Online Egocentric Models for Citation Networks

Hao Wang and **Wu-Jun Li**
Shanghai Key Laboratory of Scalable Computing and Systems
Department of Computer Science and Engineering, Shanghai Jiao Tong University, China
js05212@sjtu.edu.cn, liwujun@cs.sjtu.edu.cn

Abstract

With the emergence of large-scale evolving (time-varying) networks, dynamic network analysis (DNA) has become a very hot research topic in recent years. Although a lot of DNA methods have been proposed by researchers from different communities, most of them can only model snapshot data recorded at a very rough temporal granularity. Recently, some models have been proposed for DNA which can be used to model large-scale citation networks at a fine temporal granularity. However, they suffer from a significant decrease of accuracy over time because the learned parameters or node features are static (fixed) during the prediction process for evolving citation networks. In this paper, we propose a novel model, called *online egocentric model* (OEM), to learn time-varying parameters and node features for evolving citation networks. Experimental results on real-world citation networks show that our OEM can not only prevent the prediction accuracy from decreasing over time but also uncover the evolution of topics in citation networks.

1 Introduction

Network analysis [Goldenberg *et al.*, 2009; Li and Yeung, 2009; Li *et al.*, 2009a; 2009b; Wang *et al.*, 2010; Li *et al.*, 2011; Li and Yeung, 2012; Zhu, 2012; McAuley and Leskovec, 2012; Kim and Leskovec, 2012; Myers *et al.*, 2012], especially dynamic network analysis (DNA) has become increasingly important in many fields like social science and biology. Although there have been a lot of works on DNA, most of them either focus on large-scale data at a very rough temporal granularity [Fu *et al.*, 2009; Wyatt *et al.*, 2010; Hanneke *et al.*, 2010; Richard *et al.*, 2012; Sarkar *et al.*, 2012; Jin *et al.*, 2011; Wang and Groth, 2011; Nori *et al.*, 2011] or focus on small networks at a fine temporal granularity [Wasserman, 1980; Snijders, 2005]. Recently, dynamic egocentric model (DEM) [Vu *et al.*, 2011b], which is based on multivariate counting processes, has been successfully proposed to model large-scale evolving citation networks at a fine temporal granularity of individual time-stamped events.

Although DEM can dynamically update the *link features* (statistics) of the nodes (papers), the learned *parameters* and *topic features* of DEM[1] are static (fixed) during the prediction process for evolving networks. Hence, DEM suffers from a decrease of accuracy over time because typically both the *parameters* and the *topic features* of the papers will evolve over time. For example, one of the link features reflects the in-degree (number of citations) of a paper until some time point. As time goes on, the cumulative number of citations for existing papers will become larger and larger. Hence, the distribution of the citations for the whole data set will change over time. As a consequence, the corresponding parameter which typically reflects the distribution of the features should also change over time. At first sight, it seems a little confusing that the topic features of a paper can change over time because the content of a published paper is typically static. However, the citations to an existing paper are dynamic. It is more reasonable to combine both the citation and content information to decide the topic of a paper. For example, a paper about *neural network* considered to be highly related to the topic *psychology* in the 1950s may be more likely to be classified as a *machine learning* paper today because more and more machine learning papers cite that neural network paper. Hence, it is very obvious to find that the topic features will also change over time. Without the ability to adaptively learn the parameters and topic features, DEM fails to model the evolution of networks. This phenomenon of decreasing prediction accuracy over time can also be observed from the experimental results in Figure 2 of [Vu *et al.*, 2011b].

In this paper, we propose an online extension of DEM, called *online egocentric model* (OEM), to capture the evolution of both topic features and model parameters. The contributions of this paper are briefly outlined as follows:

- OEM takes the evolution of both topic features and parameters into consideration and maintains high prediction accuracy regardless of the elapse of time.

[1]In [Vu *et al.*, 2011b], there are two variants of DEM. One models only the link features, and the other models both the link and topic features (textual information). Unless otherwise stated, the DEM in this paper refers to the variant with both link and topic features because it achieves far better accuracy than DEM without topic features [Vu *et al.*, 2011b] and we can always get the topic features for a paper if we want.

- During the online training of OEM, we can also uncover the evolution of topic features for each paper and the propagation of topic features between pairs of papers.
- Extensive experiments on two real-world citation networks are performed to demonstrate the effectiveness of our novel model.

2 Dynamic Egocentric Model

In this section, we briefly review the DEM [Vu *et al.*, 2011b] which we base our work on. For ease of understanding, we use the same notations as those in [Vu *et al.*, 2011b].

Let n denote the total number of nodes (papers) in the network. DEM tries to model a dynamic (citation) network by placing a counting process $N_i(t)$ on each node i ($i = 1, 2, \cdots, n$) , where $N_i(t)$ denotes the cumulative number of *events* associated with node i until time t. The definition of events depends on context. For example, an event corresponds to a citation in citation networks.

Although a continuous-time model can be estimated by maximizing the full likelihood of the counting process, for citation networks it is more practical to estimate the parameters associated with the time-dependent statistics at event times by maximizing the partial likelihood. Then the DEM tries to maximize the following likelihood of the whole network:

$$L(\boldsymbol{\beta}) = \prod_{e=1}^{m} \frac{\exp(\boldsymbol{\beta}^T \mathbf{s}_{i_e}(t_e))}{\sum_{i=1}^{n} Y_i(t_e) \exp(\boldsymbol{\beta}^T \mathbf{s}_i(t_e))}, \tag{1}$$

where m is the total number of citation events, e is the index of each citation event, i_e denotes the paper cited in event e, t_e denotes the time of event e, $Y_i(t)$ is 1 if node i already exists at time t and 0 otherwise, $\mathbf{s}_i(t_e)$ denotes the feature vector of node i at the time t_e, and $\boldsymbol{\beta}$ is a vector of parameters to learn.

The features in $\mathbf{s}_i(t_e)$ can be divided into two types. One type is called *link features (statistics)*, and the other type is called *topic features*. In [Vu *et al.*, 2011b], eight link features, including three preferential attachment statistics, three triangle statistics and two out-path statistics, are extracted for each node. Fifty topic features are extracted by performing Latent Dirichlet Allocation (LDA) [Blei *et al.*, 2003] on the abstracts of the papers. More specifically, assuming the arriving paper is i at the time t_e, we can compute the topic features of any existing paper j as follows:

$$\mathbf{s}_j^{LDA}(t_e) = \boldsymbol{\theta}_i \circ \boldsymbol{\theta}_j,$$

where $\boldsymbol{\theta}_i$ denotes the topic proportion of paper i, $\circ$ denotes the element-wise multiplication.

Hence, $\mathbf{s}_i(t_e)$ is a vector of 58 features, with the first 8 features being link features and the last 50 features being topic features. Correspondingly, $\boldsymbol{\beta}$ is a vector of length 58. More details about the features can be found in [Vu *et al.*, 2011b].

3 Online Egocentric Model

During the prediction process for evolving networks, the link features of nodes will be dynamically updated in DEM. However, both the learned *parameters* ($\boldsymbol{\beta}$) and *topic features* ($\boldsymbol{\theta}_i$) of DEM will not change with the evolution of networks, which will cause the accuracy of DEM to decrease over time. In this section, we present our *online egocentric model* (OEM) to solve the problems faced by DEM. The basic idea is to adaptively update the parameters and topic features after some new events are observed.

Although we can learn the whole LDA model from the collection of papers, it will be very time-consuming in general even if we adopt the online LDA model [Hoffman *et al.*, 2010]. Hence, in this paper we just learn the topic proportions $\boldsymbol{\theta}$ with the topics fixed. This is reasonable because the main topics in the citation networks are relatively stable although the topic proportions for some papers will change over time. We only need to update the whole topics after a long time period. From our experiments, we find that good performance can still be achieved by only updating the topic proportions.

Hence, our OEM tries to minimize the following objective function after observing some new events:

$$\begin{aligned} & minimize \quad -\log L(\boldsymbol{\beta}, \boldsymbol{\omega}) + \lambda \sum_{k=1}^{n} \|\boldsymbol{\omega}_k - \boldsymbol{\theta}_k\|_2^2 \\ & subject\ to: \quad \boldsymbol{\omega}_k \succeq \mathbf{0},\ \mathbf{1}^T \boldsymbol{\omega}_k = 1, \end{aligned} \tag{2}$$

where $\boldsymbol{\omega}_k$ is the new topic proportions of node k that need to be learned and $\boldsymbol{\theta}_k$ is the current topic proportions of node k, $\boldsymbol{\omega} = \{\boldsymbol{\omega}_k\}_{k=1}^n$, $L(\boldsymbol{\beta}, \boldsymbol{\omega})$ has *the same definition* as $L(\boldsymbol{\beta})$ in (1) by treating both $\boldsymbol{\beta}$ and topic proportions as variables[2], $\boldsymbol{\omega}_k \succeq \mathbf{0}$ denotes each element in $\boldsymbol{\omega}_k$ is non-negative, $\mathbf{1}$ is a vector of all 1s, the constraints are used to guarantee that all elements in $\boldsymbol{\omega}_k$ are non-negative and the summation of the elements in $\boldsymbol{\omega}_k$ is 1, λ is a hyperparameter to control the tradeoff between two terms.

When a new event or a set of new events are observed, the second term in (2) will guarantee that the updated topic proportion $\boldsymbol{\omega}_k$ will not be too far away from the current topic proportion $\boldsymbol{\theta}_k$. Furthermore, we use the current $\boldsymbol{\beta}$ as initialization to get the updated $\boldsymbol{\beta}$. Hence, by effectively using the information of the existing events, we successfully get an online learning algorithm.

It is easy to see that the optimization problem in (2) is not jointly convex in $(\boldsymbol{\beta}, \boldsymbol{\omega})$. But we can prove that the objective function is convex in either $\boldsymbol{\beta}$ or $\boldsymbol{\omega}$ with the other variable fixed. In this paper, we design an alternating projection algorithm to find the solutions. More specifically, each time we fix one variable and then update the other one. The procedure is briefly outlined as follows:

- online $\boldsymbol{\beta}$ step: Fix $\boldsymbol{\omega}$, and apply Newton's method to update the parameter $\boldsymbol{\beta}$ by using the current $\boldsymbol{\beta}$ for initialization;
- online topic step: Fix $\boldsymbol{\beta}$, and minimize the problem in (2) to get the updated topic proportions $\boldsymbol{\omega}_k$ based on the current topic proportions $\boldsymbol{\theta}_k$.

The above procedure will be repeated for several iterations until some termination condition is satisfied. We can prove that the learning algorithm is convergent.

[2]Please note $L(\boldsymbol{\beta}, \boldsymbol{\omega})$ is different from $L(\boldsymbol{\beta})$. In $L(\boldsymbol{\beta})$, only $\boldsymbol{\beta}$ is a variable and $\boldsymbol{\omega}$ is a constant (fixed value).

Mini-batches: In OEM introduced above, every time a new paper i arrives, we can join it into the network and iterate between the online $\boldsymbol{\beta}$ step and the online topic step until it converges. However, it's computationally expensive for large-scale citation networks. A common technique is to consider multiple events per update. Not only can this kind of mini-batch strategy save computational cost, but it also reduces noise [Hoffman *et al.*, 2010]. Hence, in our implementation, rather than perform updating for each event, we perform one updating for every q citation events. q is set to about 1500 in our experiments.

The following content in this section will detail the algorithms for online $\boldsymbol{\beta}$ learning and online topic proportion learning.

3.1 Online β Step

With $\boldsymbol{\omega}$ fixed, the objective function to learn $\boldsymbol{\beta}$ is as follows:

$$L(\boldsymbol{\beta}) = \prod_{e=x}^{x+q-1} \frac{\exp(\boldsymbol{\beta}^T \mathbf{s}_{i_e}(t_e))}{\sum_{i=1}^{n} Y_i(t_e)\exp(\boldsymbol{\beta}^T \mathbf{s}_i(t_e))},$$

where x is the starting event in the mini-batch of q events.

To avoid walking through all the citation events when updating the parameter $\boldsymbol{\beta}$, we can use a training window to restrict the training in a small subset of the citation events. With the training window of width W_t ($1 \leq W_t \leq q$), $\boldsymbol{\beta}$ can be trained by optimizing:

$$L_w(\boldsymbol{\beta}) = \prod_{e=x+q-W_t}^{x+q-1} \frac{\exp(\boldsymbol{\beta}^T \mathbf{s}_{i_e}(t_e))}{\sum_{i=1}^{n} Y_i(t_e)\exp(\boldsymbol{\beta}^T \mathbf{s}_i(t_e))}.$$

Furthermore, we can cache the link features of each node to further cut the computational time, as done in [Vu *et al.*, 2011b].

3.2 Online Topic Step

In this section, we first formulate the *full online topic step* to learn the updated topic proportions $\boldsymbol{\omega}_k$. After that, we derive an *approximative online topic step* to speed up the optimization process.

Full Online Topic Step

It is very time-consuming if we update all the topic proportions in $\boldsymbol{\omega}$ at a time. We also design an alternating algorithm for updating $\boldsymbol{\omega}$. More specifically, each time we optimize the topic proportion for one paper, say $\boldsymbol{\omega}_k$, with all the other $\{\boldsymbol{\omega}_i | i \neq k\}$ fixed. Given a mini-batch of size q, if node k gets cited at citation event $e_1, e_2, \ldots, e_p$ and doesn't get cited at time $e_{p+1}, e_{p+2}, \ldots, e_q$ (note that e_2 does not necessarily happen before e_{p+2} although its subscript is smaller), the function $f(\boldsymbol{\omega}_k)$ we optimize is:

$$-\log\Big(\prod_{i=1}^{p} \frac{\alpha_i \exp(\mathbf{a}_i^T \boldsymbol{\omega}_k)}{A_i + \alpha_i \exp(\mathbf{a}_i^T \boldsymbol{\omega}_k)} \prod_{u=p+1}^{q} \frac{C_u}{B_u + \gamma_u \exp(\mathbf{b}_u^T \boldsymbol{\omega}_k)}\Big) + \lambda \|\boldsymbol{\omega}_k - \boldsymbol{\theta}_k\|_2^2, \tag{3}$$

where

$$\begin{aligned}
\alpha_i &= \exp(\boldsymbol{\beta}_l^T \mathbf{s}_k^l(t_{e_i})), \\
\gamma_u &= \exp(\boldsymbol{\beta}_l^T \mathbf{s}_k^l(t_{e_u})), \\
A_i &= \sum_{j \neq k} Y_j(t_{e_i}) \exp(\boldsymbol{\beta}^T \mathbf{s}_j(t_{e_i})), \\
B_u &= \sum_{j \neq k} Y_j(t_{e_u}) \exp(\boldsymbol{\beta}^T \mathbf{s}_j(t_{e_u})), \\
\mathbf{a}_i &= \boldsymbol{\beta}_t \circ \boldsymbol{\theta}_i, \\
\mathbf{b}_u &= \boldsymbol{\beta}_t \circ \boldsymbol{\theta}_u.
\end{aligned}$$

Here, $\boldsymbol{\beta}_l$ contains the first 8 elements of the parameter $\boldsymbol{\beta}$ (corresponding to link features), $\boldsymbol{\beta}_t$ contains the last 50 elements of $\boldsymbol{\beta}$ (corresponding to the topic features), $\boldsymbol{\theta}_i$ is the topic proportion of the citer at citation event e_i and $\mathbf{s}_k^l(t_{e_i})$ is the link features (first 8 features) of node k at citation event e_i, C_u is a constant irrelevant to $\boldsymbol{\omega}_k$.

The first and second order derivatives of (3) are as follows:

$$\begin{aligned}
\frac{\partial f}{\partial \boldsymbol{\omega}_k} =& -\sum_{i=1}^{p} \mathbf{a}_i + \sum_{i=1}^{p} \frac{\mathbf{a}_i \alpha_i \exp(\mathbf{a}_i^T \boldsymbol{\omega}_k)}{A_i + \alpha_i \exp(\mathbf{a}_i^T \boldsymbol{\omega}_k)} \\
&+ \sum_{u=p+1}^{q} \frac{\mathbf{b}_u \gamma_u \exp(\mathbf{b}_u^T \boldsymbol{\omega}_k)}{B_u + \gamma_u \exp(\mathbf{b}_u^T \boldsymbol{\omega}_k)} \\
&+ 2\lambda(\boldsymbol{\omega}_k - \boldsymbol{\theta}_k),
\end{aligned} \tag{4}$$

$$\begin{aligned}
\frac{\partial^2 f}{\partial \boldsymbol{\omega}_k^2} =& \sum_{i=1}^{p} \frac{A_i \alpha_i \mathbf{a}_i \mathbf{a}_i^T \exp(\mathbf{a}_i^T \boldsymbol{\omega}_k)}{(A_i + \alpha_i \exp(\mathbf{a}_i^T \boldsymbol{\omega}_k))^2} \\
&+ \sum_{u=p+1}^{q} \frac{B_u \gamma_u \mathbf{b}_u \mathbf{b}_u^T \exp(\mathbf{b}_u^T \boldsymbol{\omega}_k)}{(B_u + \gamma_u \exp(\mathbf{b}_u^T \boldsymbol{\omega}_k))^2} + 2\lambda \mathbf{I},
\end{aligned}$$

where $\mathbf{I}$ is an identity matrix.

It is easy to prove that the second order derivative (Hessian) is positive definite (PD). Hence, the function in (3) is convex. We can use some solver to find a global optimal solution.

Approximative Online Topic Step

In (4), A_i is far larger than $\mathbf{a}_i \alpha_i \exp(\mathbf{a}_i^T \boldsymbol{\omega}_k)$ and $\alpha_i \exp(\mathbf{a}_i^T \boldsymbol{\omega}_k)$, and p is relatively small in each batch. Similarly, B_u is far larger than $\mathbf{b}_u \gamma_u \exp(\mathbf{b}_u^T \boldsymbol{\omega}_k)$ and $\gamma_u \exp(\mathbf{b}_u^T \boldsymbol{\omega}_k)$, and $(q - p)$ is relatively small. Hence, the second and third terms in (4) are much smaller than the other two. Consequently, we can remove these two smaller terms to get the following approximative gradient:

$$\frac{\partial f}{\partial \boldsymbol{\omega}_k} \approx -\sum_{i=1}^{p} \mathbf{a}_i + 2\lambda(\boldsymbol{\omega}_k - \boldsymbol{\theta}_k).$$

Based on the above approximative gradient, we can recover the following approximative objective function of (2):

$$\begin{aligned}
&minimize \;\; -\sum_{i=1}^{p} \mathbf{a}_i^T \boldsymbol{\omega}_k + \lambda \sum_{k=1}^{n} \|\boldsymbol{\omega}_k - \boldsymbol{\theta}_k\|_2^2 \\
&subject\ to: \;\; \boldsymbol{\omega}_k \succeq \mathbf{0}, \mathbf{1}^T \boldsymbol{\omega}_k = 1.
\end{aligned} \tag{5}$$

We call the OEM variant in (5) *approximative OEM* and the original OEM in (2) *full OEM*. In our experiments, we find that the approximative OEM achieves accuracy comparable with that of full OEM with much less computational cost.

Table 1: Information of data sets

DATA SET	#PAPERS	#CITATIONS	#UNIQUE TIMES
ARXIV-TH	14226	100025	10500
ARXIV-PH	16526	125311	1591

3.3 Convergence Analysis

In each iteration, the learning algorithm ensures that the objective function value always decreases. Furthermore, the objective function is bounded below by 0. Hence, the learning algorithm will converge.

4 Experiment

We apply DEM and OEM to two citation networks and compare the results between them. Furthermore, we will also analyze the evolution of papers' topic proportions.

4.1 Data Sets

Since this paper mainly focuses on citation network analysis which is one of the most important applications of dynamic network analysis, we use two citation data sets, arXiv-TH and arXiv-PH, which are crawled from arXiv[3]. The general information of these data sets is summarized in Table 1.

The arXiv-TH data set is a collection of articles on high energy physics theory. It spans from 1993 to 1997 with high-resolution timestamps (millisecond resolution). The arXiv-PH data set is a collection of articles on high energy physics phenomenology. It spans from 1993 to 1997 on a daily scale. Since the resolution is high enough, we assume that every new paper joins the network at a unique time and obviously there can be more than one citation event happening at each unique time. As mentioned in the previous section, we update the topic proportions and parameters batch by batch. More specifically, we partition the data sets into mini-batches and each mini-batch contains citation events happening in a span of unique times. For arXiv-TH the number of unique times per batch is 100 and for arXiv-PH the number is 20. And the number of events for each mini-batch is about 1500.

4.2 Baseline

We choose the following four models for comparison in the experiments:

- **DEM**: The original DEM with 8 link features and 50 topic features. Note that the original DEM is not online and the parameters and topic features are fixed after training.
- **OEM-$\boldsymbol{\beta}$**: The OEM with only *online $\boldsymbol{\beta}$ step*, where the $\boldsymbol{\beta}$ will be adaptively updated while the topic features (topic proportions) of each paper will not change over time.
- **OEM-full**: The full OEM with both online $\boldsymbol{\beta}$ and topic steps, where both the topic features and parameters are adaptively learned over time using the objective function in (2).

[3]http://snap.stanford.edu/data

Table 2: Data set partition for building, training and testing.

DATA SETS	BUILDING	TRAINING	TESTING
ARXIV-TH	62239	1465	36328
ARXIV-PH	82343	1739	41229

- **OEM-appr**: The approximative OEM with both online $\boldsymbol{\beta}$ and topic steps, where both the topic features and parameters are adaptively learned over time using the approximative objective function in (5).

4.3 Evaluation Metrics

As in [Vu *et al.*, 2011b], we evaluate the above models with the following three metrics:

- *Average held-out log-likelihood:* By taking a logarithm of the likelihood $L(\boldsymbol{\beta})$ in (1) for each *testing* citation event, we can get the held-out log-likelihood. We divide the summation of log-likelihood over all testing events in a batch by the number of events in that batch to get the average held-out log-likelihood of that batch. A higher average held-out log-likelihood indicates a better testing accuracy.
- *Recall of top-K recommendation list:* The recall is defined as the fraction of true citation events in the top K (measured using likelihood) possible citation events. Here K is the cut-point.
- *Average held-out normalized rank:* The "rank" for each citation event is the position of the true citation in the sorted list (in decreasing order of likelihoods) of possible citations. The rank is then normalized by the number of possible citations. A lower rank indicates a better predictive power.

4.4 Results and Analysis

As in DEM [Vu *et al.*, 2011b], we split each data set into three parts which are used for the building phase, training phase and testing phase, respectively. The building phase is aimed to build up the statistics of the citation networks and it is relatively long in order to mitigate the truncation effect (citation events happening before 1993 is not contained in the data sets) and avoid biases. In the training phase, we train the initial model parameters and the topic features. To fully demonstrate and compare the predictive power of these models, we have a relatively long testing phase and the testing phase is divided into 24 batches. Please remember that the statistics (link features) are dynamically changed in both training and testing phases. The sizes (measured by the numbers of citation events) of each phase are listed in Table 2.

To further cut the time cost of OEM, we randomly choose a fraction of unique times in every batch to optimize the topic proportions. For example, when optimizing the topic proportions for paper i after the first batch comes, we randomly choose 10% (here 10% is what we call *citer percentage* in the rest of this paper) of citers instead of the whole set of them. This can be used to speed up the computation. In OEM, we set the hyperparameters $\lambda = 0.1$ and *citer percentage* = 10 unless otherwise stated. The effect of these hyperparameters (*citer percentage* and λ) will be detailed in later experiments.

The testing procedure of OEM is detailed as follows. We first train an initial OEM using the building and training data sets. Hence, this initial OEM is actually equivalent to DEM. Then, we evaluate the predictive power on Batch 1 of the testing set (note that we don't use the data of Batch 1 during the training). After that (testing Batch 1), we absorb Batch 1 as extra training data and update OEM with the online learning algorithms. Then, we use the updated OEM to predict Batch 2. That is to say, to test a specific batch, we DO NOT use any data from this batch for training. Hence, the testing results of OEM will truly reflect the generalization/predictive ability of OEM.

Figure 1 (a) and (b) show the average held-out log-likelihood for all the models. Because the initial OEM is equivalent to DEM, we can see that all the models have the same performance on testing Batch 1. However, as time goes on, the performance of DEM will dramatically decrease while all the OEM variants can prevent the performance from decreasing. For example, from Figure 1 (a), we can see that the log-likelihood of the original DEM decreases significantly over time and the log-likelihood of OEM-$\boldsymbol{\beta}$ decreases only from -8.24 to -8.97. Our OEM-full outperforms the previous two with the log-likelihood ranging from -7.89 to -8.38 and the log-likelihood of OEM-appr decreases only from -8.24 to -8.56.

Figure 1 (c) and (d) show the recall of top-K recommendation list with K=250. We can find that the performance of DEM, OEM-$\boldsymbol{\beta}$, and OEM-appr decreases over time. However, our OEM-full can prevent the performance from decreasing. Although the performance of OEM-appr also decreases over time, it still outperforms DEM. The performance of OEM-$\boldsymbol{\beta}$ is as bad as that of DEM, which implies that the topic features are very informative and it is not enough to learn only $\boldsymbol{\beta}$ for this metric. Please note similar results can be observed when K takes other values. Here, we omit those results due to space limitation.

Figure 1 (e) and (f) show the average held-out normalized rank. We find that the performance of DEM and OEM-$\boldsymbol{\beta}$ can not be improved over time. However, the performance of OEM-full and OEM-appr will be improved as time goes on. Note that a lower rank indicates a better predictive power. Once again, the bad performance of OEM-$\boldsymbol{\beta}$ indicates the importance of topic features for this metric. Because the number of possible citations for later batches are larger than that of former batches, the performance in terms of absolute rank of DEM actually decreases over time. But OEM-full can present the performance in terms of absolute rank from decreasing. This conforms to the results in Figure 1 (a), (b), (c), and (d).

Table 3 compares the computation cost between full OEM and approximative OEM. We can see that although the approximative OEM achieves a slightly worse predictive accuracy than full OEM, it saves more than 50% of the time.

To measure how the hyperparamters, *citer percentage* and λ, affect the predictive performance, we use the arXiv-TH data set and compute the average log-likelihood of all testing batches for each choice of citer percentage and λ. The results are shown in Table 4 and Table 5. Table 4 indicates that 0.1 is the best value for λ. Table 5 shows that the predictive performance increases moderately and the time cost increases significantly when the citer percentage is larger than 10%, which means that 10% may be a good choice of the citer percentage. Overall, our model is not sensitive the these hyperparameters.

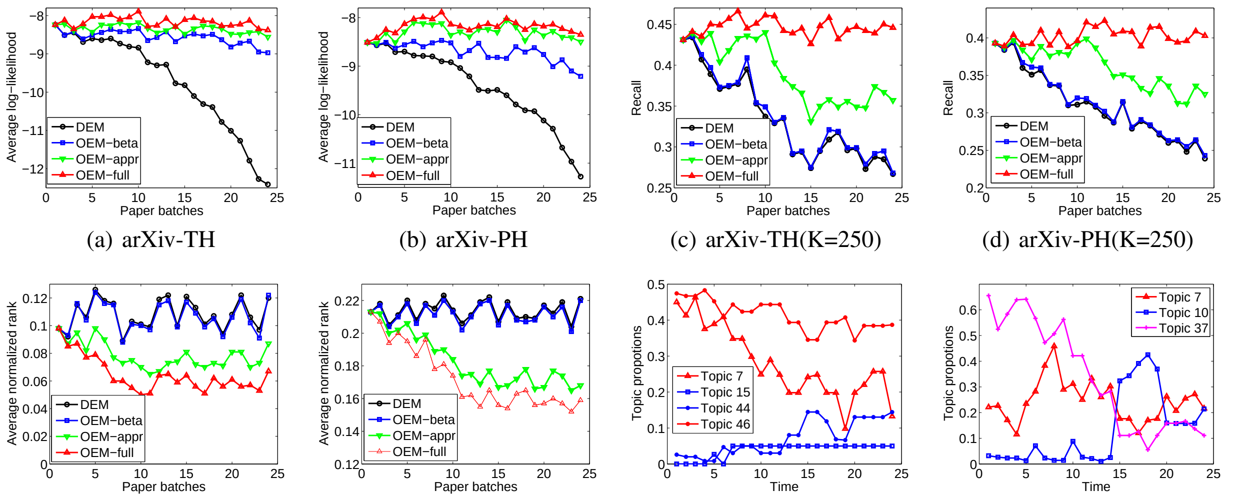
To show the topic evolution of papers, we choose three sets of papers in arXiv-TH. To avoid clutter we take the average topic proportions of each paper set and we select the topics (elements of vectors) which have the highest average proportion on all 24 time periods to demonstrate the effect of topic evolution. Specifically, we use $S_t = \{r_1, r_2, \dots, r_l\}$ to denote the set of papers cited at unique time t (thus papers in the same set are cited by the same paper). And we denote $\phi_t = \frac{1}{l}\sum_{i=1}^{l} \boldsymbol{\omega}_{r_i}$ to be the average topic vector of paper set S_t. We choose S_{8001} and S_{8005} to be our examples, which are shown in Figure 1 (g) and (h).

From Figure 1 (g), we can see that proportions of Topic 7 (namely $\phi_{8001}^{(7)}$) and Topic 46 (namely $\phi_{8001}^{(46)}$) are decreasing over time. Yet proportions of Topic 15 ($\phi_{8001}^{(15)}$) and Topic 44 ($\phi_{8001}^{(44)}$) show the opposite tendency. One explanation is that the set of papers cited at the 8001st unique time is originally on some sub-fields of physics, but as time goes on, the values of the papers are found by researchers in other sub-fields. Citations by papers of other sub-fields will transfer some proportions from the original topics (Topic 7 and Topic 46) to the new topics (Topic 15 and Topic 44). The same thing happens in the fields of statistics, phycology (as original topics) and machine learning (as the newly arising topic). The topic evolution of the paper set at the 8005th unique time (namely S_{8005}) is similar to the one at the 8001st unique time, as shown in Figure 1 (h).

5 Related Work

Dynamic network analysis (DNA) has been widely studied by researchers from a diversity of fields such as social networks, citation networks and email networks [Leskovec *et al.*, 2005; Kossinets and Watts, 2006; Viswanath *et al.*, 2009]. However, most existing works [Sarkar and Moore, 2005; Hanneke *et al.*, 2010; Fu *et al.*, 2009; Wyatt *et al.*, 2010; Foulds *et al.*, 2011; Ho *et al.*, 2011] focus either on small networks at a fine temporal granularity or on large-scale networks at a rough temporal granularity. Although DEM [Vu *et al.*, 2011b] is able to handle large-scale networks at a fine temporal granularity, its parameters remain static, which leads to an obvious decrease of accuracy over time. Continuous-time regression models for longitudinal networks [Vu *et al.*, 2011a] allow the parameter to be updated over time. However they are used to model edges rather than nodes. Furthermore, they do not take the topic information of papers into consideration. But in our proposed OEM, topic information is effectively integrated into the model's building, training and testing phases.

Another line of research related to our work is about topic models. By extending the original LDA topic model [Blei *et al.*, 2003], many methods have been proposed to model the topic evolution over time [Wang *et al.*, 2008; 2009; Chen *et al.*, 2012; Dubey *et al.*, 2013]. There also exist some methods which can model both network structure



(a) arXiv-TH (b) arXiv-PH (c) arXiv-TH(K=250) (d) arXiv-PH(K=250)

(e) arXiv-TH (f) arXiv-PH (g) Paper sets cited at the 8001st unique time (h) Paper sets cited at the 8005th unique time

Figure 1: (a) and (b) are the average held-out log-likelihood of testing citation events. (c) and (d) are the recall of top-K recommendation lists. (e) and (f) are the average held-oud normalized ranks of testing citation events. Since all models have the same initial parameters after the building and training phases, all models have the same performance on the first testing batch, which can be seen from (a) to (f). (g) and (h) are the topic evolution of sets of papers cited at the 8001st and 8005th unique time. To avoid clutter, we only show the topics with the largest proportions (top topics).

Table 3: Computation time (in seconds) of OEM-full and OEM-appr with $\lambda = 0.1$.

CITER PERCENTAGE	2%	5%	10%	20%	30%	50%	100%
OEM-FULL	0.13	0.43	0.87	1.42	1.96	2.61	3.91
OEM-APPR	0.06	0.22	0.41	0.70	0.95	1.29	1.94

Table 4: Average held-out log-likelihood when citer percentage is 10%

λ	10^{-4}	0.01	0.1	0.5	1	2	10^4
LOG-LIKELIHOOD	-8.61	-8.33	-8.15	-8.28	-8.33	-8.35	-8.56

Table 5: Average held-out log-likelihood when $\lambda = 0.1$

CITER PERCENTAGE	2%	5%	10%	20%	30%	50%	100%
LOG-LIKELIHOOD	-8.94	-8.43	-8.15	-8.10	-8.09	-8.03	-7.98
AVERAGE TIME	0.13	0.43	0.87	1.42	1.96	2.61	3.91

and node features [Kataria *et al.*, 2011; Hu *et al.*, 2012; Krafft *et al.*, 2012]. Generally, it is very time-consuming to simultaneously update the topics and topic proportions for time-varying data.

Instead of utilizing some existing online LDA models [Canini *et al.*, 2009; Hoffman *et al.*, 2010], we choose to directly adjust the topic proportions of papers. This is because online inference of LDA interacts with the text contents of the papers, which will take a lot more time to update all the LDA vectors. However in our OEM, we only need to solve small convex optimization problems to update the vectors.

6 Conclusion

In this paper, an online egocentric model (OEM) is proposed for evolving citation network modeling. By adaptively learning the parameters and topic features over time, OEM has successfully overcome the problem of DEM whose predictive accuracy will decrease significantly over time. Experimental results on real-world citation networks demonstrate that OEM can achieve very promising performance in real applications.

Although the experiments in this paper are only for paper citation networks, as stated in [Vu *et al.*, 2011b], our model can be generalized to other types of networks, which will be pursued in our future work.

7 Acknowledgements

This work is supported by the NSFC (No. 61100125), the 863 Program of China (No. 2011AA01A202), and the Program for Changjiang Scholars and Innovative Research Team in University of China (IRT1158, PCSIRT).

References

[Blei *et al.*, 2003] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:992–1022, 2003.

[Canini *et al.*, 2009] Kevin Robert Canini, Lei Shi, and Thomas L. Griffiths. Online inference of topics with latent dirichlet allocation. In *AISTATS*, 2009.

[Chen *et al.*, 2012] Changyou Chen, Nan Ding, and Wray L. Buntine. Dependent hierarchical normalized random measures for dynamic topic modeling. In *ICML*, 2012.

[Dubey *et al.*, 2013] Avinava Dubey, Ahmed Hefny, Sinead Williamson, and Eric P. Xing. A non-parametric mixture model for topic modeling over time. In *SDM*, 2013.

[Foulds *et al.*, 2011] James R. Foulds, Christopher DuBois, Arthur U. Asuncion, Carter T. Butts, and Padhraic Smyth. A dynamic relational infinite feature model for longitudinal social networks. In *AISTATS*, 2011.

[Fu *et al.*, 2009] Wenjie Fu, Le Song, and Eric P. Xing. Dynamic mixed membership blockmodel for evolving networks. In *ICML*, 2009.

[Goldenberg *et al.*, 2009] Anna Goldenberg, Alice X. Zheng, Stephen E. Fienberg, and Edoardo M. Airoldi. A survey of statistical network models. *Foundations and Trends in Machine Learning*, 2:129–233, 2009.

[Hanneke *et al.*, 2010] Steve Hanneke, Wenjie Fu, and Eric P. Xing. Discrete temporal models of social networks. *Electronic Journal of Statistics*, 4:585–605, 2010.

[Ho *et al.*, 2011] Qirong Ho, Le Song, and Eric P. Xing. Evolving cluster mixed-membership blockmodel for time-evolving networks. In *AISTATS*, 2011.

[Hoffman *et al.*, 2010] Matthew D. Hoffman, David M. Blei, and Francis R. Bach. Online learning for latent dirichlet allocation. In *NIPS*, 2010.

[Hu *et al.*, 2012] Yuheng Hu, Ajita John, Fei Wang, and Subbarao Kambhampati. ET-LDA: joint topic modeling for aligning events and their twitter feedback. In *AAAI*, 2012.

[Jin *et al.*, 2011] Yingzi Jin, Ching-Yung Lin, Yutaka Matsuo, and Mitsuru Ishizuka. Mining longitudinal network for predicting company value. In *IJCAI*, 2011.

[Kataria *et al.*, 2011] Saurabh Kataria, Prasenjit Mitra, Cornelia Caragea, and C. Lee Giles. Context sensitive topic models for author influence in document networks. In *IJCAI*, 2011.

[Kim and Leskovec, 2012] Myunghwan Kim and Jure Leskovec. Latent multi-group membership graph model. In *ICML*, 2012.

[Kossinets and Watts, 2006] Gueorgi Kossinets and Duncan J Watts. Empirical analysis of an evolving social network. *Science*, 311(5757):88–90, 2006.

[Krafft *et al.*, 2012] Peter Krafft, Juston Moore, Hanna Wallach, and Bruce Desmarais. Topic-partitioned multinetwork embeddings. In *NIPS*, 2012.

[Leskovec *et al.*, 2005] Jure Leskovec, Jon M. Kleinberg, and Christos Faloutsos. Graphs over time: densification laws, shrinking diameters and possible explanations. In *ACM SIGKDD*, 2005.

[Li and Yeung, 2009] Wu-Jun Li and Dit-Yan Yeung. Relation regularized matrix factorization. In *IJCAI*, 2009.

[Li and Yeung, 2012] Wu-Jun Li and Dit-Yan Yeung. Sparse probabilistic relational projection. In *AAAI*, 2012.

[Li *et al.*, 2009a] Wu-Jun Li, Dit-Yan Yeung, and Zhihua Zhang. Probabilistic relational PCA. In *NIPS*, 2009.

[Li *et al.*, 2009b] Wu-Jun Li, Zhihua Zhang, and Dit-Yan Yeung. Latent wishart processes for relational kernel learning. In *AISTATS*, 2009.

[Li *et al.*, 2011] Wu-Jun Li, Dit-Yan Yeung, and Zhihua Zhang. Generalized latent factor models for social network analysis. In *IJCAI*, 2011.

[McAuley and Leskovec, 2012] Julian J. McAuley and Jure Leskovec. Learning to discover social circles in ego networks. In *NIPS*, 2012.

[Myers *et al.*, 2012] Seth A. Myers, Chenguang Zhu, and Jure Leskovec. Information diffusion and external influence in networks. In *ACM SIGKDD*, 2012.

[Nori *et al.*, 2011] Nozomi Nori, Danushka Bollegala, and Mitsuru Ishizuka. Interest prediction on multinomial, time-evolving social graph. In *IJCAI*, 2011.

[Richard *et al.*, 2012] Emile Richard, Stephane Gaiffas, and Nicolas Vayatis. Link prediction in graphs with autoregressive features. In *NIPS*, 2012.

[Sarkar and Moore, 2005] Purnamrita Sarkar and Andrew W. Moore. Dynamic social network analysis using latent space models. *SIGKDD Explorations*, 7(2):31–40, 2005.

[Sarkar *et al.*, 2012] Purnamrita Sarkar, Deepayan Chakrabarti, and Michael I. Jordan. Nonparametric link prediction in dynamic networks. In *ICML*, 2012.

[Snijders, 2005] Tom A B Snijders. Models for longitudinal network data. *Models and Methods in Social Network Analysis*, pages 215–247, 2005.

[Viswanath *et al.*, 2009] Bimal Viswanath, Alan Mislove, Meeyoung Cha, and P. Krishna Gummadi. On the evolution of user interaction in facebook. In *ACM Workshop on Online Social Networks*, 2009.

[Vu *et al.*, 2011a] Duy Vu, Arthur U. Asuncion, David Hunter, and Padhraic Smyth. Continuous-time regression models for longitudinal networks. In *NIPS*, 2011.

[Vu *et al.*, 2011b] Duy Vu, Arthur U. Asuncion, David Hunter, and Padhraic Smyth. Dynamic egocentric models for citation networks. In *ICML*, 2011.

[Wang and Groth, 2011] Shenghui Wang and Paul T. Groth. A framework for longitudinal influence measurement between communication content and social networks. In *IJCAI*, 2011.

[Wang *et al.*, 2008] Chong Wang, David M. Blei, and David Heckerman. Continuous time dynamic topic models. In *UAI*, 2008.

[Wang *et al.*, 2009] Chong Wang, Bo Thiesson, Christopher Meek, and David M. Blei. Markov topic models. In *AISTATS*, 2009.

[Wang *et al.*, 2010] Chi Wang, Jiawei Han, Yuntao Jia, Jie Tang, Duo Zhang, Yintao Yu, and Jingyi Guo. Mining advisor-advisee relationships from research publication networks. In *ACM SIGKDD*, 2010.

[Wasserman, 1980] Stanley Wasserman. Analyzing social networks as stochastic processes. *Journal of the American Statistical Association*, 75(370):280–294, 1980.

[Wyatt *et al.*, 2010] Danny Wyatt, Tanzeem Choudhury, and Jeff Bilmes. Discovering long range properties of social networks with multi-valued time-inhomogeneous models. In *AAAI*, 2010.

[Zhu, 2012] Jun Zhu. Max-margin nonparametric latent feature models for link prediction. In *ICML*, 2012.