Dynamic Topic Models for Temporal Document Networks

Delvin Ce Zhang [1] **Hady W. Lauw** [1]

Abstract

Dynamic topic models explore the time evolution of topics in temporally accumulative corpora. While existing topic models focus on the dynamics of individual documents, we propose two neural topic models aimed at learning unified topic distributions that incorporate both document dynamics and network structure. For the first model, by adding a time dimension, we propose Time-Aware Optimal Transport, which measures the probability of a link between two differently timestamped documents using their semantic distance. Since the gradually evolving topological structure of network may also influence the establishment of a new link, for the second model, we further design a Temporal Point Process to capture the impact of historical neighbors on the current link formation at the network level. Experiments on four dynamic document networks demonstrate the advantage of our models in jointly modeling document dynamics and network adjacency.

1. Introduction

Textual documents represent an important class of data, including academic papers, Web pages, etc. Usually these documents link to one another, forming a network structure, such as paper citation network, Web page hyperlink network. To derive latent semantics from such a corpus, we may employ a topic model, such as RTM (Chang & Blei, 2009) that leverages the interconnection between documents.

A document network does not emerge suddenly in its entirety. Rather, it is an accumulation of documents created over time. The latent themes in a corpus may also evolve over time, e.g., academic papers track the development of research across years, news articles track the chronology of events. Early attempts to capture document dynamics (DTM (Blei & Lafferty, 2006)) ignore the network aspect.

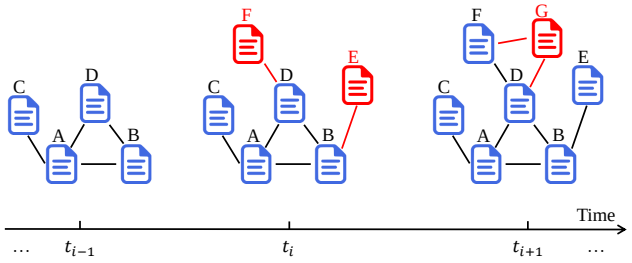


Figure 1. Illustration of a temporal document network.

We postulate that the temporal nature relates not only to when a document is created, but also to how documents created at different times may form linkages. Fig. 1 illustrates the formation of a *temporal document network*. Initially at time t_{i-1}, the network contains four documents (A, B, C, and D) and links among them. At time t_i, two new documents, E and F, are published, and bring links to documents B and D, respectively. We use red documents and links to denote the newly appearing data. Moving to time t_{i+1}, document G is published and connected to documents D and F. As time goes by, we observe the growth in terms of both corpus size and network connectivity. Modeling time could better preserve text semantics and network topology. Moreover, time also reveals the possibility of connection. Indicatively, newly published documents are likely to connect to recent neighbors rather than previous ones. Existing topic models for document networks, e.g., RTM (Chang & Blei, 2009), focus on the static network. As a result, it may predict a past link using documents published in future time.

Our strategy to better model topics in temporal document networks is a confluence of three factors. *First*, most dynamic topic models (Blei & Lafferty, 2006) deal with the plain text within documents and ignore the network connectivity across documents. However, links constitute additional information on documents' similarities, and modeling them could reveal insightful semantics. *Second*, models for networked documents (Chang & Blei, 2009) mainly focus on static networks without considering time. Dynamic process showcases topic evolution and network generation over the time. By modeling it, we may better preserve text semantics and network topology. *Third*, most topic models (Chang & Blei, 2009; Bai et al., 2018) preserve network structure

[1]School of Computing and Information Systems, Singapore Management University, 80 Stamford Road, Singapore. Correspondence to: Delvin Ce Zhang <cezhang.2018@smu.edu.sg>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

by modeling first-order neighborhood only, a limited use of network adjacency. The generation of a link between two documents may be influenced by their common historical neighbors, which is higher-order proximity.

Contributions. We propose neural topic models for dynamic document networks that jointly preserve document dynamics and network adjacency. Optimal Transport (OT) (Cuturi, 2013) measures the distance between two distributions and was adopted in topic modeling (Zhao et al., 2020), but none explored OT in a dynamic setting. In this work, we incorporate time into OT and propose two dynamic topic models, NetDTM and NetDTM++, for **D**ynamic **T**opic **M**odeling on **Net**worked documents (*first* contribution).

Specifically for NetDTM, as the *second* contribution, in addition to the topic and word dimension, we add one more time dimension to OT and develop a Time-Aware Optimal Transport, which measures the probability of a link between two differently timestamped documents using their semantic distance. OT benefits our model by incorporating semantically related word embeddings in cost matrix.

Besides the semantic-level modeling by NetDTM, we discover that the generation of a link is also influenced by the evolving topological structure of network. While NetDTM accounts for semantic modeling, as the *third* contribution, we further propose NetDTM++ for network-level modeling, which designs a Temporal Point Process to capture the impact of network structure on the current link.

Fourth, extensive experiments on four datasets demonstrate the advantage of our models over comparable baselines.

2. Related Work

Classical topic models are graphical models (Hofmann, 1999; Blei et al., 2003). Recent ones are neural models, e.g., ProdLDA (Srivastava & Sutton, 2017), NVDM (Miao et al., 2016), DVAE (Burkhardt & Kramer, 2019), WHAI (Zhang et al., 2018), KATE (Chen & Zaki, 2017), which extend VAE (Kingma & Welling, 2013) for topic modeling. For short texts (Das et al., 2015; Li et al., 2016; Dieng et al., 2020), some use pre-trained word embeddings (Mikolov et al., 2013; Pennington et al., 2014). However, these models do not explore document dynamics or network structure.

Optimal transport (Cuturi, 2013) has been used in topic modeling, including NSTM (Zhao et al., 2020), DWL (Xu et al., 2018), and OTLDA (Huynh et al., 2020). Others (Nan et al., 2019; Patrini et al., 2020) measure Wasserstein distance between predicted and ground-truth word distribution. They do not capture time information or network connectivity.

For documents with publication time, dynamic models use time to improve topic modeling. DTM (Blei & Lafferty, 2006) uses a Markov chain (Bishop, 2006) for semantic evolution. cDTM (Wang et al., 2008) uses Brownian motion. DETM (Dieng et al., 2019) incorporates word embeddings. MDTM (Iwata et al., 2010) allows online update of its parameters. Others (Bhadury et al., 2016; Jähnichen et al., 2018) speed up the inference by sampling. They incorporate time, but ignore the network adjacency across documents.

There are some models for networked documents. RTM (Chang & Blei, 2009) and PLANE (Le & Lauw, 2014) are graphical models. NRTM (Bai et al., 2018) applies VAE and multi-layer perceptron (Bishop, 2006) to predict the links. Adjacent-Encoder (Zhang & Lauw, 2020) captures network by neighbor reconstruction. SemiVN (Zhang & Lauw, 2021) models links with document labels. They model the effect of network, but ignore the dynamic process of network generation. Other models for attributed graph embedding (Kipf & Welling, 2016; Veličković et al., 2018) and temporal graph embedding (Zuo et al., 2018; Xu et al., 2020) are not comparable, since they are not topic models.

3. Preliminaries

Definition 3.1 (Temporal Document Network)**.** Let a temporal document network $\mathcal{G}$ be a tuple $\{\mathcal{D}, \mathcal{E}, \mathcal{T}\}$. $\mathcal{D} = \{\mathbf{d}_i\}_{i=1}^N$ contains documents. Each document $\mathbf{d}_i \in \mathbb{R}^{|\mathcal{V}|}$ is a vector in the vocabulary space $\mathcal{V}$. $\mathcal{E} = \{\mathcal{E}_t\}_{t=1}^T$ is a set of adjacency matrices. $\mathcal{E}_t \in \mathbb{R}^{N\times N}$ is the adjacency matrix at timestamp t, where $e_{ijt} = 1$ if there is a link between document i and j at timestamp t, $e_{ijt} = 0$ otherwise. T is the maximum timestamp. In this paper we consider an undirected network, $e_{ijt} = e_{jit}$. For a document i, its cumulative neighbors observed at timestamp t are those directly linked to i from the initial timestamp to t, denoted as $\mathcal{N}_t(i)$. We consider i as its own neighbor, $i \in \mathcal{N}_t(i)$. $\mathcal{T} = \{t_i\}_{i=1}^N$ contains timestamps, t_i is the publication time of document i. If i and j are published at the same time, $t_i = t_j$.

Given $\mathcal{G}$ as input, we propose a neural topic model and derive topic distributions that preserve document semantics $\mathcal{D}$, evolved network structure $\mathcal{E}$, and dynamics $\mathcal{T}$.

Definition 3.2 (Optimal Transport)**.** Optimal transport measures the distance between two probabilities. Given $\mathbf{r} \in \mathbb{R}^{D_r}$ and $\mathbf{q} \in \mathbb{R}^{D_q}$, where their respective dimension D_r and D_q may not be the same, their OT distance is

$$d_{\mathbf{C}}(\mathbf{r}, \mathbf{q}) = \min_{\mathbf{P}\in U(\mathbf{r},\mathbf{q})} < \mathbf{P}, \mathbf{C} >= \min_{\mathbf{P}\in U(\mathbf{r},\mathbf{q})} \sum_{u=1}^{D_r}\sum_{v=1}^{D_q} p_{uv}c_{uv}. \tag{1}$$

$\mathbf{C} \in \mathbb{R}_{\geq 0}^{D_r\times D_q}$ is cost matrix, each element c_{uv} measures the cost of transport between r_u and q_v. $\mathbf{P} \in \mathbb{R}_{>0}^{D_r\times D_q}$ is transport plan. $U(\mathbf{r}, \mathbf{q}) = \{\mathbf{P} \in \mathbb{R}_{>0}^{D_r\times D_q} | \mathbf{P}\mathbf{1}_{D_q} = \mathbf{r}, \mathbf{P}^\top \mathbf{1}_{D_r} = \mathbf{q}\}$ is the transport polytope with $\mathbf{r}$ and $\mathbf{q}$ as marginals. $\mathbf{1}_D$ is a D-dimensional vector with ones. Thus, each element $p_{uv} \in \mathbf{P}$ is the probability of transport between r_u and q_v.

Given a cost matrix $\mathbf{C}$, OT distance between $\mathbf{r}$ and $\mathbf{q}$ is to find the optimal plan $\mathbf{P}^*$ and obtain $d_{\mathbf{C}}(\mathbf{r},\mathbf{q}) =< \mathbf{P}^*, \mathbf{C} >$.

We will extend OT to incorporate time and use it to measure the semantic distance between two differently timestamped documents i and j as the probability of the link e_{ijt}.

Definition 3.3 (Temporal Point Process)**.** Temporal point process models the discrete sequential events. It measures the conditional probability $\lambda_\epsilon(t)\Delta t$ of an event ϵ happening in a tiny window $[t, t+\Delta t)$ by assuming that historical events before timestamp t can influence the occurrence of the current event. Here, $\lambda_\epsilon(t)$ is conditional intensity function of event ϵ at timestamp t. Hawkes process is a typical temporal point process. Given historical events $\{\epsilon_h | t_h < t\}$ before timestamp t, its conditional intensity function models the arrival rate of the current event ϵ at timestamp t.

$$\lambda_\epsilon(t) = \mu_\epsilon(t) + \sum_{\epsilon_h : t_h < t} \beta_{\epsilon_h,\epsilon} \kappa(t - t_h), \tag{2}$$

$\mu_\epsilon(t)$ is base intensity (the spontaneous arrival rate of the current event ϵ at timestamp t). $\beta_{\epsilon_h,\epsilon}$ is the influence of the historical event ϵ_h on the current ϵ. $\kappa(t-t_h)$ is time decay.

We will use Hawkes process to capture the influence of historical neighbors on the current link formation e_{ijt}.

4. Model Architecture and Analysis

Here, we describe the technical details of our models. See Appendix A (Table 4) for the summary of math notations.

4.1. NetDTM for Semantic-Level Modeling

We first present NetDTM for semantic-level modeling, and defer the modeling of NetDTM++ to the next subsection. Each document $\mathbf{d} \in \mathbb{R}^{|\mathcal{V}|}$ is a distribution in the word space, where each element $d_w = \frac{n_w}{\sum_{w' \in \mathcal{V}} n_{w'}}$ is normalized by the length of the document. Here, n_w is the word count of w in the document. Given document i with its content $\mathbf{d}_i$ on the network, as in (Burkhardt & Kramer, 2019), we encode it into a K-dimensional topic distribution by $\mathbf{z}_i = \theta(\mathbf{d}_i)$, i.e.,

$$\begin{aligned} \mathbf{z}_i &:= dropout(ReLU(\mathbf{W}_1 \mathbf{d}_i + \mathbf{b}_1)), \\ \mathbf{z}_i &:= softmax(batch\_norm(\mathbf{W}_2 \mathbf{z}_i + \mathbf{b}_2)). \end{aligned} \tag{3}$$

$\mathbf{W}_1 \in \mathbb{R}^{200 \times |\mathcal{V}|}$, $\mathbf{W}_2 \in \mathbb{R}^{K \times 200}$, $\mathbf{b}_1 \in \mathbb{R}^{200}$, $\mathbf{b}_2 \in \mathbb{R}^K$ are parameters. We follow (Burkhardt & Kramer, 2019) to choose 200 as intermediate dimension. $ReLU(x) = \max(0, x)$ and $softmax(x) = \frac{\exp(x_k)}{\sum_{k'=1}^K \exp(x_{k'})}$.

Time-Aware Attention. We seek an attention mechanism to evaluate the importance of neighbors. For one, neighbors with similar semantics should be assigned high attention. For another, two linked documents with close publication timestamps are more likely to share similar topics, and should preserve higher attention values. Taking both aspects into account, we design a time-aware attention with both semantic similarity and timestamp difference.

$$\begin{aligned} \tilde{a}_{ij} &= tanh([\mathbf{W}_{att}(\mathbf{z}_i || \mathbf{h}_{t_i})]^\top [\mathbf{W}_{att}(\mathbf{z}_j || \mathbf{h}_{t_j})]), \\ a_{ij} &= \frac{\exp(\tilde{a}_{ij})}{\sum_{j' \in \mathcal{N}_t(i)} \exp(\tilde{a}_{ij'})}. \end{aligned} \tag{4}$$

$(\cdot||\cdot)$ is concatenation, $\mathbf{W}_{att} \in \mathbb{R}^{K \times 3K}$ is parameter. Attention values are jointly determined by topic distribution $\mathbf{z}$ at Eq. 3 and time embedding $\mathbf{h}_t$ to be discussed shortly. Thus, two documents i and j present a high attention if their topics are similar, and their publication timestamps are close.

We now define time embedding. The relative difference between two timestamps, rather than the absolute value of any timestamp, reveals attention values, since the relative timespan informs how close two documents are. Furthermore, the attention at Eq. 4 involves the product of two time embeddings of t_i and t_j. A desirable time embedding should capture timestamp difference when taking product.

$$\mathbf{h}_t = \sqrt{\frac{1}{K}} [\cos(\omega_1 t), \sin(\omega_1 t), ..., \cos(\omega_K t), \sin(\omega_K t)]^\top. \tag{5}$$

$\{\omega_k\}_{k=1}^K$ are parameters. The reason behind such design is

$$\begin{aligned} \mathbf{h}_{t_i}^\top \mathbf{h}_{t_j} =& \frac{1}{K} [\cos(\omega_1 t_i)\cos(\omega_1 t_j) + \sin(\omega_1 t_i)\sin(\omega_1 t_j) + ... \\ &+ \cos(\omega_K t_i)\cos(\omega_K t_j) + \sin(\omega_K t_i)\sin(\omega_K t_j)] \\ =& \frac{1}{K} [\cos(\omega_1 (t_i - t_j)) + ... + \cos(\omega_K (t_i - t_j))] \\ \approx& \mathbb{E}_\omega [\cos(\omega (t_i - t_j))]. \end{aligned} \tag{6}$$

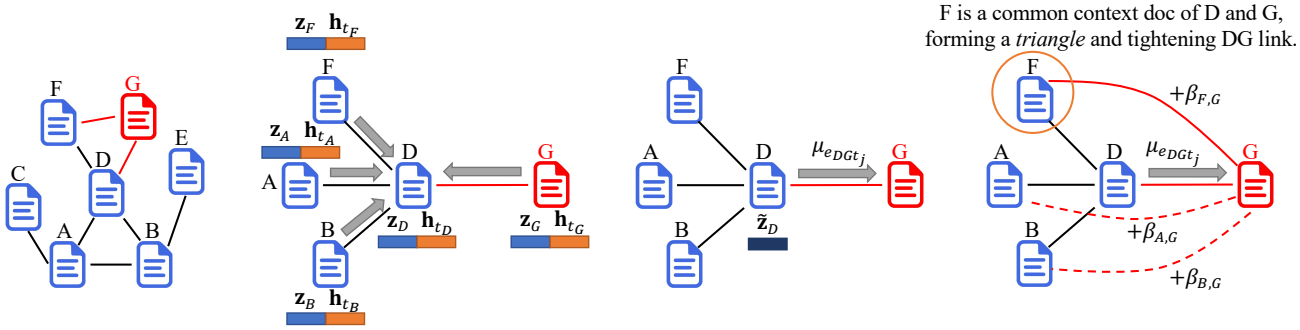
The product of two time embeddings is transformed into the timestamp difference, which aligns with our requirement.

Linked documents tend to share similar topics, e.g., cited papers discuss similar research problems. We thus aggregate topics of document i's neighbors to itself and obtain

$$\tilde{\mathbf{z}}_i = \sum_{j \in \mathcal{N}_t(i)} a_{ij} \mathbf{z}_j. \tag{7}$$

At Fig. 2(a), document G is published at time t_j. We model link e_{DGt_j} for illustration. Fig. 2(b) shows time-aware attention. We aggregate topics of neighbors to document D.

Time-Aware Optimal Transport. OT has achieved promising results in topic modeling (Zhao et al., 2020), but none designs its dynamic version. We are motivated to incorporate time into OT. A document i is represented by two distributions, latent topic distribution $\tilde{\mathbf{z}}_i$ and observed content $\mathbf{d}_i$. They should consistently reflect the same document. We thus seek to minimize the OT semantic distance between topic $\tilde{\mathbf{z}}_i$ and word distribution $\mathbf{d}_i$, i.e., $\min d_{\mathbf{C}}(\tilde{\mathbf{z}}_i, \mathbf{d}_i)$.



(a) A network at time t_j. We model link e_{DGt_j}.

(b) Time-aware attention and aggregation.

(c) Decoding with time-aware OT and log-likelihood.

(d) Hawkes process with influence of historical neighbors.

Figure 2. Illustration of modeling process.

Since links indicate a similar latent semantics of two documents, we allow OT to push topic distribution $\tilde{\mathbf{z}}_i$ to document i's neighbors $j \in \mathcal{N}_t(i)$, i.e., $\min d_{\mathbf{C}}(\tilde{\mathbf{z}}_i, \mathbf{d}_j)$. Since documents are sequential, with recently published documents linking to previous ones, the publication timestamps t_i and t_j may not be the same. However, original optimal transport at Eq. 1 does not preserve such time information. To model document dynamics, we now propose Time-Aware Optimal Transport, which also takes timestamps as inputs.

$$\min d_{\mathbf{C}}(\tilde{\mathbf{z}}_i, \mathbf{d}_j, t_i, t_j), \tag{8}$$

where $j \in \mathcal{N}_t(i)$ is document i's neighbors at timestamp t.

Definition 4.1 (Time-Aware Optimal Transport)**.** Given $\tilde{\mathbf{z}}_i \in \mathbb{R}^K$ with time t_i, and $\mathbf{d}_j \in \mathbb{R}^{|\mathcal{V}|}$ with time t_j (without loss of generality, $t_j \geq t_i$), the time-aware OT distance is

$$d_{\mathbf{C}}(\tilde{\mathbf{z}}_i, \mathbf{d}_j, t_i, t_j) = \min_{\mathbf{P} \in U(\tilde{\mathbf{z}}_i, \mathbf{d}_j, t_i, t_j)} \sum_{t=t_i}^{t_j} \sum_{k=1}^{K} \sum_{w=1}^{|\mathcal{V}|} p_{tkw} c_{tkw}. \tag{9}$$

$\mathbf{C} \in \mathbb{R}_{\geq 0}^{(t_j - t_i + 1) \times K \times |\mathcal{V}|}$ is cost matrix, each element c_{tkw} measures the cost of transport between topic k and word w at timestamp t. $\mathbf{P} \in \mathbb{R}_{>0}^{(t_j - t_i + 1) \times K \times |\mathcal{V}|}$ is transport plan. $U(\tilde{\mathbf{z}}_i, \mathbf{d}_j, t_i, t_j) = \{\mathbf{P} \in \mathbb{R}_{>0}^{(t_j - t_i + 1) \times K \times |\mathcal{V}|} | \mathbf{1}_{(t_j - t_i + 1)} \mathbf{P} \mathbf{1}_{|\mathcal{V}|} = \tilde{\mathbf{z}}_i, (\mathbf{1}_{(t_j - t_i + 1)} \mathbf{P})^\top \mathbf{1}_K = \mathbf{d}_j\}$ is the transport polytope with $\tilde{\mathbf{z}}_i$ and $\mathbf{d}_j$ as marginals. $\mathbf{1}_D$ is a D-dimensional vector with ones. Each element $p_{tkw} \in \mathbf{P}$ is the probability of transport between topic k and word w at time t. Given a cost matrix $\mathbf{C}$, time-aware OT distance between $\tilde{\mathbf{z}}_i$ and $\mathbf{d}_j$ is to find the optimal plan $\mathbf{P}^*$ and obtain $d_{\mathbf{C}}(\tilde{\mathbf{z}}_i, \mathbf{d}_j, t_i, t_j) = < \mathbf{P}^*, \mathbf{C} >$.

Comparing Eq. 9 with original OT at Eq. 1, in addition to the summation over topics and words, we further add one more time dimension for summation across the timespan $t_j - t_i + 1$. Thus, time-aware OT measures the semantic distance between topic $\tilde{\mathbf{z}}_i$ and word distribution $\mathbf{d}_j$ across the timespan. Original OT becomes a special case of time-aware OT when document i and j are published at the same timestamp with no timespan. Intuitively, two documents that are semantically similar and published closely should present a low OT distance, since they do not transport with much semantic cost across a long timespan, and vice versa. Thus, we can use time-aware OT to measure the probability of link e_{ijt}. Lower the distance, higher the probability.

Algorithm 1 Time-Aware Sinkhorn Iteration

Input: Document i's topic distribution $\tilde{\mathbf{z}}_i$, neighbor j's word distribution $\mathbf{d}_j$, timestamp t_i and t_j, cost matrix $\mathbf{C}$, γ.
Output: Time-aware OT distance $d_{\mathbf{C}}(\tilde{\mathbf{z}}_i, \mathbf{d}_j, t_i, t_j)$.

Initialize $\mathbf{\Psi}_1 = \frac{\mathbf{1}_K}{K}$, $\mathbf{t} = \frac{\mathbf{1}_{(t_j - t_i + 1)}}{t_j - t_i + 1}$, $\mathbf{\Phi} = \exp(-\frac{\mathbf{C}}{\gamma})$.
while not converged **do**
  $\mathbf{\Psi}_2 = \frac{\mathbf{d}_j}{(\mathbf{t}\mathbf{\Phi})^\top \mathbf{\Psi}_1}$, $\mathbf{\Psi}_1 = \frac{\tilde{\mathbf{z}}_i}{(\mathbf{t}\mathbf{\Phi})\mathbf{\Psi}_2}$.
end while
Obtain OT plan $\mathbf{P}^* = \text{diag}(\mathbf{\Psi}_1)(\mathbf{t}\mathbf{\Phi})\text{diag}(\mathbf{\Psi}_2)$
Obtain time-aware OT $d_{\mathbf{C}}(\tilde{\mathbf{z}}_i, \mathbf{d}_j, t_i, t_j) = < \mathbf{P}^*, \mathbf{C} >$.

We now define the semantic cost matrix $\mathbf{C}$. Each element c_{tkw} is the semantic dissimilarity between topic k and word w at time t. We choose cosine dissimilarity, $c_{tkw} = 1 - \cos(\mathbf{g}_{tk}, \mathbf{e}_w)$. $\mathbf{g}_{tk}$ is the randomly initialized embedding for topic k at time t, and $\mathbf{e}_w$ is the pre-trained word embedding. External knowledge is incorporated into our model.

OT distance can be calculated by Sinkhorn iteration (Cuturi, 2013). Since we extend the original OT for time information, we propose Time-Aware Sinkhorn Iteration at Algo. 1.

Decoding. Time-aware optimal transport at Eq. 8 pushes topic distribution $\tilde{\mathbf{z}}_i$ to neighboring word distribution $\mathbf{d}_j$, which is similar to a decoding process. We further explicitly design a decoder below to generate the content of neighbors.

$$\hat{\mathbf{d}}_j = \frac{1}{t_j - t_i + 1} \sum_{t=t_i}^{t_j} softmax((2 - \mathbf{C}_t)^\top \tilde{\mathbf{z}}_i). \tag{10}$$

Here, $\mathbf{C}_t \in \mathbb{R}^{K \times |\mathcal{V}|}$ is the t^{th} slice of semantic cost ma-

trix $\mathbf{C}$, which captures topic-word distribution and is used as decoding parameter. We average the output across the timespan $t_j - t_i + 1$ as the generated content. We obtain log-likelihood, $l(\mathbf{d}_j, \hat{\mathbf{d}}_j) = \mathbf{d}_j^\top \log \hat{\mathbf{d}}_j$, of the generative process. Finally, as in (Zhao et al., 2020), combining log-likelihood and time-aware OT, we have the following loss function.

$$\begin{aligned} \mu_{e_{ijt}} &= l(\mathbf{d}_j, \hat{\mathbf{d}}_j) - \eta_{OT} d_{\mathbf{C}}(\tilde{\mathbf{z}}_i, \mathbf{d}_j, t_i, t_j), \\ L_{NetDTM} &= -\sum_{t=1}^{T} \sum_{e_{ijt} \in \mathcal{E}_t} \mu_{e_{ijt}} + \eta_p L_p. \end{aligned} \tag{11}$$

η_{OT} balances log-likelihood and time-aware OT. $\mu_{e_{ijt}}$ is the probability of link e_{ijt}. Fig. 2(c), we generate G's content.

Different timestamps have their own topic embeddings $\{\mathbf{g}_{tk}\}_{t=1}^T$ and topic-word distributions $\{\mathbf{C}_t\}_{t=1}^T$. To associate the modeling of different timestamps and capture topic evolution across the whole time period, we seek to chronologically *chain* the topics. Following (Dieng et al., 2019), we draw topic embeddings using a Markov chain with Gaussian distribution, $\mathbf{g}_{tk} \sim p(\mathbf{g}_{tk}|\mathbf{g}_{t-1,k}) = \mathcal{N}(\mathbf{g}_{t-1,k}, \sigma^2\mathbf{I})$ for $t = 2, ..., T$ and $k = 1, ..., K$. Its log-likelihood is

$$\log p(\mathbf{g}_{tk}|\mathbf{g}_{t-1,k}) \propto -\frac{1}{2\sigma^2}||\mathbf{g}_{tk} - \mathbf{g}_{t-1,k}||^2 \tag{12}$$

We set $\eta_p = -\frac{1}{2\sigma^2}$ for simplicity. Summing all the timestamps and topics, we obtain $\eta_p L_p = \eta_p \sum_{t=2}^{T} \sum_{k=1}^{K} ||\mathbf{g}_{tk} - \mathbf{g}_{t-1,k}||^2$. Adding such a prior term to the loss function at Eq. 11 as a regularizer, we obtain multiple topic embeddings $\{\mathbf{g}_{tk}\}_{t=1}^T$, which capture topic evolution.

4.2. NetDTM++ for Network-Level Modeling

The above process captures network structure by generating content of neighbors. Such a process measures the semantic similarity of two documents, which is also the *spontaneous* probability of an event (the natural formation of link e_{ijt}), without the impact of historical events. Besides the internal semantics, we discover that link e_{ijt} is also externally influenced by the existing network topology generated so far. Two papers with a lot of common citations also likely cite each other, and such common citations enhance the possibility of their similar research topics. While NetDTM captures semantic modeling, here we propose an extended model, NetDTM++, to also model the impact of network topology. Thus, we model the impact of previous links.

Hawkes Process Modeling. For a document i, we apply the same encoding process at Eq. 3 and time-aware attention at Eq. 4 to obtain its aggregated topics $\tilde{\mathbf{z}}_i$. To model the effect of previous links $\{e_{t_h}\}_{t_h \le t}$ on the establishment of the current link e_{ijt}, we design a Hawkes Process, i.e.,

$$\lambda_{e_{ijt}} = \mu_{e_{ijt}} + \eta_{HP} \sum_{e_{t_h}: t_h \le t} \beta_{e_{t_h}, e_{ijt}} \kappa(t - t_h). \tag{13}$$

$\mu_{e_{ijt}}$ is base intensity obtained at Eq. 11, with time-aware OT and log-likelihood. The second term models the impact of previous links. $\beta_{e_{t_h}, e_{ijt}}$ is the influence of a previous link e_{t_h} on the current e_{ijt}, $\kappa(t - t_h)$ is time decay term. Hyperparameter η_{HP} balances semantic and network modeling. NetDTM becomes a special case when $\eta_{HP} = 0$.

However, the second term requires the summation over the entire link set generated so far, which is inefficient in computation. Moreover, usually neighbors of document i influence the formation of e_{ijt} the most; links multiple hops away from e_{ijt} almost have no impact, but likely bring noisy information. Thus, we modify Eq. 13 to only consider links between i and its neighbors, but not all the links. This process models the second-order proximity at Fig. 2(d).

$$\lambda_{e_{ijt}} = \underbrace{\mu_{e_{ijt}}}_{\text{semantic modeling}} + \underbrace{\eta_{HP} \sum_{p \in \mathcal{N}_t(i)} \beta_{pj} a_{ip}}_{\text{network modeling}}. \tag{14}$$

β_{pj} models the influence of second-order proximity p, which represents the surrounding network context between i and j. As mentioned, two documents sharing similar contextual vertices should preserve a high semantic similarity. At Fig. 2(d), document F is a common context of D and G. Such network structure forms a triangle, which tightens the link between D and G and enhances their semantic similarity. As in LINE (Tang et al., 2015), to model second-order proximity, we introduce a context embedding $\mathbf{w}_p \in \mathbb{R}^K$.

$$\beta_{pj} = \log \sigma(\mathbf{w}_p^\top \tilde{\mathbf{z}}_j) + \sum_{m=1}^{M} \mathbb{E}_{d \sim \Pr_n(d)} [\log \sigma(-\mathbf{w}_d^\top \tilde{\mathbf{z}}_j)]. \tag{15}$$

$\sigma(x) = \frac{1}{1+\exp(-x)}$, M is the number of negative samples, $\Pr_n(d)$ is a noise distribution. A high value of β_{pj} increases $\lambda_{e_{ijt}}$, the log-likelihood of the link. At Fig. 2(d), besides the semantic decoding, D's neighbors also influence G by adding context $\beta_{\cdot,G}$. Higher-order proximity is modeled.

Time-aware attention a_{ip} at Eq. 14 measures the importance of neighbors, including semantic similarity and time difference. Since a_{ip} already contains time difference, we do not add an extra time decay. Finally, the loss of NetDTM++ is

$$L_{NetDTM++} = -\sum_{t=1}^{T} \sum_{e_{ijt} \in \mathcal{E}_t} \lambda_{e_{ijt}} + \eta_p L_p. \tag{16}$$

After training convergence, we infer the topic distribution of a previously unseen document $\mathbf{d}'$ by $\mathbf{z}' = \theta(\mathbf{d}')$.

Complexity. Encoding is $\mathcal{O}(200(|\mathcal{V}| + K))$. Time-aware attention is $\mathcal{O}(K^2 + \deg_{\max} K)$. $\deg_{\max}$ is the maximum degree. Time-aware OT is $\mathcal{O}(WK|\mathcal{V}|T)$. W is the dimension of word embeddings. Decoding is $\mathcal{O}(K|\mathcal{V}|T)$. Hawkes process is $\mathcal{O}(K^2 + KN)$. To summarize, we have

Table 1. Dataset statistics.

Name	#Documents	#Links	Vocabulary	#Labels	#Timestamps
ML	1,489	3,474	3,302	7	10
PL	1,424	3,955	3,062	9	13
HEP-TH	27,770	352,285	3,027	N.A.	12
Web	188,741	207,963	5,000	N.A.	10

$\mathcal{O}(200(|\mathcal{V}| + K) + K^2 + WK|\mathcal{V}|T)$ for NetDTM, and $\mathcal{O}(200(|\mathcal{V}|+K)+K^2+WK|\mathcal{V}|T+KN)$ for NetDTM++. Our focus is effectiveness, not efficiency. We briefly report running time. On the largest data Web, NetDTM takes 100 min to converge, NetDTM++ takes 124 min. Experiments were done on a Tesla K80 GPU with 11441MiB. Speeding up the training with possibly online learning is future work.

5. Experiments

The goal of experiments is to evaluate the topics learned by our models against baselines by evaluation tasks, such as document classification, link prediction, topic analysis, etc.

Datasets. Cora (McCallum et al., 2000) is a citation network with abstracts as content and citations as links. Each paper has a publication year. As in (Zhu et al., 2007), we create two independent datasets, Machine Learning (**ML**) and Programming Language (**PL**). ML papers are published between 1989 and 1998, and PL between 1987 and 1999. **HEP-TH** (Leskovec et al., 2005) is a citation network of Physics papers from January 1993 to April 2003. Timestamp can be defined by season, half year, or year, with 46, 23, or 12 timestamps, respectively. **Web** (Leskovec et al., 2009) is a Web page hyperlink network. Each page contains frequent phrases of a news article between August and December 2008. Timestamp can be defined by semimonthly or monthly, with 10 or 5 timestamps. For the following experiments, we use yearly timestamp for ML, PL, and HEP-TH, since academic conferences are usually held annually. For Web, we use semimonthly as timestamp, due to the transience of news articles. Table 1 shows the statistics.

Baselines. We compare to four categories of baselines. *i*) **Static topic models without networks**, ProdLDA (Srivastava & Sutton, 2017), WLDA (Nan et al., 2019), and NSTM (Zhao et al., 2020). WLDA applies Wesserstein distance. NSTM uses optimal transport. They do not model document dynamics or network connectivity. *ii*) **Dynamic topic models**, DTM (Blei & Lafferty, 2006) and DETM (Dieng et al., 2019). DTM extends LDA in a dynamic setting. DETM extends VAE and uses pre-trained word embeddings for dynamic modeling. They incorporate time, but ignore the document adjacency. By comparing to them, we highlight the advantage of jointly modeling dynamics and network structure. *iii*) **Topic models for document networks**, RTM (Chang & Blei, 2009), NRTM (Bai et al., 2018), Adjacent-Encoder (Zhang & Lauw, 2020), and LANTM (Wang et al., 2021). They consider texts and network structure, but ignore dynamic process. By comparison, we show the utility of dynamic modeling. *iv*) **Temporal graph embedding** learns node embeddings on temporal graphs. Strictly speaking, they are not topic models, nor baselines. For completeness, we still compare to M2DNE (Lu et al., 2019).

Implementation Details. We set 2 as Dirichlet prior for RTM. We use 300D GloVe embeddings. For our models, $\eta_{OT} = \eta_{HP} = \eta_p = 1$ after searching in $[0.5, 1, 2, 4, 10]$. Dropout rate is 0.75, $\gamma = 20$, $M = 5$. Each result is obtained by 5 independent runs, with average and std.dev.

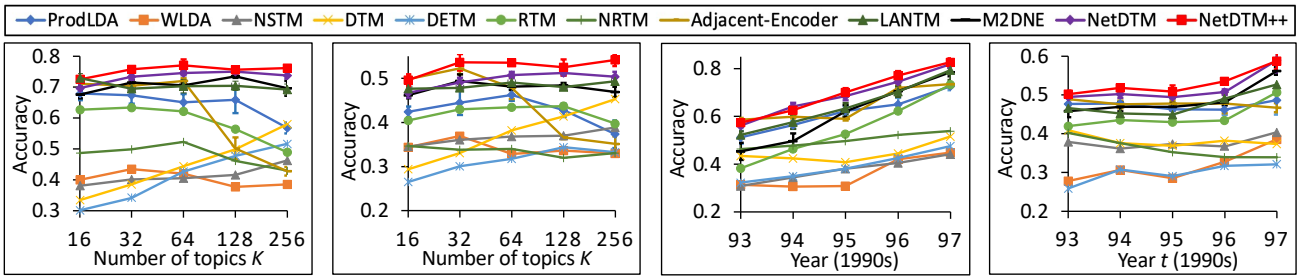
5.1. Quantitative Evaluation

Document Classification. As in LDA (Blei et al., 2003), we conduct document classification to evaluate topic quality. Since we observe network evolution, we split the datasets using timestamps. We split documents before timestamp T (inclusive) for training (10% are for validation). T is the maximum timestamp in the training set. We observe training documents and links among them for training. Labels are never involved for training. After convergence, we infer topics of test documents after T (exclusive). We apply kNN for classification. We input topics of training documents to the classifier, and predict the labels of test documents.

We vary the number of topics K and report classification accuracy with 5NN on ML and PL dataset at Fig. 3(a-b). Here, we train the models using documents and links generated before year $T = 1996$ (inclusive), and predict the labels of documents after $T = 1996$ (exclusive). Such timestamp provides around 80/20 split. Our models and M2DNE show better results than others, since network connectivity indicates similarities among documents. Ours and M2DNE are generally better than Adjacent-Encoder, due to the modeling of dynamic process. Since most models peak their results at 64 topics, we keep $K = 64$ for the following experiments.

We then vary the observed timestamps T from 1993 to 1997, and present the accuracy with 5NN and 64 topics at Fig. 3(c-d). Horizontal axis contain different years T. The goal is to test how models perform when we observe different number of timestamps. As time goes by, the network becomes larger and we observe more timestamps for training, thus the accuracy of most models is increasing. At $T = 1993$ where only a few timestamps are observed, our models show a competitive performance with Adjacent-Encoder, since we can not make full use of time. After moving to recent years, we discover a significant improvement of our models, due to the benefit of document dynamics. NetDTM++ generally classifies documents more accurately than NetDTM, due to Hawkes process modeling the influence of historical events.

We also vary the number of nearest neighbors k for kNN, and put the results in Appendix B.1 (Fig. 6).



(a) Number of topics on ML (b) Number of topics on PL (c) Different years (1990s) on ML (d) Different years (1990s) on PL

Figure 3. Classification accuracy w.r.t. (a-b) different number of topics, and (c-d) different years.

Table 2. Link prediction MAP (left), perplexity (middle), and topic coherence NPMI (right) at $K = 64$ (results are in percentage).

Model	Link Prediction				Perplexity				Topic Coherence			
	ML	PL	HEP-TH	Web	ML	PL	HEP-TH	Web	ML	PL	HEP-TH	Web
ProdLDA	20.1±1.1	19.3±1.2	1.3±0.2	10.7±0.3	8.07±0.00	8.02±0.00	8.71±0.00	8.52±0.00	8.4±0.4	10.2±0.3	11.2±1.0	0.6±0.6
WLDA	7.0±1.0	7.0±2.1	2.1±0.2	11.7±0.2	8.63±0.10	8.62±0.16	42.98±0.07	44.09±0.06	9.4±0.2	11.0±0.5	14.6±0.4	24.2±0.7
NSTM	10.1±2.2	12.1±0.7	1.7±0.1	1.7±0.0	7.97±0.00	7.89±0.00	7.93±0.00	8.28±0.00	16.8±0.9	18.5±0.4	**18.3±0.7**	24.8±1.5
DTM	13.8±2.9	13.5±1.5	6.5±0.3	4.4±0.0	8.10±0.00	8.02±0.00	8.01±0.00	11.61±0.10	10.2±0.3	12.5±0.3	14.2±0.1	13.7±0.4
DETM	15.6±2.8	8.0±1.7	5.3±0.2	16.1±0.0	9.63±0.16	8.06±0.05	7.91±0.06	8.84±0.14	8.3±0.5	8.4±0.4	11.4±0.3	21.1±0.3
RTM	24.3±0.7	21.3±0.8	7.0±0.2	13.8±0.0	7.90±0.02	**7.65±0.02**	7.81±0.00	9.87±0.12	7.1±0.6	8.9±0.3	6.9±0.3	20.1±0.8
NRTM	10.6±1.6	9.0±0.3	1.2±0.0	1.0±0.3	22.72±0.27	30.19±0.14	21.21±0.34	38.43±1.33	6.9±0.4	9.2±0.4	11.6±0.4	19.9±1.7
Adjacent-Encoder	26.3±0.7	22.2±0.4	13.3±0.2	14.8±0.1	8.07±0.03	7.97±0.04	7.89±0.07	8.72±0.09	11.8±0.8	13.4±0.6	17.6±0.0	1.4±0.0
LANTM	24.4±2.1	23.8±1.2	—	—	8.05±0.00	7.98±0.00	—	—	5.4±0.2	6.7±0.7	—	—
M2DNE	16.4±0.2	16.1±0.2	10.1±0.0	1.0±0.0	—	—	—	—	—	—	—	—
NetDTM	25.8±0.7	24.0±0.4	11.4±0.1	**16.7±0.1**	7.89±0.04	7.89±0.05	7.96±0.19	8.69±0.05	**18.9±0.6**	**19.4±0.5**	17.3±0.5	**29.3±0.9**
NetDTM++	**28.3±1.0**	**26.8±0.8**	**14.0±0.3**	**16.7±0.1**	**7.79±0.02**	7.72±0.03	**7.77±0.01**	**8.11±0.21**	16.6±0.6	**19.4±0.6**	17.8±0.3	29.0±1.2

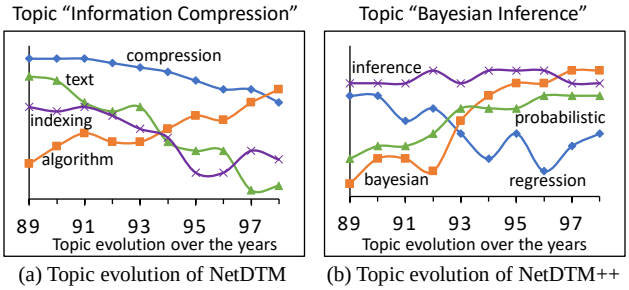
Link Prediction. A topic model should well preserve network structure. As in RTM (Chang & Blei, 2009), we predict network links. We observe training documents and links within them for training. We infer topics of test documents and predict the links among them. As in (Wang et al., 2021), the probability of a link is $p(e_{ij}|\mathbf{z}_i, \mathbf{z}_j) \propto \exp(-||\mathbf{z}_i - \mathbf{z}_j||^2)$. We use mean average precision (MAP) (Zhang & Lauw, 2020) as metric. Again, we split datasets using timestamps. Since different datasets have different timespan, some cross years, others cross a few months, for consistency purpose, we split datasets by 80% timestamps. Here, 80% means we observe documents and links in previous 80% timestamps, and predict links among documents in the future.

Table 2(left) shows that network models perform better than others, since network reveals document relatedness. DTM performs better than WLDA and NSTM, due to dynamic modeling. Compared to them, our models show a significant improvement. This enhances the advantage of jointly modeling dynamics and network. LANTM cannot run on large datasets even on a machine with 256GB memory.

We also vary the number of observed timestamps for link prediction and put the results in Appendix B.2 (Table 5).

5.2. Topic Analysis

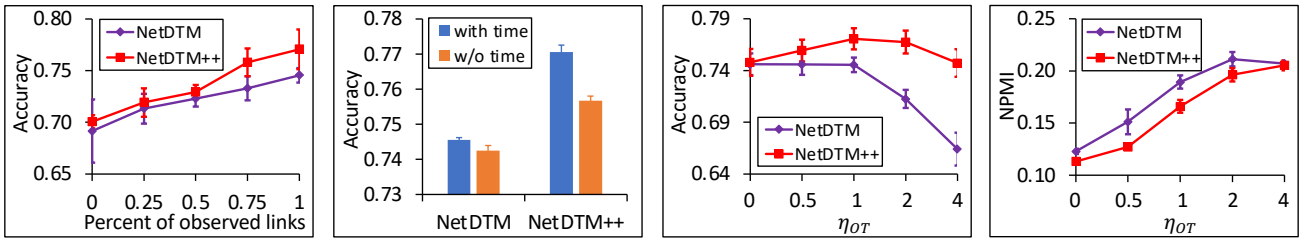
Perplexity. Following LDA (Blei et al., 2003) and DTM (Blei & Lafferty, 2006), we conduct perplexity to evaluate topic quality. For dynamic topic models, we obtain a series of topic-word distributions $\{2 - \mathbf{C}_t\}_{t=1}^{T}$, which capture



(a) Topic evolution of NetDTM (b) Topic evolution of NetDTM++

Figure 4. Topic evolution on ML dataset.

topic evolution over the time. The latest distribution $2 - \mathbf{C}_T$ can best represent the current topic-word distribution. To generalize to future documents published after timestamp T, we should use $2 - \mathbf{C}_T$. This is consistent with (Blei & Lafferty, 2006). Because perplexity, $\exp\{-\frac{\log \Pr(D_{test})}{\sum_{d' \in D_{test}} N_{d'}}\}$, is exponential and varies w.r.t. its power, we show the power, $-\frac{\log \Pr(D_{test})}{\sum_{d' \in D_{test}} N_{d'}}$. Lower is better. Again, we split datasets using 80% timestamps and show the results at Table 2(middle). M2DNE is not a topic model, thus cannot evaluate perplexity. Network models perform the best among baselines. Network indicates document similarity, thus modeling it helps semantic learning. Due to dynamic modeling, DTM provides decent results. By combining both network and dynamics, our models outperform baselines significantly. NetDTM++ is generally better than NetDTM, since Hawkes process incorporating historical events better encode semantically similar document closely. Perplexity with varying the



(a) Effect of network structure (b) Effect of time (c) Effect of OT on doc classification (d) Effect of OT on topic coherence

Figure 5. Model analysis on ML dataset.

Table 3. Time granularity on link prediction (in percentage).

Model	HEP-TH			Web	
	Quarterly	Semianually	Annually	Semimonthly	Monthly
NetDTM	9.55±0.09	9.06±0.11	**11.42±0.15**	**16.72±0.06**	16.70±0.06
NetDTM++	11.51±0.25	10.81±0.09	**13.96±0.33**	**16.73±0.11**	16.66±0.04

number of observed timestamps is found in Appendix B.3.

Topic Coherence. Decoding parameter $2-\mathbf{C}_t \in \mathbb{R}^{K\times|V|}$ captures the keywords of each topic. Each row is the distribution of a topic over the vocabulary. The keywords of that topic are those with the highest values on that row. Following ProdLDA (Srivastava & Sutton, 2017), we evaluate the coherence of top-10 keywords of each topic and report NPMI. We use *Google Web 1T 5-gram Version 1* (Evert, 2010) as external corpus. Table 2(right) shows the results. M2DNE is not a topic model and is excluded. Benefiting from optimal transport, NSTM is the best model among baselines. By comparing to it, our models extend OT to incorporate time information, and improve the performance.

Topic Evolution. To intuitively understand how our models capture topic evolution, Fig. 4 shows the plot on ML dataset. Horizontal axis is different years, and vertical axis is the word probability in $\{2-\mathbf{C}_t\}_{t=1989}^{1998}$. Four lines are randomly selected words of the same topic. For NetDTM, "algorithm" remained a popular research over years. Researchers gradually shifted their focus away from "text indexing", potentially because topic models (PLSA (Hofmann, 1999)) were proposed, and traditional indexing became inefficient. For NetDTM++, "bayesian inference" attracted much attention, while "regression" gradually decayed, possibly because neural network started to present as a universal approximator, and traditional regression models became less interesting.

5.3. Model Analysis

Effect of Network Structure. To verify network structure brings useful information, we randomly remove a proportion of links on the network. We vary the percentage of remaining observed links and report the classification accuracy on ML dataset at Fig. 5(a). As we observe more links, the accuracy increases. Compared to the case with no links, adding a small proportion of links can significantly boost the results, which verifies that network reveals document similarities, and modeling it can improve semantic learning.

Effect of Time. We analyze if modeling dynamics benefits topic modeling. We set the time of all the documents to be the same, i.e., we observe the whole static network without any evolution. We show classification with and without time at Fig. 5(b). Both models increase accuracy when considering time. NetDTM does not improve much, since it mainly models semantics without historical events. NetDTM++ improves significantly, since previous links reveal network evolution, and ignoring time leads to worse accuracy.

Effect of Optimal Transport. To investigate the effectiveness of optimal transport, we vary η_{OT} at Eq. 11. For classification at Fig. 5(c), as η_{OT} increases, the accuracy keeps flat or even becomes higher at the beginning, after which both models decrease the results. For topic coherence at Fig. 5(d), OT can significantly enhance the coherence. Combining Fig. 5(c) and (d), we conclude that compared to the case with no OT, an appropriate value of η_{OT} maintains or even boosts the result, while an overly high value hurts some tasks. Taking the trade-off between two tasks, we set $\eta_{OT}=1$ to combine both OT and log-likelihood.

Effect of Timestamp Granularity. Thus far, we set annually and semimonthly as one timestamp period for HEP-TH and Web, respectively. Here, we use different periods to test the effect of timestamp granularity. Table 3 shows link prediction results. For HEP-TH, a short period of timestamp (quarterly and semiannually) may not observe a significant change of research topics, but brings more parameters. Overfitting problem may decrease the results. For Web, due to the short effective period of news, a long period (monthly) contains too much change of news storyline, and cannot capture the transient topic evolution, thus the results decrease. We follow DTM (Blei & Lafferty, 2006) and set granularity as hyperparameter. Adaptively learning it is a future work.

6. Conclusion

We propose two neural topic models for dynamic document networks. NetDTM designs time-aware OT for neighbor generation. NetDTM++ incorporates the effect of historical links. Experiments verify the effectiveness of our models.

References

Bai, H., Chen, Z., Lyu, M. R., King, I., and Xu, Z. Neural relational topic models for scientific article analysis. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pp. 27–36, 2018.

Bhadury, A., Chen, J., Zhu, J., and Liu, S. Scaling up dynamic topic models. In *Proceedings of the 25th International Conference on World Wide Web*, pp. 381–390, 2016.

Bishop, C. M. *Pattern recognition and machine learning*. springer, 2006.

Blei, D. M. and Lafferty, J. D. Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*, pp. 113–120, 2006.

Blei, D. M., Ng, A. Y., and Jordan, M. I. Latent dirichlet allocation. *the Journal of machine Learning research*, 3: 993–1022, 2003.

Burkhardt, S. and Kramer, S. Decoupling sparsity and smoothness in the dirichlet variational autoencoder topic model. *J. Mach. Learn. Res.*, 20(131):1–27, 2019.

Chang, J. and Blei, D. Relational topic models for document networks. In *Artificial intelligence and statistics*, pp. 81–88. PMLR, 2009.

Chen, Y. and Zaki, M. J. Kate: K-competitive autoencoder for text. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 85–94, 2017.

Cuturi, M. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26:2292–2300, 2013.

Das, R., Zaheer, M., and Dyer, C. Gaussian lda for topic models with word embeddings. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 795–804, 2015.

Dieng, A. B., Ruiz, F. J., and Blei, D. M. The dynamic embedded topic model. *arXiv preprint arXiv:1907.05545*, 2019.

Dieng, A. B., Ruiz, F. J., and Blei, D. M. Topic modeling in embedding spaces. *Transactions of the Association for Computational Linguistics*, 8:439–453, 2020.

Evert, S. Google web 1t 5-grams made easy (but not for the computer). In *Proceedings of the NAACL HLT 2010 Sixth Web as Corpus Workshop*, pp. 32–40, 2010.

Hofmann, T. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 50–57, 1999.

Huynh, V., Zhao, H., and Phung, D. Otlda: A geometry-aware optimal transport approach for topic modeling. *Advances in Neural Information Processing Systems*, 33, 2020.

Iwata, T., Yamada, T., Sakurai, Y., and Ueda, N. Online multiscale dynamic topic models. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 663–672, 2010.

Jähnichen, P., Wenzel, F., Kloft, M., and Mandt, S. Scalable generalized dynamic topic models. In *International Conference on Artificial Intelligence and Statistics*, pp. 1427–1435. PMLR, 2018.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Kipf, T. N. and Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Le, T. M. and Lauw, H. W. Probabilistic latent document network embedding. In *2014 IEEE International Conference on Data Mining*, pp. 270–279. IEEE, 2014.

Leskovec, J., Kleinberg, J., and Faloutsos, C. Graphs over time: densification laws, shrinking diameters and possible explanations. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pp. 177–187, 2005.

Leskovec, J., Backstrom, L., and Kleinberg, J. Meme-tracking and the dynamics of the news cycle. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 497–506, 2009.

Li, C., Wang, H., Zhang, Z., Sun, A., and Ma, Z. Topic modeling for short texts with auxiliary word embeddings. In *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval*, pp. 165–174, 2016.

Lu, Y., Wang, X., Shi, C., Yu, P. S., and Ye, Y. Temporal network embedding with micro-and macro-dynamics. In *Proceedings of the 28th ACM international conference on information and knowledge management*, pp. 469–478, 2019.

McCallum, A. K., Nigam, K., Rennie, J., and Seymore, K. Automating the construction of internet portals with machine learning. *Information Retrieval*, 3(2):127–163, 2000.

Miao, Y., Yu, L., and Blunsom, P. Neural variational inference for text processing. In *International conference on machine learning*, pp. 1727–1736. PMLR, 2016.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pp. 3111–3119, 2013.

Nan, F., Ding, R., Nallapati, R., and Xiang, B. Topic modeling with wasserstein autoencoders. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 6345–6381, 2019.

Patrini, G., van den Berg, R., Forre, P., Carioni, M., Bhargav, S., Welling, M., Genewein, T., and Nielsen, F. Sinkhorn autoencoders. In *Uncertainty in Artificial Intelligence*, pp. 733–743. PMLR, 2020.

Pennington, J., Socher, R., and Manning, C. D. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, 2014.

Srivastava, A. and Sutton, C. Autoencoding variational inference for topic models. In *5th International Conference on Learning Representations*, 2017.

Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., and Mei, Q. Line: Large-scale information network embedding. In *Proceedings of the 24th international conference on world wide web*, pp. 1067–1077, 2015.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., and Bengio, Y. Graph attention networks. In *International Conference on Learning Representations*, 2018.

Wang, C., Blei, D., and Heckerman, D. Continuous time dynamic topic models. In *Proceedings of the Twenty-Fourth Conference on Uncertainty in Artificial Intelligence*, pp. 579–586, 2008.

Wang, Y., Li, X., and Ouyang, J. Layer-assisted neural topic modeling over document networks. In *International Joint Conference on Artificial Intelligence*, pp. 3148–3154, 2021.

Xu, D., Ruan, C., Korpeoglu, E., Kumar, S., and Achan, K. Inductive representation learning on temporal graphs. *arXiv preprint arXiv:2002.07962*, 2020.

Xu, H., Wang, W., Liu, W., and Carin, L. Distilled wasserstein learning for word embedding and topic modeling. In *Advances in Neural Information Processing Systems*, pp. 1716–1725. Neural information processing systems foundation, 2018.

Zhang, C. and Lauw, H. W. Topic modeling on document networks with adjacent-encoder. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 6737–6745, 2020.

Zhang, D. C. and Lauw, H. W. Semi-supervised semantic visualization for networked documents. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 762–778. Springer, 2021.

Zhang, H., Chen, B., Guo, D., and Zhou, M. Whai: Weibull hybrid autoencoding inference for deep topic modeling. *arXiv preprint arXiv:1803.01328*, 2018.

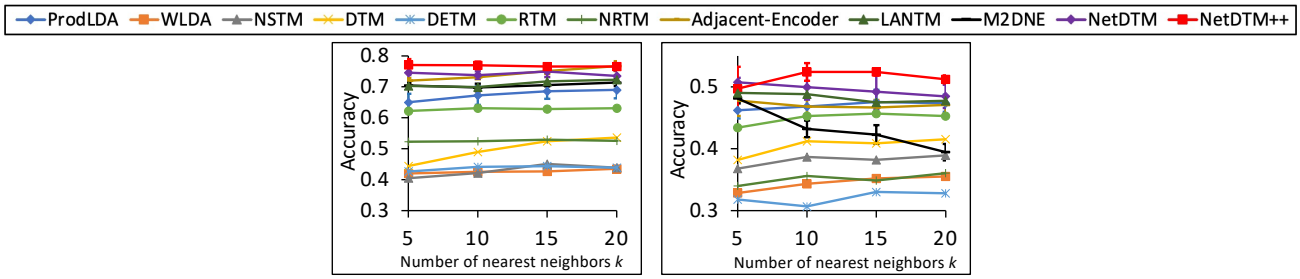
Zhao, H., Phung, D., Huynh, V., Le, T., and Buntine, W. Neural topic model via optimal transport. In *International Conference on Learning Representations*, 2020.

Zhu, S., Yu, K., Chi, Y., and Gong, Y. Combining content and link for classification using matrix factorization. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 487–494, 2007.

Zuo, Y., Liu, G., Lin, H., Guo, J., Hu, X., and Wu, J. Embedding temporal network via neighborhood formation. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 2857–2866, 2018.

Table 4. Summary of math notations.

Notation	Description
$\mathcal{G}$	a document network
$\mathcal{D}$	a corpus of documents, $\mathcal{D} = \{\mathbf{d}_i\}_{i=1}^{N}$
N	number of documents in the corpus, $N = \lvert\mathcal{D}\rvert$
$\mathcal{E}$	a set of adjacency matrices, $\mathcal{E} = \{\mathcal{E}_t\}_{t=1}^{T}$, $\mathcal{E}_t \in \mathbb{R}^{N\times N}$
T	maximum timestamp observed in training set
$\mathbf{d}$	a document representation in the word space, $\mathbf{d} \in \mathbb{R}^{\lvert\mathcal{V}\rvert}$
$\mathcal{V}$	vocabulary
$\mathcal{N}_t(i)$	the neighbor set of document i at time t
$\mathcal{T}$	a set of timestamps, $\mathcal{T} = \{t_i\}_{t=1}^{N}$
$\mathbf{z}_i$	topic distribution of document i, $\mathbf{z}_i \in \mathbb{R}^K$
K	number of topics
a_{ij}	attention value between document i and j
$\mathbf{h}_t$	time embedding of timestamp t, $\mathbf{h}_t \in \mathbb{R}^{2K}$
$\mathbf{C}$	cost matrix of time-aware OT, $\mathbf{C} \in \mathbb{R}^{T\times K\times\lvert\mathcal{V}\rvert}$
$\mathbf{P}$	transport plan of time-aware OT
$\mathbf{g}_{tk}$	topic embedding of topic k at timestamp t
$\mathbf{e}_w$	word embedding of word w
M	number of negative samples



(a) Number of neighbors on ML (b) Number of neighbors on PL

Figure 6. Classification accuracy w.r.t. different number of nearest neighbors k for kNN.

A. Summary of Notations

We provide a summary of math notations used in the main paper.

B. Additional Experiments

B.1. Document Classification with Different Number of Nearest Neighbors k

In the main paper, we use $k = 5$ nearest neighbors for kNN classification. Here, we vary the value of k and report classification accuracy for ML and PL at Fig. 6. Overall, our models present a stable performance across different number of nearest neighbors. NetDTM is competitive with Adjacent-Encoder on ML, but outperforms the latter on PL. M2DNE generally deteriorates the results when more nearest neighbors are considered. NetDTM++ classifies documents more accurately than NetDTM, since Hawkes process benefits the model by considering the impact of historical neighbors. Comparing our models against network models, including Adjacent-Encoder, we highlight that modeling dynamics is indeed useful to improve topic modeling. Compared to dynamic models without network structure, we verify that network connectivity uncovers semantic similarities across documents, and modeling it improves topic quality.

Table 5. Link prediction MAP at $K = 64$ (results are in percentage) when varying the percentage of total timestamps.

Model	ML			PL			HEP-TH			Web		
	40%	60%	80%	40%	60%	80%	40%	60%	80%	40%	60%	80%
ProdLDA	6.5±0.4	10.8±0.3	20.1±1.1	6.9±0.3	11.7±0.8	19.3±1.2	0.4±0.0	0.5±0.0	1.3±0.2	10.7±0.3	10.7±0.3	10.7±0.3
WLDA	3.2±0.5	3.4±0.3	7.0±1.0	2.9±0.2	4.3±0.5	7.0±2.1	0.5±0.0	0.7±0.1	2.1±0.2	7.0±0.0	9.6±0.1	11.7±0.2
NSTM	3.6±0.4	5.7±0.4	10.1±2.2	3.9±0.1	7.2±0.4	12.1±0.7	0.6±0.0	0.8±0.0	1.7±0.1	1.1±0.1	1.3±0.1	1.7±0.0
DTM	6.9±0.3	7.9±0.6	13.8±2.9	7.1±0.2	10.2±0.8	13.5±1.5	2.1±0.0	3.3±0.0	6.5±0.3	3.7±0.0	4.1±0.0	4.4±0.0
DETM	1.7±0.0	9.6±0.4	15.6±2.8	4.5±0.4	6.9±1.5	8.0±1.7	2.7±0.0	3.2±0.0	5.3±0.2	**13.5±0.0**	14.7±0.0	16.1±0.0
RTM	10.4±0.2	15.0±0.4	24.3±0.7	9.5±0.2	15.1±0.5	21.3±0.8	3.3±0.1	4.1±0.1	7.0±0.2	11.1±0.1	12.4±0.0	13.8±0.0
NRTM	6.2±0.4	7.0±0.6	10.6±1.6	6.2±0.5	8.3±0.6	9.0±0.3	0.6±0.0	0.6±0.0	1.2±0.0	0.5±0.0	0.5±0.0	1.0±0.3
Adjacent-Encoder	10.3±0.4	16.3±0.6	26.3±0.7	7.8±0.5	18.9±0.4	22.2±0.4	**6.1±0.1**	**7.9±0.2**	13.3±0.2	**13.5±0.0**	14.5±0.0	14.8±0.1
LANTM	**13.2±0.4**	17.2±0.6	24.4±2.1	**15.0±0.4**	19.1±0.3	23.8±1.2	—	—	—	—	—	—
M2DNE	3.1±0.0	7.2±0.0	16.4±0.2	3.5±0.0	12.2±0.0	16.1±0.2	4.3±0.0	5.6±0.0	10.1±0.0	0.5±0.0	0.7±0.0	1.0±0.0
NetDTM	12.2±0.4	17.3±0.8	25.8±0.7	11.2±0.4	17.8±0.5	24.0±0.4	4.8±0.1	6.2±0.1	11.4±0.1	**13.5±0.0**	14.9±0.0	**16.7±0.1**
NetDTM++	12.0±0.3	**18.0±0.2**	**28.3±1.0**	11.5±0.3	**19.9±0.3**	**26.8±0.8**	5.7±0.0	7.3±0.1	**14.0±0.3**	**13.5±0.0**	**15.0±0.0**	**16.7±0.1**

Table 6. Perplexity experiment at $K = 64$ when varying the percentage of total timestamps. Lower is better.

Model	ML			PL			HEP-TH			Web		
	40%	60%	80%	40%	60%	80%	40%	60%	80%	40%	60%	80%
ProdLDA	8.16±0.00	8.07±0.00	8.07±0.00	8.15±0.00	8.03±0.00	8.02±0.00	8.58±0.05	8.60±0.00	8.71±0.00	8.52±0.00	8.52±0.00	8.52±0.00
WLDA	8.63±0.10	8.60±0.14	8.12±0.99	8.62±0.16	8.34±0.03	8.49±0.44	44.30±0.24	44.50±0.33	42.98±0.07	44.43±0.02	43.73±0.04	44.09±0.06
NSTM	8.05±0.00	7.99±0.00	7.97±0.00	7.97±0.00	7.90±0.00	7.89±0.00	7.93±0.00	7.93±0.00	7.93±0.00	8.28±0.00	8.27±0.00	8.28±0.00
DTM	8.10±0.00	8.10±0.00	8.10±0.00	8.02±0.00	8.02±0.00	8.02±0.00	8.01±0.00	8.01±0.00	8.01±0.00	11.61±0.07	11.59±0.07	11.61±0.10
DETM	11.66±0.18	11.63±0.25	9.63±0.16	8.78±0.09	8.20±0.07	8.06±0.05	8.28±0.09	8.09±0.05	7.91±0.06	10.39±0.19	9.46±0.05	8.84±0.14
RTM	7.99±0.02	7.97±0.01	7.90±0.02	7.84±0.03	7.72±0.05	**7.65±0.02**	7.81±0.00	7.81±0.00	7.81±0.00	20.68±1.51	18.73±0.95	9.87±0.12
NRTM	33.10±0.44	28.99±0.13	22.72±0.27	38.61±2.32	33.43±0.16	30.19±0.14	22.17±0.12	21.32±0.13	21.21±0.34	33.56±0.76	33.56±0.76	38.43±1.33
Adjacent-Encoder	7.94±0.01	7.99±0.02	8.07±0.03	7.82±0.08	7.76±0.09	7.97±0.04	7.86±0.03	7.86±0.03	7.89±0.07	8.72±0.09	8.72±0.09	8.72±0.09
LANTM	8.05±0.00	8.05±0.00	8.05±0.00	7.98±0.00	7.98±0.00	7.98±0.00	—	—	—	—	—	—
NetDTM	**7.90±0.02**	7.89±0.05	7.89±0.04	7.78±0.01	7.81±0.03	7.89±0.05	7.79±0.06	7.81±0.06	7.96±0.19	8.79±0.22	8.79±0.06	8.69±0.05
NetDTM++	**7.90±0.02**	**7.80±0.02**	**7.79±0.02**	**7.72±0.01**	**7.69±0.02**	7.72±0.03	**7.77±0.02**	**7.79±0.02**	**7.77±0.01**	**8.13±0.11**	**7.92±0.08**	**8.11±0.21**

B.2. Link Prediction When Varying Observed Timestamps

In the main paper, we report link prediction results when we split datasets using 80% timestamps. Here, we further vary the percentage of observed timestamps from 40% to 80%. The goal is to investigate how our models perform when we observe different number of timestamps.

Table 5 shows that as the percentage of timestamps increases, most models improve their results, because they observe more documents and links for training. Note that we do not report LANTM on large datasets, since it cannot run even on a machine with 256GB memory. When we observe only 40% timestamps, our models are competitive with LANTM, since we do not observe too many timestamps and our models cannot make full use of time information. When the observed timestamps accumulate to 80%, our models present a significant improvement over LANTM. This enhances the benefit of dynamics in our models as compared to static network models. Compared to models without network structure, we emphasize that incorporating network can bring useful information.

B.3. Perplexity When Varying Observed Timestamps

Similar to document classification and link prediction, we also vary the number of observed timestamps for perplexity experiment. Table 6 shows the results. Our models generally perform better than baseline models. NetDTM++ shows more satisfying results than NetDTM, since the former models higher-order proximity at the network-level using Hawkes process.