Dynamic Mixture Models for Multiple Time Series

Xing Wei
Computer Science Department
Univeristy of Massachusetts
Amherst, MA 01003
xwei@cs.umass.edu

Jimeng Sun
Computer Science Department
Carnegie Mellon University
Pittsburgh, PA 15213
jimeng@cs.cmu.edu

Xuerui Wang
Computer Science Department
Univeristy of Massachusetts
Amherst, MA 01003
xuerui@cs.umass.edu

Abstract

Traditional probabilistic mixture models such as Latent Dirichlet Allocation imply that data records (such as documents) are fully exchangeable. However, data are naturally collected along time, thus obey some order in time. In this paper, we present Dynamic Mixture Models (DMMs) for online pattern discovery in multiple time series. DMMs do not have the noticeable drawback of the SVD-based methods for data streams: negative values in hidden variables are often produced even with all nonnegative inputs. We apply DMM models to two real-world datasets, and achieve significantly better results with intuitive interpretation.

1 Introduction

Multiple co-evolving time series or data streams are ubiquitous in many different real-world applications. Considering a sensor network, multiple sensors continuously collect different measurements over time (e.g., chlorine concentration at different locations in a water distribution system; temperature or light measurements of different rooms; host status of different machines in a data center; etc.). A central site usually analyzes those measurements for main trends and anomaly detection. There has been success in mining multiple streams. However, the existing methods on analyzing such streams still have significant weaknesses:

First, existing monitoring softwares require considerable time and expertise to be properly configured despite the fact they treat the streams as independent. For each data stream that an administrator intends to monitor, he/she must make decisions upon proper thresholds for the data values. That is, he/she must define, for each data stream, what constitutes normal behaviors. The correlations across streams are usually not captured.

Second, the state-of-the-art algorithms on automatically summarizing multiple streams often adopt matrix decompositions, like Singular Value Decomposition (SVD) and its variants. For example, Papadimitriou et al. uses online SVD tracking to summarize the multiple streams incrementally through a small number of hidden variables [Papadimitriou *et al.*, 2005]. The hidden variables computed there are linear combinations of all streams projecting onto the maximum variance directions. This construction often seems not intuitive to the end users due to the reification of the mathematical properties of the SVD techniques. In particular, some streams may have negative coefficients for hidden variables even when all the measurements are nonnegative. This drawback is intrinsic due to the Gaussian distribution assumption on the input streams.

Using mixture of hidden variables for data representation has recently been of considerable interest in text modeling. One early example is the Latent Semantic Indexing (LSI) model, which is also based on the SVD technique. More recently, Hoffman presents the probabilistic Latent Semantic Indexing (pLSI) technique [Hoffman, 1999]. This approach uses a latent variable model that represents documents as mixtures of topics in a probabilistic framework. Latent Dirichlet Allocation (LDA) [Blei *et al.*, 2003], which resembles the generative process of pLSI but overcomes some of its drawbacks, has quickly become one of the most popular probabilistic text modeling techniques in machine learning and has inspired a series of research works in this direction [Girolami and Kaban, 2005; Teh *et al.*, 2004]. In particular, LDA has been shown to be effective in some text-related tasks such as document classification [Blei *et al.*, 2003] and information retrieval [Wei and Croft, 2006], but its effectiveness on continuous data and time-series remains mostly unknown.

Most topic models mentioned above assume that data records (e.g., documents) are fully exchangeable, which are not true in time-series model from many real-world applications. Exactly because of that, topic modeling over time starts to receive more and more attentions. In the recent Dynamic Topic Models [Blei and Lafferty, 2006], topic evolutions are modeled through collections sliced into certain periods of time. However, the time dependency of individual data records (documents) inside a collection/period are not considered. In the Topic over Time (TOT) model [Wang and McCallum, 2006], continuous time stamps are put into an LDA-style topic model as observations. But drawing time stamps from one distribution as in the TOT model is often not enough for dealing with bursty data, common in data streams.

In this paper, we present a Dynamic Mixture Model (DMM), a latent variable model that takes into consideration the time stamps of data records in dynamic streams. The values in streams are represented as mixtures of hidden variables, and for each time stamp, the mixture of hidden variables for all streams is dependent on the mixture of the previous time stamp. In this way we can capture the short-term dependencies. Compared to the SVD techniques, it does not have their common drawback: non-intuitive negative values of hidden variables, and thus has a more interpretable explanation within a probabilistic framework. Compared to the static LDA-style models, it can take advantage of the time dependency of multiple streams. Finally, we want to point out that topic models have been mainly designed for discrete data. To apply the LDA-style mixture models to continuous data, the standard way is discretization, which we show in Section 4 works well in DMM.

2 Related Work

Data streams, containing much more interesting information than static data, have been extensively studied in recent years. Evidence has shown that temporal information plays a crucial role in capturing many meaningful and interesting patterns. The main objective of temporal analysis is to efficiently discover and monitor patterns when data are streaming into a system. A recent survey [Muthukrishnan 2005] has discussed many data streams algorithms. Among of these, SVD and PCA are popular tools for summarizing multiple time series into a number of hidden variables [Papadimitriou *et al.*, 2005]. However, it assumes independency across time stamps; the results are often lack of probabilistic argument and intuitive interpretation.

Probabilistic mixture models [Hoffman, 1999; Blei *et al.*, 2003] are widely used to summarize data based on statistical frameworks. To capture time evolution, the usage of time within probabilistic mixture models has been around for a while. For example, time can be taken care of in a post-hoc way. One could first fit a time-unaware mixture model, and then order the data in time, slice them into discrete subsets, and examine the mixture distributions in each time-slice [Griffiths and Steyvers, 2004]. Alternatively, non-joint modeling can also pre-divide the data into discrete time slices, fit a separate mixture model in each slice and possibly align the mixture components across slices [Wang *et al.*, 2006].

More systematic ways to take advantage of the temporal information in mixture models can be put into two categories. First, the dynamic behaviors are driven by state transitions. This kind of models generally make a Markov assumption, i.e., the state at time $t + 1$ or $t + \Delta t$ is independent of all other history given the state at time t. For instance, Blei and Lafferty present dynamic topic models (DTMs) in which the alignment among topics across collections of documents is modeled by a Kalman filter [Blei and Lafferty, 2006]. DTMs target on topic evolution over sets of large amounts of data records and local dependencies between two records are not captured. Second, the other type of models does not employ a Markov assumption over time, but instead treats time as an observed continuous variable such as the topics over time (TOT) models [Wang and McCallum, 2006]. This helps capture long-range dependencies in time, and also avoids a Markov model's risk of inappropriately dividing a mixture component (topic) in two when there is a brief gap in its appearance. However, with all time stamps drawn from one distribution, TOT is not appropriate for data streams, which are usually bursty and multi-modal.

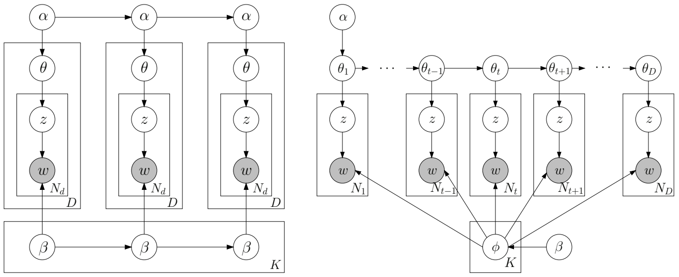

Figure 1: Graphical model representation of Dynamic Topic Model (left, 3 slices) and Dynamic Mixture Model (right)

SYMBOL	DESCRIPTION
K	number of hidden components / topics
D	number of snapshots / documents
V	number of parameters / words
N_t	sum of parameter values in the tth snapshot / number of words in the tth document
t	snapshot index / time stamp of documents
θ	mixture distribution of hidden components for a snapshot / document
ϕ	mixture distribution over parameters / words for a hidden component /topic
z	hidden component / topic
w	one unit value of a parameter / word token
α	hyperparameter for the multinomial θ_1
β	hyperparameter for ϕ

Table 1: Notation correspondence

3 Dynamic Mixture Models

Mixture models have been widely used in modeling various complex data with very competitive results. But the usage of latent mixture model in streaming data has not been explored, and largely remains unknown. We now present a dynamic mixture model (DMM) incorporating temporal information in data.

The graphical model representation of the DMM is shown in Figure 1 (right). As other mixture models for discrete data, the DMM can be regarded as a topic model as well. For presentation convenience, we explain this model in the background of both text and the streaming data (Section 4 describes the streaming data in detail). Table 1 summarizes the notation correspondences for both text and data streams. In particular, each snapshot corresponds to a document in topic models; each stream corresponds to a word occurrence over time, e.g., the number of occurrences of word w_i over time represents a single stream w_i, which could be the chlorine concentrations over time, or the light intensities over time of ith sensor.

The generative process of the DMM is similar to Latent Dirichlet Allocation [Blei *et al.*, 2003], but the mixture distribution θ for each document does not have a Dirichlet prior (except the very first one); instead, θ_t is dependent on the mixture distribution of the previous snapshot θ_{t-1}.

First, we assume there are strong evolution dependencies. The continuous stream data are evenly sampled in time as successive snapshots that reflect the intrinsic dependencies between the values along time series.

Second, the time dependency is complicated and the changes hardly follow a continuous time distribution as in the TOT model [Wang and McCallum, 2006]. Thus neighboring dependency (Markov assumption) is more appropriate.

Compared to the TOT model, the Dynamic Mixture Model is constructed on discrete time stamps and assumes dependencies between two consecutive snapshots. Although the TOT model captures both short-term and long-term changes by treating time stamps as observed random variables, DMM is capable to capture more detailed changes and to model the dependency between any two consecutive time shots, which is especially appropriate for many streaming data when the data are equally sampled in time.

Compared to Dynamic Topic Models (DTM, see Figure 1, left), DMM captures the evolution between snapshots (documents) instead of between snapshot-groups (collections in text model). In both DTM and LDA, documents in a collection and words in a document are fully exchangeable; in DMM, snapshots of multiple time series, which correspond to documents in text model, have very strong temporal order and exchanges of snapshots can lead to very different results. From this perspective, DMM is a true online model. Note that SVD also treats time series as vectors, thus the permutation of snapshots will not make a difference.

We model the dependency by setting the expectation of the distribution of θ_t to be θ_{t-1}, i.e., θ_t is generated from a distribution with θ_{t-1} as the 1st order moment (we will discuss the concrete distribution in Section 3.1). The process of generating time series streams in DMM is as follows:

1. Pick a multinomial distribution ϕ_z for each topic (hidden dimension) z from a Dirichlet distribution with parameter β ;
2. For time shot $t = 0$, sample a multinomial distribution θ_t from a Dirichlet distribution with parameter α,
3. For each time shot $t > 0$, sample a multinomial distribution θ_t from a distribution with expectation θ_{t-1},
4. Sample a hidden variable / topic $z \in \{1, \cdots, K\}$ from a multinomial distribution with parameter θ_t,
5. Add a unit value to parameter w picked from a multinomial distribution with parameter ϕ_{z_w} / Pick a word w from a multinomial distribution with parameter ϕ_{z_w}.

Thus, the likelihood of generating a data set of multiple data streams is:

$$P(\mathrm{Snapshot}_1, \cdots, \mathrm{Snapshot}_D | \alpha, \beta) = \iint \prod_{z=1}^{K} p(\phi_z|\beta)$$

$$\times p(\theta_1|\alpha) \prod_{t=2}^{D} p(\theta_t|\theta_{t-1}) \prod_{t=1}^{D} \prod_{i=1}^{N_t} \sum_{z_i=1}^{K} \left(P(z_i|\theta) P(w_i|\phi_{z_i})\right) \mathrm{d}\theta \mathrm{d}\phi$$

3.1 Dirichlet Dynamic Mixture Models

As we have described, θ_t is generated from a distribution with θ_{t-1} as the expectation. There are no strict requirements for this distribution, and we just want to build up connections between two successive snapshots, i.e., the mixture distribution of the current snapshot is dependent on the one of its previous snapshot. In this setting, a continuous one-modal distribution is desired. Gaussian distribution is a straightforward selection when θ is represented by natural parameters, which was adopted by [Blei and Lafferty, 2006] to model the dependency between two consecutive α (Figure 1, left). However, Gaussian distribution is not conjugate to the multinomial distribution that is used to generate hidden components / topics, which makes inference more difficult. Although, to the best of our knowledge, there is no conjugate prior for Dirichlet distribution; Dirichlet distribution is conjugate to multinomial distribution; and we use a Dirichlet distribution to model the dependency between consecutive θ. Expectation only is not enough to make the proposed distribution *identifiable*, since there are infinitely many Dirichlet distribution having the same expectation. Thus we introduce another *precision* constraint that all the parameters of the Dirichlet distributions sum to ψ that is also equal to the sum of all parameters α in the first Dirichlet distribution. A Dirichlet distribution can be completely parameterized by the mean and precision parameters [Minka, 2003]. For notation convenience, we define $\theta_0 = \alpha/\psi$. That is, $\theta_t|\theta_{t-1} \sim \mathrm{Dir}(\psi\theta_{t-1})$.

Inference

Inference can not be done exactly in complex graphical models such as DMMs. The non-conjugacy between Dirichlet distributions makes standard Gibbs sampling methods [Geman and Geman, 1994] harder in approximate inference of DMMs. Here, we use a simple, but effective iterated sampling procedure considering that streaming data are very different from traditional data sets such as large text collections: first, massive amounts of data arrive at high rates, which makes efficiency the most concern; second, users, or higher-level applications, require immediate responses and cannot afford any post-processing (e.g., in network intrusion detection) [Papadimitriou *et al.*, 2005]. In summary, the algorithm is expected to be efficient, incremental, scalable, and possibly scaling linearly with the number of streams. As shown in [Griffiths and Steyvers, 2004], we define $m_{z,w}$ to be the number of tokens of word w assigned to topic z, and the posterior distribution of ϕ_z can be approximated by

$$\hat{\phi}_{z,w} = \frac{m_{z,w} + \beta_w}{\sum_{w=1}^{V} (m_{z,w} + \beta_w)}.$$

To estimate the posterior distribution of θ_t in streaming data, we use a technique that is commonly used in mean-field variational approximation: assume each latent variable to be independent to the others. Then we use an iterative procedure to update all the θ periodically. We define $n_{t,z}$ to be the number of tokens in document t assigned to topic z. Thus, we have,

$$\hat{\theta}_{t,z} = \frac{n_{t,z} + \psi\hat{\theta}_{t-1,z}}{\sum_{z=1}^{K} (n_{t,z} + \psi\hat{\theta}_{t-1,z})}.$$

In each sample, we draw in turn $z_{t,i}$ according to a probability proportional to $\hat{\theta}_{t,z_{t,i}} \times \hat{\phi}_{z_{t,i},w_{t,i}}$, and update $\hat{\theta}$ after each iteration.

Streams are very special data and require the corresponding algorithms to be highly efficient in order to react in a timely manner. It could be certainly possible to derive a variational approximation from the scratch, however, we have empirically found that our procedure is very effective and efficient for streaming data presented in Section 4. With this iterated sampling algirithm, we keep the information from offline training, and run sampling only for the coming snapshot with a couple of iterations. The parameter estimation is updated after each new snapshot arrives.

4 Experiments

In this section we present case studies on real and realistic datasets to demonstrate the effectiveness of our approach in discovering the underlying correlations among streams.

4.1 Chlorine Concentrations

Description: The Chlorine dataset was generated by EPANET2.0[1] that accurately simulates the hydraulic and chemical phenomena within drinking water distribution systems. Given a network as the input, EPANET tracks the flow of water in each pipe, the pressure at each node, the height of water in each tank, and the concentration of a chemical species throughout the network, during a simulation period comprised of multiple time stamps. We monitor the chlorine concentration level at all the 166 junctions in the network shown in Figure 5 for 4310 time stamps during 15 days (one time tick every five minutes). The data was generated by using the input network with the demand patterns, pressures, flows specified at each node.

Data characteristics: The two key features are: 1) A clear global periodic pattern (daily cycle, dominating residential demand pattern). Chlorine concentrations reflect this, with few exceptions. 2) A slight time shift across different junctions, which is due to the time it takes for fresh water to flow down the pipes from the reservoirs.

Thus, most streams exhibit the same sinusoidal-like pattern, except with gradual phase shifts as we go further away from the reservoir.

Reconstruction: Our method can successfully summarize the data using just two numbers (hidden variables) per time tick (see Figure 3), as opposed to the original 166 numbers. Figure 2 shows the reconstruction for one sensor (out of 166). The reconstructions of the other sensors achieve similarly results. Note that only two hidden variables give very good reconstruction.

Hidden variables: The two hidden variables (see Figure 3) reflect the two key dataset characteristics: 1) The first hidden variable captures the global, periodic pattern; 2) The second one also follows a very similar periodic pattern, but with a slight phase shift.

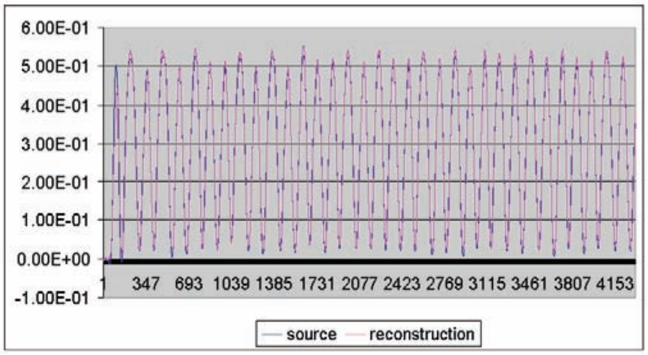

Figure 2: Chlorine reconstruction; reconstruction based on 2 hidden variables are very close to the original value.

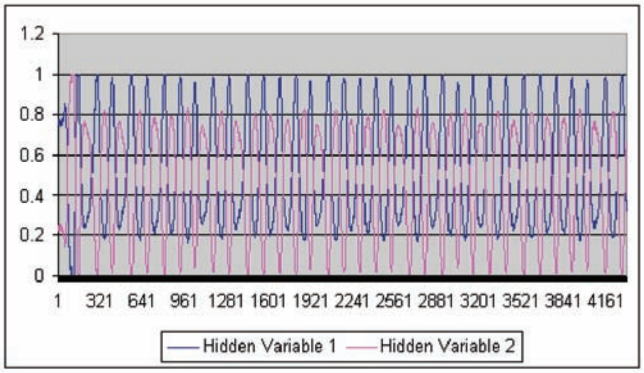

Figure 3: Hidden variables for chlorine data; the first captures the daily cycles; the second reflects the time shift of different sensors.

Interpretation of hidden variables: Each hidden variable follows a multinomial distribution with 166 parameters with the constraint that the probability mass over all parameters equals one. Unlike the SVD where a hidden variable is a linear combination of all streams, our method interprets the hidden variable as a generating process behind all 166 streams which is more intuitive (see Figure 4). Notice that there is the high probability mass on stream 1-7 because those sensors (highlighted on Figure 5) are close to the reservoir and on the main pipe of the water distribution network.

4.2 Light Measurements

Description: This dataset consists of light intensity measurements collected using Berkeley Mote sensors, at several different locations in a lab (see Figure 6) over a period of a month [Deshpande *et al.*, 2004]. We simulate streams with this dataset by cutting the set into two even parts, and process the snapshots in the second part one by one in an online manner with the trained model from the first part. The results we present of this dataset are all from stream simulation running.

Data characteristics: The main characteristics are: 1) A clear global periodic pattern (daily cycle). 2) Occasional big spikes from some sensors (outliers). Reconstruction: Similar to the chlorine data, the reconstruction error on light data is also very small.

Hidden variable: The first hidden variable exhibits the daily periodicity as shown in Figure 8. The probability distri-

[1] http://www.epa.gov/ORD/NRMRL/wswrd/epanet.html

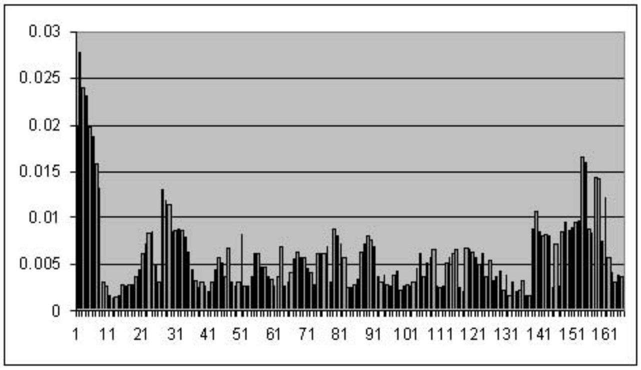

Figure 4: The distribution of 1st hidden variable; Note that sensor 1 to 7 have significantly higher mass because they are on the main pipe of water distribution.

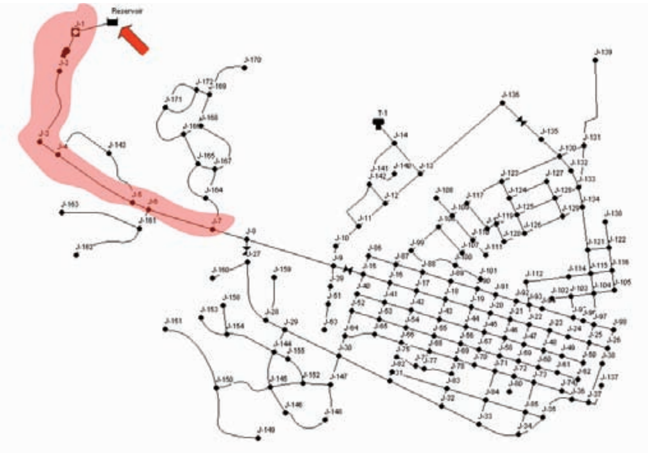

Figure 5: Water distribution network: Sensor 1-7 are highlighted since they are close to reservoir (red arrow) and on the main pipe.

bution (see Figure 9) concentrates on the sensor 37-44 (highlighted in Figure 6) since they receive more sunshine and close to window.

Comparison with static mixture modeling: To show the effectiveness of modeling mixtures dynamically, and modeling the dependency between mixtures of hidden variables, we compare our streaming modeling based on time series to the non-time dependency latent variable modeling techniques, e.g., the LDA model. The sums of reconstruction errors over time are compared and significant less ($p < 1e - 20$ under t-test) error rate were achieved with the Dynamic Mixture Model from all of 1 to 3 iterations, as shown in Figure 10.

5 Conclusion and Future Work

In this paper, we have presented Dynamic Mixture Models (DMMs) for online pattern discovery in multiple time series, and shown interesting results on two data sets. The generative process of DMMs is similar to traditional, static probabilistic mixture models such as Latent Dirichlet Allocation, but in those models, the order of the data is ignored, thus all data records are assumed to be fully exchangeable. The DMMs, on the contrary, take into consideration the temporal information (e.g., order) implied in the data . Also, compared to the state-of-the-art SVD-based methods for data streams, DMMs naturally give positive values of hidden variables, so the results from DMMs are much more intuitive and interpretable. We believe that probabilistic mixture modeling is a promising direction for streaming data, especially Dynamic Mixture Models have been shown to be a very effective model for multiple time series application.

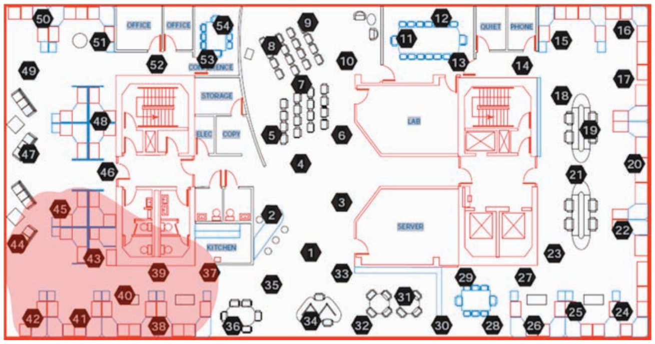

Figure 6: Sensor map, the highlighted region receives more sunshine.

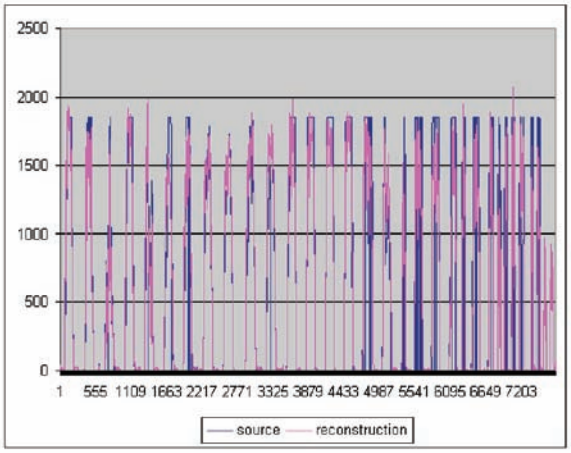

Figure 7: Light reconstruction; reconstruction based on 4 hidden variables are very close to the original value.

For future work, we plan to apply DMMs to larger datasets and improve both of its performance and efficiency on online streaming data. Modeling the dependencies using other distributions other than Dirichlet distribution will also be interesting.

Acknowledgments

This work was supported in part by the Center for Intelligent Information Retrieval, in part by the National Science Foundation under Grants No. IIS-0209107, IIS-0205224, INT-0318547, SENSOR-0329549, IIS-0326322, in part by the

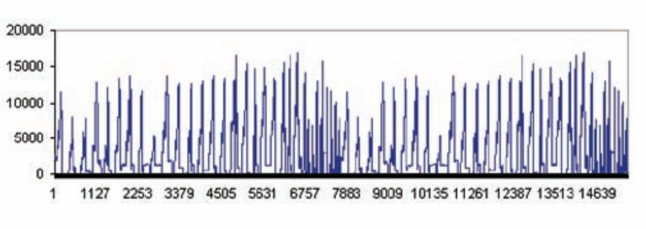

Figure 8: 1st hidden variable captures daily periodicity

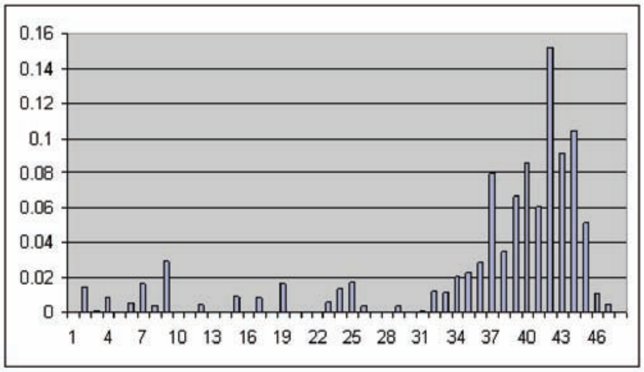

Figure 9: The distribution of 1st hidden variable, the mass focus on the sensor 37-44 due to the vicinity to window.

Pennsylvania Infrastructure Technology Alliance (PITA), a partnership of Carnegie Mellon, Lehigh University and the Commonwealth of Pennsylvania's Department of Community and Economic Development (DCED), and in part by the Defense Advanced Research Projects Agency (DARPA) under contract number HR0011-06-C-0023. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect those of the sponsors.

References

[Blei and Lafferty, 2006] David Blei and John Lafferty. Dynamic topic models. In *Proceedings of the 23rd International Conference on Machine Learning*, pages 113–120, 2006.

[Blei *et al.*, 2003] David Blei, Andrew Ng, and Micheal Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[Deshpande *et al.*, 2004] Amol Deshpande, Carlos Guestrin, Samuel Madden, Joseph M. Hellerstein, and Wei Hong. Model-driven data acquisition in sensor networks. In *Proceedings of the 30th International Conference on Very Large Data Bases*, pages 588–599, 2004.

[Geman and Geman, 1994] Stuart Geman and Donald Geman. Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:721–741, 1994.

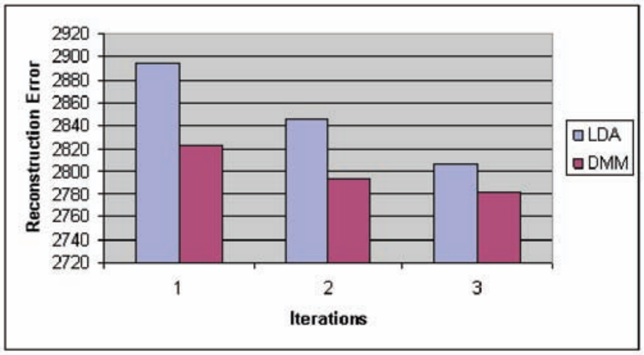

Figure 10: Comparison of reconstruction error by dynamic mixture model and static mixture model (LDA).

[Girolami and Kaban, 2005] Mark Girolami and Ata Kaban. Sequential activity profiling: latent Dirichlet allocation of Markov chains. *Data Mining and Knowledge Discovery*, 10:175–196, 2005.

[Griffiths and Steyvers, 2004] Tom Griffiths and Mark Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(suppl. 1):5228–5235, 2004.

[Hoffman, 1999] Thomas Hoffman. Probabilistic latent semantic indexing. In *Proceedings of the 22nd Annual ACM Conference on Research and Development in Information Retrieval*, pages 50–57, 1999.

[Minka, 2003] Thomas P. Minka. Estimating a Dirichlet distribution, 2003. http://research.microsoft.com/∼minka /papers/dirichlet.

[Papadimitriou *et al.*, 2005] Spiros Papadimitriou, Jimeng Sun, and Christos Faloutsos. Streaming pattern discovery in multiple time-series. In *Proceedings of the 31st International Conference on Very Large Data Bases*, pages 697–708, 2005.

[Teh *et al.*, 2004] Yee Whye Teh, Michael I. Jordan, Matthew J. Beal, and David M. Blei. Hierarchical Dirichlet processes. Technical Report 653, Department of Statistics, University of California at Berkeley, 2004.

[Wang and McCallum, 2006] Xuerui Wang and Andrew McCallum. Topics over time: A non-Markov continuous-time model of topical trends. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 424–433, 2006.

[Wang *et al.*, 2006] Xuerui Wang, Natasha Mohanty, and Andrew McCallum. Group and topic discovery from relations and their attributes. In *Advances in Neural Information Processing Systems 18*, pages 1449–1456, 2006.

[Wei and Croft, 2006] Xing Wei and W. Bruce Croft. LDA-based document models for ad-hoc retrieval. In *Proceedings of the 29th Annual ACM Conference on Research and Development in Information Retrieval*, pages 178–185, 2006.