# Detect and Track Latent Factors with Online Nonnegative Matrix Factorization

**Bin Cao[1], Dou Shen[2], Jian-Tao Sun[3], Xuanhui Wang[4], Qiang Yang[2] and Zheng Chen[3]**
[1]Peking university, Beijing, China
[2]Hong Kong University of Science and Technology, Hong Kong
[3]Microsoft Research Asia, 49 Zhichun Road, Beijing, China
[4]University of Illinois at Urbana, Champaign Urbana, USA
caobin@pku.edu.cn, {dshen, qyang}@cse.ust.hk
{jtsun, zhengc}@microsoft.com, xwang20@uiuc.edu

## Abstract

Detecting and tracking latent factors from temporal data is an important task. Most existing algorithms for latent topic detection such as Nonnegative Matrix Factorization (NMF) have been designed for *static* data. These algorithms are unable to capture the dynamic nature of temporally changing data streams. In this paper, we put forward an *online* NMF (ONMF) algorithm to detect latent factors and track their evolution while the data evolve. By leveraging the already detected latent factors and the newly arriving data, the latent factors are automatically and incrementally updated to reflect the change of factors. Furthermore, by imposing orthogonality on the detected latent factors, we can not only guarantee the unique solution of NMF but also alleviate the partial-data problem, which may cause NMF to fail when the data are scarce or the distribution is incomplete. Experiments on both synthesized data and real data validate the efficiency and effectiveness of our ONMF algorithm.

## 1 Introduction

Discovering latent factors (or topics) from an evolving data collection is an important topic in many applications ranging from text mining [Xu and Gong, 2004] to image processing [Guillamet *et al.*, 2003]. Many existing methods, including Nonnegative Matrix Factorization (NMF) [Lee and Seung, 1999], Latent Semantic Indexing (LSI) [Deerwester *et al.*, 1990] and Probabilistic Latent Semantic Indexing (PLSI) [Hofmann, 1999] are widely used to find latent factors from a *static* data collection. However, in reality, we need to consider the dynamic nature of data when they arrive over time in the form of a stream bearing the meaningful time stamps. For example, in the news mining problem domain, news articles on a certain event appear one after another, reflecting the development of an event. By considering the time information of the event, we may discover the evolution pattern of the latent factors, which reflect the event's appearance, development, fading away, and termination. Such evolution patterns are beneficial for the understanding of the event as a whole. In this paper, we will focus on the detection and tracking of latent topics that characterize the temporal data.

A key challenge of detecting and tracking the latent topics is that the data may contain various topics and the topics may evolve with time. Take the reports about Asia tsunami disaster as an example. At the beginning, one major topic in the reports is about the "financial aids", but finally this topic evolves to "debt" and "reconstruct" when the tsunami disaster is ending. Thus this task requires that the algorithm adapts itself to the topic evolvement quickly and accurately. However, most existing approaches mentioned before cannot be directly used on this task because they are aimed at handling static data. Techniques for data-stream classification, such as [Domingos and Hulten, 2000], are not designed for handling text data, while techniques for topic detection in data streams such as [Mei and Zhai, 2005] are not incremental in nature.

In this paper, we tackle these challenges by improving the basic NMF method along two directions. In previous works, the basic NMF method has been proven to be an effective method for discovering latent factors from co-occurrence data by seeking the nonnegative factors [Lee and Seung, 1999]. However, the basic NMF and its existing variations assume the latent factors and data are static, which prohibits them from reflecting the dynamic nature of data streams. In order to apply these NMF-based methods on data streams, where the data continuously arrive in a sequential manner, we have to re-calculate the latent factors from scratch every time new data come. This procedure is clearly time-consuming. Furthermore, the factors discovered at different times are independent of each other, which cannot be made to reflect the evolution of the factors. Thus, our first direction is aimed at improving the basic NMF method by developing an online version, known as ONMF, which can automatically update the latent factors by combining the old factors with the newly arrived data. At the same time, ONMF can discover the connections between the old and new factors, so that we can track the evolutionary patterns of latent factors naturally. A second direction in our research is that when data are incomplete, the latent factors found by NMF may be incorrect. To prevent this so-called partial-data problem to occur in ONMF, we impose a set of *orthogonal constraints* on all the latent factors and design an orthogonal NMF algorithm. The orthogonal NMF guarantees the uniqueness of NMF decomposition, a good property in tracking the latent factors. Experiments on both synthesized data and real data help validate the efficiency and effectiveness of our proposed approach in ONMF.

## 2 Related Work

Our work is related to temporal data analysis and NMF. A major direction in temporal data analysis is Topic Detection and Tracking (TDT) [Allan *et al.*, 1998]. TDT aims at discovering and threading together topically related material in streams of data. In this problem, a "topic" is actually a specific event or activity, described by a series of news stories. In our work, we assume that a news story/document may contain *multiple* topics and these topics may evolve over time. [Wang *et al.*, 2003] discussed the classification problem on the time series data and [Edmond H.C. Wu, 2005; Aggarwal *et al.*, 2003] discussed the clustering problem. These works are all on the "document" level rather than "topic" level. In [Mei and Zhai, 2005], the authors conducted research with an aim similar to ours and proposed a *Temporal Text Mining* framework. They split a document collection into sub-collections according to time stamps and then extracted topics from each sub-collection independently using a simple probabilistic mixture model. They then judged the connections among the discovered topics from different sub-collections by KL-divergence. Different from their work, we propose an online NMF approach to extract the topics and exploit the relations among topics in a *unified framework*.

NMF has attracted much attention in the past years [Berry *et al.*, 2006]. Most of the previous works focused on designing factorization algorithms for NMF [Lee and Seung, 2000] and imposing certain constraints to improve NMF's performance [Hoyer, 2004; J. Piper and Giffin, 2004; Li *et al.*, ]. There is also some work on accelerating the algorithms' convergence rate [Wild *et al.*, 2004]. To our best knowledge, this paper is the first attempt to extend NMF to an online setting for exploiting temporal data.

## 3 Nonnegative Matrix Factorization (NMF)

NMF seeks a lower rank decomposition of a nonnegative matrix [Berry *et al.*, 2006]. It is formalized in Equation (1):

$$V \approx WH \tag{1}$$

where $V$, $W$ and $H$ are nonnegative matrices. $V$ is a $m \times n$ matrix, in which each row represents a data sample and each column corresponds to an attribute. $H$ is a $k \times n$ matrix with each row representing a latent factor. $W$ is an $m \times k$ matrix, reflecting the association weights between the data samples and the factors. To simplify our following derivations, we use equality sign "$=$" and the approximately equal sign ( "$\approx$" ) interchangeably.

The NMF problem is solved by minimizing the distance between the original matrix and the reconstructed one, as shown in Equation (2):

$$\min ||V - WH|| \tag{2}$$

where $||\cdot||$ is a norm operator.

Often, the solution to this problem is not unique. If $V = WH$, we can find another solution $(WP)(P^{-1}H)$ so long as $WP$ and $P^{-1}H$ are nonnegative matrices.

## 4 Online Nonnegative Matrix Factorization

The conventional NMF assumes that the input data and the latent factors are static. Clearly, this assumption does not hold for temporally changing data. A straightforward way to apply NMF on temporal data is to feed NMF with the global up-to-date data matrix whenever new data come. However, the approach is not efficient since we need to work on a larger and larger data matrix without leveraging the previous factorization results. Another method is to split the data to sub-collections according to different time spans and apply NMF on each sub-collection independently. But this approach cannot detect the relations between the factors from different time. To cater for temporal data, we put forward our proposed online NMF, or ONMF, approach.

### 4.1 ONMF Problem Formulation

In order to formulate the online NMF problem, we consider matrix factorization at two neighboring time spans $t$ and $t+1$. Assume that at time $t$, we have a $m \times n$ data matrix $V$ where each row represents a data sample; $V$ is factorized by

$$V = WH$$

At time $t+1$, assume that there are $p$ new data samples that are represented by a $p \times n$ nonnegative matrix $U$. Hence the whole data matrix becomes $\widetilde{V} = \begin{pmatrix} V \\ U \end{pmatrix}$. Now the online NMF problem is how to integrate $W$ and $H$ into $\widetilde{W}$ and $\widetilde{H}$ so that $\widetilde{V} = \widetilde{W}\widetilde{H}$.

### 4.2 Our Solution to ONMF Problem

The following theorem makes it possible to design an online version of NMF.

**Theorem 1.** ***(Full-Rank Decomposition Theorem)***
*If $V = WH$ and $V = W'H'$ are both full rank decompositions, then there exists one invertible matrix $P$ satisfying $W = W'P$ and $H = P^{-1}H'$.*

*Proof.* With the condition $WH = W'H'$, by multiplying $H^T$ in both sides we have $WHH^T = W'H'H^T$. From full rank condition we get $W = W'H'(HH^T)^{-1} = W'P$, where $P = H'(HH^T)^{-1}$. As the same we can get $H = (W^TW)^{-1}W^TW'H' = QH'$. It is easy to validate $PQ = QP = I$ where $I$ is the identity matrix. Therefore, $Q = P^{-1}$. □

Consider the factorization problem.

$$\widetilde{V} = \begin{pmatrix} V \\ U \end{pmatrix} = \widetilde{W}\widetilde{H} = \begin{pmatrix} \widetilde{W_1} \\ \widetilde{W_2} \end{pmatrix} \widetilde{H} \tag{3}$$

where $\widetilde{W_1}$, $\widetilde{W_2}$ are blocks of $\widetilde{W}$ corresponding to $V$ and $U$ respectively. Therefore, we have $V = \widetilde{W_1}\widetilde{H}$. Since we already have another form of decomposition of $V$ with $V = WH$, according to Theorem (1), we can build the relationship between the two decomposition forms by $\widetilde{W_1} = WP$ and $\widetilde{H} = P^{-1}H$ where $P$ is an invertible matrix. Thus, the original factorization problem is converted to

$$U = \widetilde{W_2}\widetilde{H} \quad s.t.\ \widetilde{H} = PH \tag{4}$$

$P$ reflects the relations between the new factor matrix $\widetilde{H}$ and the old factor matrix $H$. All the decompositions satisfy the nonnegative constraint. We return to this discussion in Section 4.3.

In order to find a solution to Equation (4) , we consider the factorization of the new data matrix by replacing $V$ by $H$

$$\begin{pmatrix} H \\ U \end{pmatrix} = W^* H^* = \begin{pmatrix} W_1^* \\ W_2^* \end{pmatrix} H^* \tag{5}$$

By solving this problem we obtain $H = W_1^* H^*, U = W_2^* H^*$. $H = W_1^* H^*$ implies that $H^* = W_1^{*-1} H$ if $W_1^*$ is invertible. Now we get the solution to Equation (4) by setting $\widetilde{H} = H^*$, $P = W_1^{*-1}$, $\widetilde{W_2} = W_2^*$. Based on the previous factorization result $V = WH = WW_1^* H^*$, we have

$$\widetilde{V} = \begin{pmatrix} WW_1^* \\ W_2^* \end{pmatrix} \widetilde{H} = \widetilde{W}\widetilde{H}$$

Here we summarize the factor updating rules:

$$\widetilde{W} = \begin{pmatrix} WW_1^* \\ W_2^* \end{pmatrix} \tag{6a}$$

$$\widetilde{H} = W_1^{*-1} H \tag{6b}$$

Since the solution to Equation (5) is solved by minimizing a target function which is not convex, our current solution to Equation (3) is an approximation rather than an exact solution. However, the following analysis shows the approximate solution is reasonable. Furthermore, the empirical results on two datasets validate the reasonability of this approach.

### 4.3 Discussions of ONMF

From the algebra point of view, the task of NMF is to find a set of nonnegative bases to represent the input data by a linear combination. When new data arrive, the bases need to be updated to represent the new data. Since the old bases can be used to represent the old data, we can update the bases using the previous bases and the new data instead of using all the data. This is the philosophy behind our ONMF approach. In order to adjust the contributions of the old factors, we can modify our current ONMF by introducing a weighting schema. That is, we can use $\Lambda H$ to replace $H$ in Equation (5). $\Lambda$ is a nonnegative diagonal matrix with $\Lambda_{ii}$ representing the weight of factor $h_i$. Then the relation between the old factors and the new factors is $H^* = W_1^{-1} \Lambda H$ and the update rules become:

$$\widetilde{W} = \begin{pmatrix} W\Lambda^{-1}W_1^* \\ W_2^* \end{pmatrix}, \ \widetilde{H} = W_1^{*-1} \Lambda H \tag{7}$$

Now we show how our method deals with the temporal nature of data streams. As shown in Section 4.2, $\widetilde{H} = PH$ represents the relation between the old latent factors and the new factors through a linear transformation. In some real applications, it is possible that the relations between $H$ and $\widetilde{H}$ are not linear. Just as a nonlinear smooth function can be approximated by a linear function within a small region, the linear relation can be a good approximation in a short time span while the latent factors are changing smoothly. This claim is verified by our experiment on a simulated image dataset, which is discussed in Section 6.1.

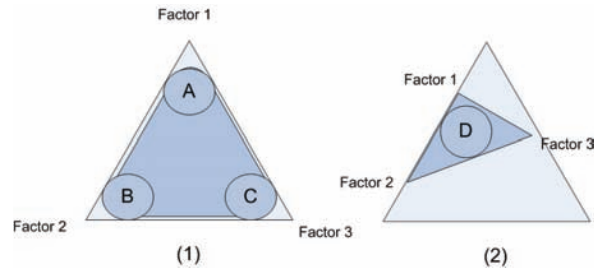


Figure 1: An Example of Partial-Data Problem

## 5 Orthogonality Constraint on ONMF

As discussed in [Donoho and Stodden, 2004] and verified in our experiments, when the data are scarce or incomplete in distribution, NMF may find the latent factors correctly. We refer to this problem as the *partial-data* problem. Figure 1 shows an illustration of this problem. In this figure, assume that we have three hidden factors, each being represented by a corner of a triangle in Figure 1. In Figure 1(1), we have enough observed data which are distributed in regions represented by A, B, C, respectively. In Figure 1(2), the observed data are only distributed within region D. As a result, the factors discovered from Figure 1(1) are correct. However, since the possible factors that generate the data in Figure 1(2) are not unique, the discovered factors are also wrong. This partial-data problem can be a serious problem in temporal data as the data may arrive in an irregular fashion.

The partial-data problem is inherently related to the unique solution problem of NMF decomposition. In [Plumbley, 2002], the authors studied the unique solution of NMF under permutation. In [Donoho and Stodden, 2004], the authors proposed the complete factorial sampling requirement for correct factorization from a geometric point of view. However, none of the above work answers how to solve the partial-data problem. To solve the partial-data problem, below we first give Theorem 2 to clarify the condition for unique NMF decomposition from an algebra point of view. Then we introduce the orthogonal constraints on ONMF to tackle the partial-data problem. The proofs of theorems in this section are not provided for space reasons.

**Theorem 2.** *Suppose that we are given a nonnegative factorization $V = WH$, where $W$ and $H$ satisfy $W = P_1 \begin{pmatrix} \Delta_1 \\ W_1 \end{pmatrix}$, $H = (\Delta_2, H_1) P_2$, and where $P_1$ and $P_2$ are permutation matrices, while $\Delta_1$ and $\Delta_2$ are diagonal matrices. The factorization is unique under permutation (apart from a scaling factor).*

Intuitively, Theorem (2) requires the latent factors to have distinct features from each other and the data distribution should be complete. In order to make the solution unique when the data are incomplete, more strict requirements are needed for the factors.

**Theorem 3.** *If we restrict $H$ to satisfy $h_i \cdot h_j = 0$ for $i \neq j$ ($h_i, h_j$ are the $i^{th}$ and $j^{th}$ rows of $H$), then the nonnegative factorization $V = WH$ is unique under permutation (apart from a scaling factor).*

Theorem 3 requires the factors are orthogonal, thus the decomposition problem is converted to a minimization problem:

$$\begin{aligned} &\min J = \frac{1}{2}||V - WH||_F^2 \\ &s.t. W \geq 0, H \geq 0, \\ &\quad h_i \cdot h_j = 0,\ i \neq j \end{aligned} \tag{8}$$

where $||X||_F^2 = \sum_{ij} X_{ij}^2$. $(\cdot)$ is the inner product. By introducing a regularizer for the orthogonality constraint, the minimization problem is further converted to the following problem:

$$J = \frac{1}{2}||V - WH||_F^2 + \alpha \Gamma H H^T \tag{9}$$

where $\Gamma$ is a symmetry nonnegative matrix with diagonal elements equal to zero and other elements greater than zero. $\alpha$ is a positive number. According to the Karush-Kuhn-Tucker(KKT) conditions [Xu and Gong, 2004], we can obtain the solution to Equation (8) by the following the iterative formulas:

$$\begin{aligned} w_{ij} &\leftarrow w_{ij} \frac{(VH^T)_{ij}}{(WHH^T)_{ij}} \\ h_{ij} &\leftarrow h_{ij} \frac{(W^T V)_{ij}}{(W^T W H + \alpha \Gamma H)_{ij}} \end{aligned} \tag{10}$$

The proof of this algorithm' convergence is omitted here, but we note that during the iterations, we need to let $\alpha$ increase from a small value to infinity in order to solve Equation (8). Detailed study of the change of $\alpha$ is left to our future work.

Now we can summarize the procedure of applying ONMF on data streams in Algorithm 1:

**Algorithm 1** : ONMF

**Timestep 0:** Initialization, using current data $V$ to calculate $W$ and $H$ by orthogonal NMF(10);
**Timestep t:**
**Substep 1:** Use new data $U$ and $H$ to calculate $\widetilde{W}$ and $\widetilde{H}$ by orthogonal NMF(10);
**Substep 2:** Update $W$ and $H$ by $\widetilde{W}$ and $\widetilde{H}$ by online NMF (7);
**Timestep T:** Output final $W$ and $H$.

Because the algorithm of orthogonal NMF (10) is iterative, initial parameter values must chosen appropriately. We can set these values randomly, but in the context of online classification tasks, a natural choice at time step $t$ is the result get from time step $t-1$.

## 6 Experiments

We conduct experiments using three different datasets to verify the effectiveness of our ONMF approach.

### 6.1 Experiments on Temporal Image Data

The first dataset consists of a series of simulated image data. Figure 2 shows the factors used for data generation. Each factor corresponds to a picture of $10 \times 10$ pixels with a horizontal or vertical bar. The intensity of any pixel is between 0 and 1. Each simulated picture is generated by a linear combination of these factors. In order to introduce evolving latent factors during the data generation process, the two factors on the left-most side in Figure 3 are made to change over time, while the other factors are kept static. The short horizontal bar moves from left to right and the short vertical bar moves from bottom to top step by step. In each step, we generated 1000 data. A subset of the simulated data are shown in Figure 4. Our task is to detect the latent factors and track their evolution from the simulated data.

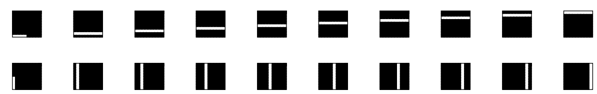

Figure 2: Factors Used for Data Generation

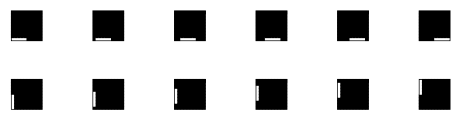

Figure 3: Two Evolving Factors Used for Data Generation

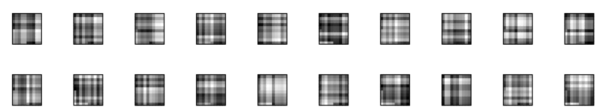

Figure 4: Simulated Data Generated from Real Factors

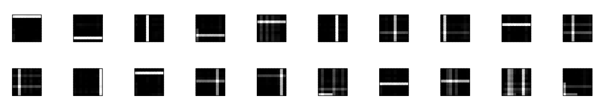

Figure 5: Factors Reconstructed by NMF

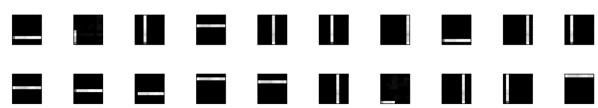

Figure 6: Factors Reconstructed by Orthogonal NMF

We fist investigate whether the orthogonal NMF can help handle the partial-data problem. Figure 5 shows the factors learned by NMF and figure 6 shows the factors learned by orthogonal NMF on 100 randomly sampled simulated data. We can see that the orthogonal NMF correctly detects the real factors while NMF only learns the mixtures of them. In this experiment, we let $\alpha = 0.1 \times 1.01^n$ during the iterations, where $n$ is the iteration number. Figure 7 shows the KL distance between the reconstructed factors and the real factors during the iterations. Clearly, compared with NMF, the orthogonal NMF can find better latent factors, which validates

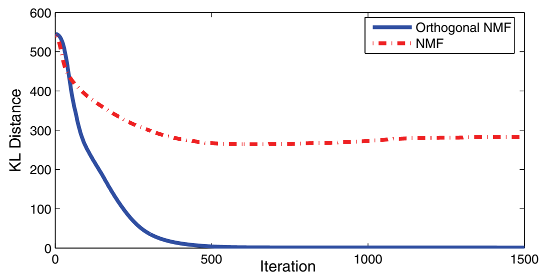


Figure 7: Reconstructed Factors' Error

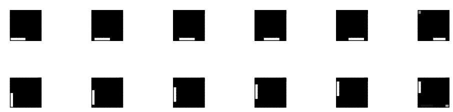

Figure 8: Two Moving Factors Reconstructed by ONMF

the effectiveness of orthogonal constraint on alleviating the partial-data problem.

Figure 8 shows the evolution of the latent factors discovered by our ONMF. Although all the factors, including the static ones, have been detected, we only show the two dynamic factors for brevity. From this figure, we can see that ONMF successfully detects and tracks the two evolving factors. As shown in Figure 3, the relations between the old factors and the new factors does not follow a linear transformation. But our ONMF can approximate the nonlinear relations and track the latent factors, as the two evolving factors change smoothly. This fact validates our conclusions in Section 4.3.

## 6.2 Experiments on 20NG Dataset

The second experiment is carried out on the 20NG dataset[1], consisting of approximately 20,000 newsgroup documents which are evenly distributed across 20 categories. Since the original 20NG did not contain time information, we manually construct 3 threads for experiments, illustrated in Figure 9. In each thread, 1000 documents are ordered according to their category sequence. The first thread contains documents about autos (denoted by "rec.*") . In the second thread ("comp.*"), the first 500 documents are about 'comp.ibm.hardware' and the remaining documents are about 'comp.mac.hardware'. In the third thread ("talk.*"), the first 300 documents are about 'talk.politics.mideast', while the middle 400 documents are about 'talk.politics.misc', and the last 300 documents are about 'talk.religon.misc'.

Documents in different threads are then mixed together. Our ONMF updates its latent factors when 300 new documents arrive. Figure 10 shows the change in the similarity between the topics at time $t$ and those at time $t-1$. From this figure, we can find both the topic evolvement trend and the time spans with severe topic change detected by our ONMF algorithm are consistent with the real data.

## 6.3 Experiments on Tsunami News Data

We also tested our approach on the real tsunami news data set used by [Mei and Zhai, 2005]. This dataset consists of 7,468

[1]http://people.csail.mit.edu/jrennie/20Newsgroups/

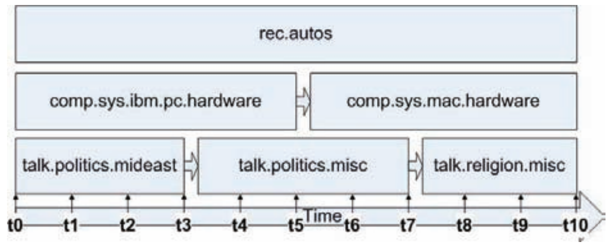


Figure 9: Three Threads of Topics of 20NG Dataset

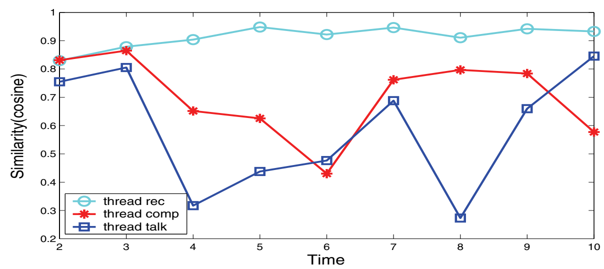


Figure 10: Changes of Similarity of Topics

news articles about the Asian Tsunami event dated from Dec. 19, 2004 to Feb. 8, 2005, sorted by their reporting times. These articles are collected from 10 sources, with the keyword query "tsunami".

To track the latent topics, we feed the news articles sequentially to our ONMF algorithm. To illustrate the topics tracked in the whole process, we select a special topic "finance" and use three representative terms "aids","reconstruct" and "debt" to show the evolution of the topic. To avoid the "report delay" problem mentioned in [Mei and Zhai, 2005], only news from CNN are used in this experiment. As shown in Figure (11), the "aids" topic has a large probability at the beginning and then decreases in the following days. "debt" and "reconstruct" have small probabilities at the very beginning but increase in the following days.

## 6.4 Time Complexity

Our ONMF is more efficient for finding new factors when new data arrive. The basic NMF algorithm [Lee and Seung, 2000] has a complexity of $O(mn)$. Our online NMF has a complexity of $O(pn)$ to update the latent factors while the basic NMF computation needs $O((m+p)n))$ to get the new factors. Figure 12 shows the comparison of the computational

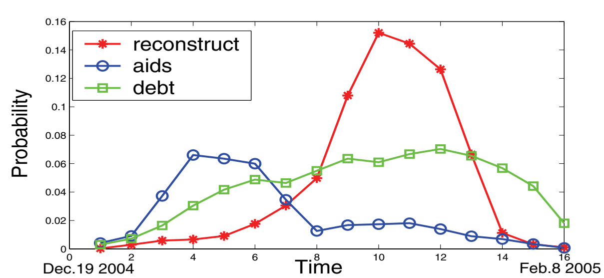


Figure 11: Evolution of Probability for Terms "reconstruct", "aids" and "debt"

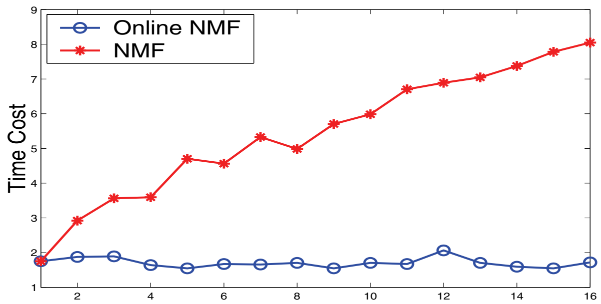

Figure 12: Time Comparison

time (in seconds) used by ONMF and NMF on the tsunami data, which validates the efficiency of ONMF.

## 7 Conclusions and Future Work

In this paper, we proposed a novel framework for detecting and tracking the latent factors in data streams by extending the Nonnegative Matrix Factorization techniques. Different from the previous approaches, our proposed methods can solve problem of topic detection and tracking in data streams efficiently within a unified framework. We also impose orthogonal constraints on NMF for tackling the partial data problem, which is an important component of our overall framework.

However, transforming the factorization problem into a minimization problem may give us only the local optimal solutions. Therefore, in the future a better algorithm needs to be designed to find the globally optimized factorization result. We also plan to test the effectiveness of our approach on more data sets.

### Acknowledgement

Dou Shen and Qiang Yang are supported by a grant from Hong Kong RGC HKUST 6187/04E.

## References

[Aggarwal *et al.*, 2003] C. Aggarwal, J. Han, J. Wang, and P. Yu. A framework for clustering evolving data streams. In *Proc. of the 29th VLDB conference*, 2003.

[Allan *et al.*, 1998] James Allan, Jaime Carbonell, George Doddington, Jonathan Yamron, and Yiming Yang. Topic detection and tracking pilot study: Final report. In *Proc. of Broadcast News Transcription and Understanding Workshop*, pages 194–218, Lansdowne, VA, 1998. NIST.

[Berry *et al.*, 2006] Michael Berry, Murray Browne, Amy Langville, Paul Pauca, and Robert Plemmons. Algorithms and applications for approximate nonnegative matrix factorization. In *Submitted to Computational Statistics and Data Analysis*, January 2006.

[Deerwester *et al.*, 1990] Scott C. Deerwester, Susan T. Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, 41(6):391–407, 1990.

[Domingos and Hulten, 2000] Pedro Domingos and Geoff Hulten. Mining high-speed data streams. In *Proc. of KDD'00*, pages 71–80, New York, NY, USA, 2000. ACM Press.

[Donoho and Stodden, 2004] David Donoho and Victoria Stodden. When does non-negative matrix factorization give a correct decomposition into parts? In *Proc. of NIPS'04*. MIT Press, Cambridge, MA, 2004.

[Edmond H.C. Wu, 2005] Philip L.H. Yu Edmond H.C. Wu. Independent component analysis for clustering multivariate time series data. *Lecture Notes in Computer Science*, 3584:474 – 482, Aug 2005.

[Guillamet *et al.*, 2003] David Guillamet, Jordi Vitrià, and Bernt Schiele. Introducing a weighted non-negative matrix factorization for image classification. *Pattern Recognition Letters*, 24(14):2447–2454, 2003.

[Hofmann, 1999] Thomas Hofmann. Probabilistic latent semantic analysis. In *Proc. of UAI'99*, Stockholm, 1999.

[Hoyer, 2004] Patrik O. Hoyer. Non-negative matrix factorization with sparseness constraints. *J. Mach. Learn. Res.*, 5:1457–1469, 2004.

[J. Piper and Giffin, 2004] R. Plemmons J. Piper, P. Pauca and M. Giffin. Object characterization from spectral data using nonnegative matrix factorization and information theory. In *AMOS Technical Conference*, Maui,HI, September 2004.

[Lee and Seung, 1999] Daniel D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788–791, 1999.

[Lee and Seung, 2000] Daniel D. Lee and H. Sebastian Seung. Algorithms for non-negative matrix factorization. In *Proc. of NIPS*, pages 556–562, 2000.

[Li *et al.*, ] S. Li, X. Hou, and H. Zhang. Learning spatially localized, parts-based representation. In *Proc. of IEEE CVPR'01*.

[Mei and Zhai, 2005] Qiaozhu Mei and ChengXiang Zhai. Discovering evolutionary theme patterns from text: an exploration of temporal text mining. In *Proc. of KDD '05*, pages 198–207, New York, NY, USA, 2005. ACM Press.

[Plumbley, 2002] M. Plumbley. Conditions for non-negative independent component analysis. *IEEE Signal Processing Letters*, 9(6):177–180, 2002.

[Wang *et al.*, 2003] Haixun Wang, Wei Fan, Philip S. Yu, and Jiawei Han. Mining concept-drifting data streams using ensemble classifiers. In *Proc. of KDD '03*, pages 226–235, New York, NY, USA, 2003. ACM Press.

[Wild *et al.*, 2004] Stefan Wild, James Curry, and Anne Dougherty. Improving non-negative matrix factorizations through structured initialization. *Pattern Recognition*, 37(11):2217–2232, 2004.

[Xu and Gong, 2004] Wei Xu and Yihong Gong. Document clustering by concept factorization. In *Proc. of SIGIR '04*, pages 202–209, New York, NY, USA, 2004. ACM Press.