[4] includes 13087 press releases from the Department of Justice from 2009 to 2018 (115 unique timestamps), preprocessed to include 9591 unique words. Documents were split into 9674 for training and for 3413 testing. License: CC0: Public Domain

Elsevier OA CC-BY Corpus [Kershaw and Koeling, 2020]. The dataset[5] includes 40k open access (OA) CC-BY abstracts taken from articles from across Elsevier's journals, published from 2010 to 2019. After our preprocessing, we considered 13126 words in the vocabulary. License: CC BY 4.0

A Million News Headlines. The dataset[6] includes 1.2M news headlines published over a period of 17 Years (from 2003 to 2019). We took a random sample of 1M. After our preprocessing, we considered a vocabulary of size 22459. License: CC0: Public Domain

Twitter sentiment classification [Go et al., 2009]. The dataset[7] contains 1.6M tweets, from April to May 2009. We randomly sampled 1M tweets. We preprocessed samples using a tweet tokenizer, removing usernames and replacing repeated character sequences (length 3 or more) with sequences of length 3 [Bird et al., 2009]. After our preprocessing we considered 83582 tokens.

5.1 EXPERIMENT SETTINGS.

We split each dataset considering 75% of the samples as training and 25% as test. Documents associated with the same time stamps were assigned to the same split.

For each dynamic topic model we used a Matérn 3/2 kernel for $\boldsymbol{\beta}$, to allow topics to quickly incorporate new words. This is important especially to incorporate neologisms, and particularly for datasets such as NeurIPS conference papers and

[1] `https://u.cs.biu.ac.il/~koppel/BlogCorpus.htm`
[2] `https://kaggle.com/rtatman/state-of-the-union-corpus-1989-2017`
[3] `https://archive.ics.uci.edu/ml/datasets/NIPS+Conference+Papers+1987-2015`
[4] `https://kaggle.com/jbencina/department-of-justice-20092018-press-releases`
[5] `https://data.mendeley.com/datasets/zm33cdndxs/2`
[6] `https://kaggle.com/therohk/million-headlines`
[7] `https://www.kaggle.com/kazanova/sentiment140`

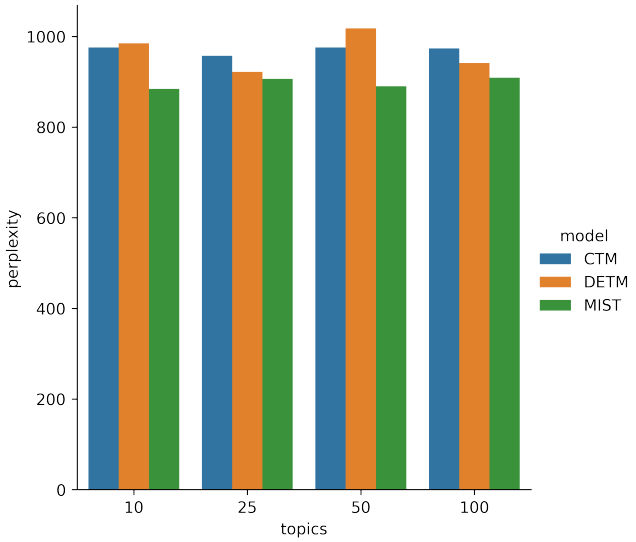

Figure 1: Perplexity at varying the number of topics for the NeurIPS dataset.

Elsevier corpus, where the names of novel models become quoted in citations (for example, "LDA" starting to appear in publications together as "topic modeling" after its introduction in 2003). For the other parameters $\boldsymbol{\mu}$ and $\boldsymbol{f}$ we use a squared exponential kernel, as we expect a smooth temporal evolution of both topic probabilities and their correlation. We initialise amplitude and length scale of kernels as 1 and 0.5 respectively, and we optimise for them using the approximate empirical Bayes approach [Maritz, 2018].

Experiments were conducted using Adam optimiser with learning rate 0.001 and up to 10k epochs until convergence. With our configuration, DCTM took around 6s/epoch to analyse 7000 training samples in 3000 dimensions using a single GPU NVIDIA Tesla V100, completing 5000 epochs in 8 hours (on average). Using MIST we achieved a runtime of $\sim$ 2.5s/epoch, completing 5000 epochs in 3.5 hours. We experimented with different number of topics, and report the results using a default choice of 30 for all datasets (20 for SotU) to maintain consistency with previous works. We also experimented with a different number of inducing points for the three components $\boldsymbol{\beta}$, $\boldsymbol{\mu}$ and $\boldsymbol{f}$, thus controlling the complexity of the variational posterior used from both DCTM and our models (static models such as LDA and CTM do not have such dynamic components). The number of inducing points used for such components is 15, 20 and 15, respectively. MIST has an additional component for the latent embedding of words in $\boldsymbol{\beta}$; we used $M_H = 200$ in $Q = 10$ dimensions. We initialised the posterior for H by transforming the words in our vocabulary using ELMO embeddings [Peters et al., 2018] pre-trained on the 1 Billion Word Benchmark, and take the first Q principal components using a PCA transformation.

For the posterior of $\boldsymbol{\eta}$, when using a static encoder (*e.g.*, for DCTM) we considered a dense neural network with three layers with size 500, 300 and 200, respectively. To account for the increased input dimensionality in our meta-encoder we instead used a dense neural network with three layers, with size 1000, 600 and 400, respectively.[8]

6 ADDITIONAL RESULTS

Varying number of topics. Our analysis does not show substantial differences when varying the number of topics. We experimented with topics varying between 10 and 100 and showed some examples in Figure 1.

References

Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.", 2009.

[8] Since the encoder is a collection of variational parameters, we emphasise that increasing its size does not overfit the model.

Alec Go, Richa Bhayani, and Lei Huang. Twitter sentiment classification using distant supervision. *CS224N project report, Stanford*, 1(12):2009, 2009.

James Hensman, Nicolo Fusi, and Neil D Lawrence. Gaussian processes for big data. In *UAI*, page 282. Citeseer, 2013.

Sébastien Jean, Kyunghyun Cho, Roland Memisevic, and Yoshua Bengio. On using very large target vocabulary for neural machine translation. *arXiv preprint arXiv:1412.2007*, 2014.

Daniel Kershaw and Rob Koeling. Elsevier oa cc-by corpus, 2020.

Johannes S Maritz. *Empirical Bayes methods with applications*. Chapman and Hall/CRC, 2018.

Valerio Perrone, Paul A Jenkins, Dario Spano, and Yee Whye Teh. Poisson random fields for dynamic feature models. *Journal of Machine Learning Research*, 18, 2017.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*, 2018.

J Schler, M Koppel, S Argamon, and J Pennebaker. Effects of age and gender on blogging. aaai spring symposium on computational approaches for analyzing weblogs, 2006.

Michalis Titsias. Variational learning of inducing variables in sparse gaussian processes. In *AISTATS*, pages 567–574, 2009.