Learning Evolving and Emerging Topics in Social Media: A Dynamic NMF approach with Temporal Regularization

Ankan Saha*
Department of Computer Science
University of Chicago, Chicago IL 60637
ankans@cs.uchicago.edu

Vikas Sindhwani
IBM T.J. Watson Research Center
Yorktown Heights, NY 10598
vsindhw@us.ibm.com

ABSTRACT

As massive repositories of real-time human commentary, social media platforms have arguably evolved far beyond passive facilitation of online social interactions. Rapid analysis of information content in online social media streams (news articles, blogs,tweets etc.) is the need of the hour as it allows business and government bodies to understand public opinion about products and policies. In most of these settings, data points appear as a stream of high dimensional feature vectors. Guided by real-world industrial deployment scenarios, we revisit the problem of online learning of topics from streaming social media content. On one hand, the topics need to be dynamically adapted to the statistics of incoming datapoints, and on the other hand, early detection of rising new trends is important in many applications. We propose an online nonnegative matrix factorization framework to capture the evolution and emergence of themes in unstructured text under a novel temporal regularization framework. We develop scalable optimization algorithms for our framework, propose a new set of evaluation metrics, and report promising empirical results on traditional TDT tasks as well as streaming Twitter data. Our system is able to rapidly capture emerging themes, track existing topics over time while maintaining temporal consistency and continuity in user views, and can be explicitly configured to bound the amount of information being presented to the user.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and RetrievalRetrieval Models

General Terms

Algorithms, Experimentation

Keywords

Dictionary Learning, NMF, Topic Models, Time Series Analysis

*Work done as a summer intern at IBM Research

WSDM'12, February 8–12, 2012, Seattle, Washington, USA.

1. INTRODUCTION

Over the last few years, the growth and ease of internet access accompanied by the advent of various facets of online social media *viz.* blogs, social networks and lately, twitter, has provided a vast continuous supply of dynamic diverse information content. When analyzed with appropriate statistical and computational tools, social media content can be turned into invaluable scientific and business insights. A recent large-scale study [9] of 500 million tweets generated in 2009, for example, concluded that the appearance of flu-related topics in Twitter was highly predictive of future influenza rates in the general population.

While the early years of social media platforms were focussed on developing software infrastructure for connecting people, the emphasis has only very recently begun to shift towards understanding collective public opinion by using deep, data-driven *social media analytics* [21]. One of the most basic and necessary tasks that arises in this setting is to organize streaming social media into coherent threads of discussion that can be easily analyzed and utilized to improve various services that are affected by social media. In this paper, we argue that traditional topic detection and tracking methodologies historically rooted in Information Retrieval literature need to be revisited in the context of the demands of these emerging applications.

Over regular intervals of time (which might be per day or even every few hours), any user who is continuously mining social media content, has the following natural expectations from the system: (1) to be alerted to any new emerging themes of discussion that is fast gathering steam in social media, (2) to be able to follow the evolution of existing topics that have already been identified as being of particular interest and (3) not to be overloaded with excessive bits of information that is time consuming to sift through. Several tradeoffs become naturally evident in the design of a satisfactory system. By definition, an emerging theme is one that has not been observed before and is somewhat of an anomaly in the data stream, not necessarily distinguishable from noise when first encountered. Yet, not every anomaly can be presented to the user as an "early warning" as this would lead to excessive information overload in a large-scale setting. At the same time, information presented in the past sets up expectations for what the user expects in the future, e.g., the ability to clearly see how a topic has evolved possibly in response to marketing or PR interventions. What is needed, is an ability to distinguish valid emerging topics (with steep information content) from "noise" and some method to continuously summarize essential data character-

istics in terms of a small number of human interpretable components.

Several social media monitoring tools rely on communicating individual keywords whose usage has rapidly increased in recent time, as proxies for emerging topics. Such a methodology has obvious limitations in characterizing the separate strands of conversations that may have simulatenously Joemerged in the data stream. In this paper, we describe a system for online analysis of streaming text using more rigorous machine learning and optimization methodologies in the form of powerful topic modeling and non negative matrix factorization techniques. Unlike previously proposed IR techniques, our approach incorporates special algorithmic constructs to attempt to detect emerging topics early, maintains temporal continuity while evolving existing topics in response to the statistics of incoming data, and allows the amount of information being presented to the user to be explictly configured. While we demonstrate these ideas in the context of online topic modeling, our methods apply much more broadly to early detection problems in more general signal separation and decomposition settings. A preliminary version of this work appeared in the NIPS 2010 workshop on Social Computing.

2. RELATED WORK AND OVERVIEW

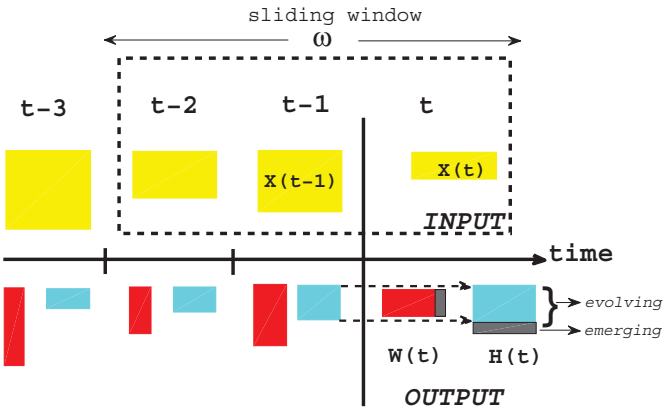
Over the last decade, different methods have been used for topic modeling and detection from a corpus of documents available as a batch or an online stream. Early impetus in this research was provided by DARPA sponsorship of Topic Detection and Tracking evaluations which led to the design of several TDT engines [26, 1]. One of the best performing engines, GAC-INCR [26], uses a clustering algorithm (GAC) to cluster incoming new data in the first phase, and then, based on a similarity/novelty threshold, either merges each of these clusters with those discovered in the past or treat them as a new cluster/topic to be tracked going forward.

Probabilistic latent semantic indexing (pLSI) [16] and Latent Dirichlet Allocation (LDA) [5] are probabilistic methods that have found remarkable success in building topic models of text. Both of them characterize topics as a multinomial distribution over a vocabulary of words rather than clusters of documents. The topics are treated as latent variables and the joint probability of the terms and documents is represented as the mixture of conditional probabilities over the latent topics which are typically inferred by maximum likelihood or Bayesian procedures that involve either variational inference or Gibbs sampling techniques. The notion of a "topic" is then communicated to a user via keywords that have highest mass in these learnt distributions.The two models are essentially equivalent [12], the key difference being that while the PLSI approach may be viewed as maximum likelihood estimation of model parameters, LDA applies a Dirichlet prior on them. Variants of pLSI and LDA have been proposed for online and dynamic topic modeling (see [6, 13, 15, 4, 2] and references therein).

Another line of seemingly unrelated work which finds use in topic modeling is that of dictionary learning and non-probabilistic matrix factorizations[20]. Dictionary Learning is the problem of estimating a collection of basis vectors over which a given data collection can be accurately reconstructed, often with sparse encodings. It may be formulated in terms of uncovering low-rank structure in the data using matrix factorizations possibly with sparsity-inducing priors [20]. These are closely related to probabilistic topic models (pLSI, LDA) for textual datasets.

In this paper, we propose a framework for online topic detection to handle streaming non-negative data matrices with possibly growing number of components. Our methods are rooted in non-negative matrix factorizations (NMF) [18, 25] whose unregularized variants for (generalized) KL-divergence minimization can be shown to be equivalent to pLSI [10]. For squared loss, NMF finds a low-rank approximation to a data matrix $\mathbf{X}$ by minimizing the Frobenius norm of $\|\mathbf{X} - \mathbf{WH}\|_{fro}^2$ under non-negativity and scaling constraints on the factors $\mathbf{W}$ and $\mathbf{H}$. Finding the minimum rank NMF of $\mathbf{X}$ is a non-convex problem and the general algorithm used is due to the multiplicative weight methods of [18]. It is common to add some form of l_1/l_2 regularization, generally to encourage sparse factors and prevent overfitting. If $\mathbf{X}$ is an $N \times D$ document-term matrix, then $\mathbf{W}$ is a $N \times K$ matrix of topic encodings of documents where each column corresponds to a topic and represents the contribution of the documents to the particular topic. $\mathbf{H}$ is a $K \times D$ matrix of topic-word associations, whose rows are the dictionary elements learnt by the NMF approach.

Figure 1: A snapshot of the online NMF system for tracking and capturing topics



We give a preview of our online learning framework in Figure 1. At any given timepoint, our system consumes the incoming data together with recently seen documents over a short time window. The output is an NMF that yields a new set of topics together with encodings for the most recently seen documents. These topics can be divided into two sets, which we call *evolving* and *emerging* sets. The evolving set is a smooth evolution of previously discovered topics. This evolution is constrained to prevent excessive drift or change that can negatively affect user interpretability. The emerging set comprises of a small number of topics injected into the model for the purpose of detecting emerging themes. This is done by finding the *optimal* word distributions that show rising temporal trends after correcting for spurious discontinuities. We show that constrained topic evolution and trend estimation can be posed naturally as extended matrix factorization problems that infact also have a link to margin-based

learning methods such as SVMs. We then develop scalable alternating optimization algorithms using efficient schemes to solve rank-one subproblems. Once the online model is learnt, the emerging set (whose size can be configured) is presented to the user who may choose to explicitly discard a subset from current consideration. Going forward in time, the emerging set becomes part of the evolving set, and new emerging bandwidth is introduced for the next timepoint. The use of explicit temporal regularizers for emerging topic detection in a matrix factorization framework in this manner is novel to the best of our knowledge. Prior work on online matrix factorization has not dealt with low-rank approximations with gradually increasing rank. In the next section, we describe the details of our online models.

Notation: In the sequel we abuse notation to denote $\mathbf{h}_i$ as the i^{th} row of $\mathbf{H}$ and $\mathbf{h}_{ij} = \mathbf{H}_{ij}$. Δ_D denotes the D dimensional simplex. $[K]$ refers to the set $\{1, 2 \dots K\}$ and $\mathbf{0}_d, \mathbf{1}_d$ refers to the vector of all 0's and 1's of dimension d.

3. DYNAMIC NMF FRAMEWORK

Let $\{\mathbf{X}(t) \in \mathbb{R}^{N(t) \times D(t)}, t = 1, 2 \dots t, ..\}$ denote a sequence of streaming matrices where each row of $\mathbf{X}(t)$ represents an observation whose time stamp is t. For simplicity in notation and exposition, we will assume that $D(t) = D$ for all t. In topic modeling applications over streaming documents, $\mathbf{X}(t)$ will represent the highly sparse document-term matrix observed at time t [1]. We will use the conventional vector space model [23] used in the information retrieval literature. Terms in a document are statistically weighted using standard measures Term Frequency(TF) and Inverse-Document Frequency(IDF). The (d, r)-th entry of $\mathbf{X}(t)$ corresponding to document d and term r is given by

$$\mathbf{X}(t)(d, r) = \frac{(1 + \log_2 TF(d, r)) \times \log_2 IDF(r)}{C}$$

where C normalizes the representation to unit norm l_2 norm vectors. We use $\mathbf{X}(t_1, t_2)$ to denote the document-term matrix formed by vertically concatenating $\{\mathbf{X}(t), t_1 \leq t \leq t_2\}$. Since we operate in an online framework, we introduce a short sliding time window ω over which trends are estimated at every time point.

At the current time point t, our model consumes the incoming data $\mathbf{X}(t)$ and appends to documents seen in a recent ω-window, $\mathbf{X}(t - \omega + 1, t)$. It generates a factorization $(\mathbf{W}^\star, \mathbf{H}(t))$ comprising of $K(t)$ topics (see equation (1)). Each column of $\mathbf{W}^\star$ and each row of $\mathbf{H}(t)$ corresponds to a topic. Note that $\mathbf{W}^\star$ has the same number of rows as all the documents accumulated over the ω-window ending at time point t. The last $N(t)$ rows of $\mathbf{W}^\star$(corresponding to $\mathbf{X}(t)$) represent the weight matrix $\mathbf{W}(t)$ at time t. Furthermore, since $\mathbf{H}(t)$ is the matrix of topic-term dependence at time t, we normalize each row of $\mathbf{H}(t)$ so that it resembles a probability distribution of words over the corresponding topic. This online factorization mechanism is designed to satisfy two considerations:

- The first $K(t-1)$ topics in $\mathbf{H}(t)$ must be smooth evolutions of the $K(t-1)$ topics found upto the previous time point in $\mathbf{H}(t-1)$. These topics are assumed to be gradually changing over time (with steady temporal profile). We call this the *evolving set* and introduce an evolution parameter, δ, which constrains the evolving set so that each entry of these $K(t-1)$ topics in $\mathbf{H}(t)$ resides within a box of size δ on the probability simplex around their previous values in $\mathbf{H}(t-1)$ (see equation (4)). With minor modifications, δ can also be made topic or word-specific e.g., to take topic volatility or word dominance into account.
- The second consideration is the fast detection of emerging topics. At each time point, we inject additional topic bandwidth for this purpose which constitute the $K(t) - K(t-1)$ remaining rows of $\mathbf{H}(t)$. This represents the *emerging set*.

Thus the topic variable $\mathbf{H}(t)$ can be partitioned into an evolving set of $K(t-1)$ topics, $\mathbf{H}^{ev}$, and an emerging set of K^{em} topics $\mathbf{H}^{em}$ where $K(t) = K(t-1) + K^{em}$. It should be noted that the set $\mathbf{H}^{ev}$ is an increasing set over time. Older topics that are no longer active may be removed for efficiency, but for simplicity, we do not discuss such a removal process in this paper. As new discussions take place over social forums, new topics emerge and are added to the system. Furthermore, we assume that emerging topics can be distinguished from noise based on their temporal profile. In other words, the number of documents that a true emerging topic associates with rapidly increases. This may occur due to a sudden large increase in discussions about the topic in documents over time.

The discussion above motivates the following objective function that is optimized at every time point t.

$$(\mathbf{W}^\star, \mathbf{H}(t)) = \underset{\mathbf{W}, \mathbf{H}}{\operatorname{argmin}} \|\mathbf{X}(t - \omega + 1, t) - \mathbf{W}\mathbf{H}\|_{fro}^2 + \mu\Omega(\mathbf{W}) \tag{1}$$

where Ω plays the role of a temporal emergence regularizer (described in section 4) which penalizes static temporal profiles and encourages the discovery of topics whose temporal profiles exhibit steep increase. These intuitively correspond to the emerging topics. This objective function is minimized under the following non-negativity, normalization and evolution constraints as discussed above.

$$\mathbf{W}, \mathbf{H} \geq 0 \tag{2}$$

$$\sum_{j=1}^{D} \mathbf{H}_{ij} = 1 \qquad \forall i \in [K(t-1) + K^{em}] \tag{3}$$

$$\min(\mathbf{H}_{ij}(t-1) - \delta, 0) \leq \mathbf{H}_{ij} \leq \max(\mathbf{H}_{ij}(t-1) + \delta, 1), \quad \forall i \in [K(t-1)], \forall j \in [D] \tag{4}$$

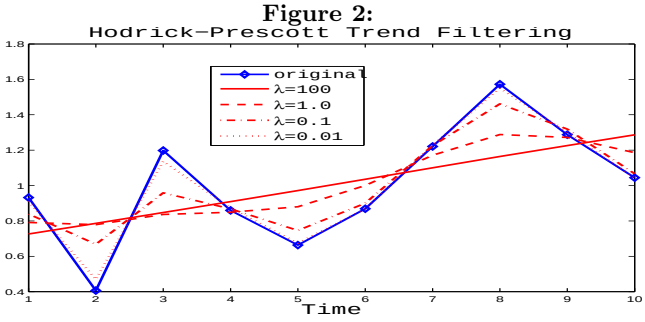
The last equation (4) enforces the smoothness condition for the evolving topics $\mathbf{H}^{ev}$ by forcing the individual topic components at time t to stay within δ of their value at the previous time step.

We then extract $\mathbf{W}(t)$ from the bottom rows of $\mathbf{W}^\star$ that correspond to $\mathbf{X}(t)$ which came in at time t. Since $\mathbf{W}(t)$ stores the weights assigned to each topic by the documents, the i^{th} document (row) in $\mathbf{X}(t)$ is tagged in the following way:

$$\pi_{system}(i) = \underset{j}{\operatorname{argmax}}\, \mathbf{W}(t)(i, j) \tag{5}$$

[1] As new documents come in and new terms are identified, we expand the vocabulary and zero-pad the previous matrices so that at the current time t, all previous and the current documents have a representation over the same vocabulary space.

Figure 2:



. Thus each document at time t is assigned to the corresponding most dominating topic by the system. Note that this gives a clustering of the documents per topic. In the next section, we define the emergence regularization operator $\Omega(\mathbf{W})$ that forms an essential component of our optimization algorithm.

4. EMERGENCE REGULARIZATION

In this section, we formulate the temporal regularization operator $\Omega(\mathbf{W})$ by chaining together trend extraction with a margin-based loss function to penalize static or decaying topics. We begin with a brief introduction to trend filtering.

4.1 Hodrick-Prescott (HP) Trend Filtering

Let $\{y_t\}_{t=1}^T$ be a univariate time-series which is composed of an unknown, slowly varying trend component $\{x_t\}_{t=1}^T$ perturbed by random noise $\{\alpha_t\}_{t=1}^T$. Trend Filtering is the task of recovering the trend component $\{x_t\}$ given the observations $\{y_t\}$. The Hodrick-Prescott filter is an approach to estimate the trend assuming that it is smooth and that the random residual is small. It is based on solving the following optimization problem:

$$\underset{\{x_t\}}{\operatorname{argmin}} \frac{1}{2}\sum_{i=1}^{T}(y_i - x_i)^2 + \lambda \sum_{t=2}^{T-1}\left((x_{t+1} - x_t) - (x_t - x_{t-1})\right)^2 \tag{6}$$

The first term tries to minimize the reconstruction error while the second term penalizes the change in the underlying time series over successive time points thus enforcing the concept that the underlying trend component is smooth.

Let us introduce the second order difference matrix $\mathbf{D} \in \mathbb{R}^{(T-2)\times T}$ such that $\mathbf{D}_{ii} = 1, \mathbf{D}_{i,i+1} = -2$ and $\mathbf{D}_{i,i+2} = 1$ for all $i \in [T-2]$. It is easy to see that the solution to the optimization problem of Equation 6 is given by:

$$\mathbf{x} = [I + 2\lambda \mathbf{D}^\top \mathbf{D}]^{-1}\mathbf{y}$$

where we use the notation $\mathbf{y} = (y_1 \dots y_T)^\top, \mathbf{x} = (x_1 \dots x_T)^\top$. In the sequel, we use F to denote $[I + 2\lambda \mathbf{D}^\top \mathbf{D}]^{-1}$, the linear smoothing operator associated with the Hodrick-Prescott Filter. Given the time series $\mathbf{y}$, the Hodrick-Prescott (HP) trend estimate simply is $\mathbf{x} = F\mathbf{y}$. Figure 2 captures the notion of Hodrick-Prescott trend filtering where a smooth reconstruction of the observed signal is demonstrated for different values of the parameter λ. In our experiments, we use $\lambda = 10$.

4.2 Loss Function for Emerging Trends

Let $\mathbf{x} = F\mathbf{y}$ be the HP trend of the time series $\mathbf{y}$. Let $\mathcal{D}$ be the forward difference operator, i.e., the only non-zero entries of $\mathcal{D}$ are: $\mathcal{D}_{i,i} = -1$ and $\mathcal{D}_{i,i+1} = 1$. If $\mathbf{z} = \mathcal{D}\mathbf{x}$, then $z_i = x_{i+1} - x_i$ reflects the discrete numerical gradient in the trend x. Given z_i, we define a margin based loss function (the ℓ_2 hinge loss), $L(z_i) = c_i \max(0, \nu - z_i)^2$. If the growth in the trend at time i is *sufficient*, i.e., greater than ν, the loss evaluates to 0. If the growth is insufficient, the loss evaluates to $c_i(\nu - z_i)^2$ where c_i is the weight of timepoint i. We observed experimentally that the best results are given by weights c_i's which typically increase with i. For a vector $\mathbf{z}$, the loss is added over the time components. In terms of the original time series $\mathbf{y}$, this loss function is,

$$L(\mathbf{y}) = \sum_{i=1}^{T-1} c_i \max(0, \nu - (\mathcal{D}F\mathbf{y})_i)^2 \tag{7}$$

Optimization Problem: As documents arrive over $t \in \{1, 2, \dots T\}$, we use $\mathbf{S}$ to denote a $T \times N$ time-document matrix, where $\mathbf{S}(i,j) = 1$ if the document j has time stamp i. Noting that each column $\mathbf{w}$ of $\mathbf{W}$, denotes the document associations for a given topic, $\mathbf{Sw}$ captures the time series of total contribution of the topic corresponding to $\mathbf{w}$, which is analogous to the temporal profile of the topic and is expected to rapidly grow for emerging topics. Finally, we concretize (1) as the following optimization problem

$$\underset{\mathbf{W},\mathbf{H}\geq\mathbf{0}}{\operatorname{argmin}} \|\mathbf{X} - \mathbf{WH}\|_{fro}^2 + \mu \sum_{\mathbf{w}_i \in \mathbf{W}^{em}} L(\mathbf{Sw}_i) \tag{8}$$

subject to constraints in equations 3 and 4. Note that the sum in the penalization term only runs over the emerging topic variables.

5. OPTIMIZATION ALGORITHMS

We approximate $\mathbf{X}$ as the sum of rank-one matrices $\mathbf{w}_i\mathbf{h}_i^\top$ and optimize cyclically over individual $\mathbf{w}_i$ and $\mathbf{h}_i$ variables while keeping all other variables fixed. This results in three specific sub-problems, each of which requires an efficient projection of a vector onto an appropriate space. Optimization of rank-one subproblems has been previously shown to be very effective for standard NMFs [14, 7] and is also reminiscent of the K-SVD approach for dictionary learning [11].

Optimization over $\mathbf{h}_i$: Firstly note that since the regularization term is independent of $\mathbf{h}_i$, it does not contribute to this optimization problem. Holding all variables except $\mathbf{h}_i$ fixed and omitting additive constants independent of $\mathbf{h}_i$, (8) can be reduced to $\operatorname{argmin}_{\mathbf{h}_i \in \mathfrak{C}} \left\|R - \mathbf{w}_i\mathbf{h}_i^\top\right\|_{fro}^2$ where

$$R = \mathbf{X} - \sum_{j \neq i} \mathbf{w}_j\mathbf{h}_j^\top \tag{9}$$

is the residual matrix independent of $\mathbf{h}_i$. Note that R is the difference of a sparse matrix and rank one matrices. While it can possibly be a dense matrix, we never need to evaluate it explicitly. In particular, our algorithm only needs to compute matrix vector products against R, namely

$$R\mathbf{h}_i = \mathbf{X}\mathbf{h}_i - \sum_{j \neq i} \mathbf{w}_j(\mathbf{h}_j^\top \mathbf{h}_i) \qquad \text{and}$$

$$R^\top \mathbf{w}_i = \mathbf{X}^\top \mathbf{w}_i - \sum_{j \neq i} \mathbf{h}_j(\mathbf{w}_j^\top \mathbf{w}_i)$$

We use sparse matrix computations to evaluate $\mathbf{X}\mathbf{h}_i$ and $\mathbf{X}^\top \mathbf{w}_i$ thus allowing us to efficiently compute the updates without evaluating R explicitly.

Algorithm 1: Online Learning Algorithm for Topic Evolution and Emergence.

Input: New data $\mathbf{X}(t) \in \mathbb{R}^{N(t)\times D}$, Old data $\mathbf{X}(t-\omega+1, t-1)$, Previous topic matrix $\mathbf{H}(t-1)$ of size $K(t-1)\times D$, Emerging Topic Bandwidth B, Hyperparameters: Evolution δ, Emergence μ, Hinge ν.
Output: $\mathbf{W}(t) \in \mathbb{R}^{N(t)\times K(t)}$, $\mathbf{H}(t) \in \mathbb{R}^{K(t)\times D}$
$\epsilon = 10^{-6}$
Set $K(t) = K(t-1) + B$
Define $\mathbf{X} = \mathbf{X}(t-\omega+1, t) \in \mathbb{R}^{N\times D}$
Initialize $\mathbf{W} = \mathbf{W}_{\text{init}}, \mathbf{H} = \mathbf{H}_{\text{init}}$ where $\mathbf{W}_{\text{init}} \in \mathbb{R}^{N\times K(t)}, \mathbf{H}_{\text{init}} \in \mathbb{R}^{K(t)\times D}$ (details in Section 5).
while not(converge) **do**
  for $i = 1, \dots K(t)$ **do**
    if $i \le K(t-1)$ **then**
      $R\mathbf{h}_i = \mathbf{X}\mathbf{h}_i - \sum_{j\neq i} \mathbf{w}_j(\mathbf{h}_j^\top \mathbf{h}_i)$
      Evolving $\mathbf{w}_i$: $\mathbf{w}_{ij} = \max\left(0, \frac{1}{\|\mathbf{h}_i\|^2}(R\mathbf{h}_i)_j\right)$.
    else
      Emerging $\mathbf{w}_i$ (use Projected Gradient):
      $\mathbf{w}_i = \text{argmin}_{\mathbf{w}\ge\mathbf{0}} \left\|\mathbf{w} - R\mathbf{h}_i/\|\mathbf{h}_i\|^2\right\|^2 + \frac{\mu}{\|\mathbf{h}_i\|^2}\sum_{j=1}^{T-1} c_j \max(0, \nu_j - \mathbf{q}_j^\top \mathbf{w}_i)^2$ (see Section 5).
    end if
  end for
  for $i = 1, \dots K(t)$ **do**
    $R^\top \mathbf{w}_i = \mathbf{X}^\top \mathbf{w}_i - \sum_{j\neq i} \mathbf{h}_j(\mathbf{w}_j^\top \mathbf{w}_i)$
    if $i \le K(t-1)$ **then**
      $l_{ij} = \max(0, \mathbf{h}_{ij}(t-1) - \delta)$,
      $u_{ij} = \min(\mathbf{h}_{ij}(t-1) + \delta, 1)$.
      Evolving $\mathbf{h}_i$ (Simplex projection with box constraints):
      $\mathbf{h}_i = \text{argmin}_{\mathbf{h}\in\mathfrak{C}_1} \left\|\mathbf{h} - R^\top \mathbf{w}_i/\|\mathbf{w}_i\|^2\right\|^2$
      where $\mathfrak{C}_1 = \{\mathbf{h} : \mathbf{h} \in \Delta_D, \quad l_{ij} \le \mathbf{h}_j \le u_{ij}\}$.
    else
      Emerging $\mathbf{h}_i$ (use Simplex Projection):
      $\mathbf{h}_i = \text{argmin}_{\mathbf{h}\in\Delta_D} \left\|\mathbf{h} - R^\top \mathbf{w}_i/\|\mathbf{w}_i\|^2\right\|^2$
    end if
  end for
  Convergence Check: Relative change in Objective value $< \epsilon$
end while
$\mathbf{H}(t) = \mathbf{H}$ and $\mathbf{W}(t)$ is last $N(t)$ rows of $\mathbf{W}$.

Simple linear algebraic operations yield that the above is equivalent to

$$\underset{\mathbf{h}_i\in\mathfrak{C}}{\text{argmin}} \left\|\mathbf{h}_i - R^\top \mathbf{w}_i/\|\mathbf{w}_i\|^2\right\|^2 \tag{10}$$

Note that the domain of dependence $\mathfrak{C}$ changes according to different constraints on $\mathbf{h}_i$ depending on whether it is an emerging or evolving topic.

Evolving $\mathbf{h}_i$: For an evolving topic, the optimization needs to be performed under the smoothness and the normalization constraints ((4) and (3) respectively). Thus the optimum $\mathbf{h}_i^\star$ is obtained by optimizing the above objective over the set $\mathfrak{C} = \{\mathbf{h}_i : \mathbf{h}_i \in \Delta_D, l_j \le \mathbf{h}_{ij} \le u_j\}$ for appropriate lower and upper bounds l_j and u_j determined by (4). This is equivalent to a projection onto a simplex with box constraints. Adapting a method due to [22], we can find the minimizer in $O(D)$ time *i.e.* linear in the number of coordinates.

Emerging $\mathbf{h}_i$: For an emerging topic, we do not have the smoothness constraints and the domain $\mathfrak{C} = \{\mathbf{h}_i : \mathbf{h}_i \in \Delta_D\}$ is just the D dimensional simplex. The optimization (10) becomes equivalent to a projection onto the simplex Δ_D. The same algorithm [22] again gives us the minimizer in linear time $O(D)$.

Evolving $\mathbf{w}_i$: When $\mathbf{w}_i \in \mathbf{W}^{ev}$, the regularization term in (8) does not contribute and the corresponding optimization problem boils down to $\mathbf{w}_i^\star = \text{argmin}_{\mathbf{w}_i\ge\mathbf{0}} \left\|R - \mathbf{w}_i\mathbf{h}_i^\top\right\|^2$. Similar to (10), simple algebraic operations yield that the above minimization is equal to the following simple projection problem,

$$\underset{\mathbf{w}_i\ge\mathbf{0}}{\text{argmin}} \left\|\mathbf{w}_i - R\mathbf{h}_i/\|\mathbf{h}_i\|^2\right\|^2 .$$

The projection set now is just the non-negative orthant, for which there is a closed form minimizer: $\mathbf{w}_{ij} = \max\left(0, \frac{1}{\|\mathbf{h}_i\|^2}(R\mathbf{h}_i)_j\right)$.

Emerging $\mathbf{w}_i$: When $\mathbf{w}_i \in \mathbf{W}^{em}$, the second term in (8) is active and the corresponding optimization problem looks like

$$\underset{\mathbf{w}_i\ge\mathbf{0}}{\text{argmin}} \left\|R - \mathbf{w}_i\mathbf{h}_i^\top\right\|^2 + \mu L(S\mathbf{w}_i)$$

Omitting the terms independent of $\mathbf{w}_i$, simple algebra yields that the above is equivalent to

$$\underset{\mathbf{w}_i\ge\mathbf{0}}{\text{argmin}} \left\|\mathbf{w}_i - R\mathbf{h}_i/\|\mathbf{h}_i\|^2\right\|^2 + \mu L(S\mathbf{w}_i)/\|\mathbf{h}_i\|^2 \tag{11}$$

Noting that we choose L to be the ℓ_2 hinge loss, (11) can be rewritten as

$$\underset{\mathbf{w}_i\ge\mathbf{0}}{\text{argmin}} \left\|\mathbf{w}_i - R\mathbf{h}_i/\|\mathbf{h}_i\|^2\right\|^2 + \frac{\mu}{\|\mathbf{h}_i\|^2}\sum_{j=1}^{T-1} c_j \max(0, \nu_j - \mathbf{q}_j^\top \mathbf{w}_i)^2$$

where $\mathbf{q}_j^\top = (\mathcal{D}F\mathbf{S})_{j,:}$, the j^{th} row of $\mathcal{D}F\mathbf{S}$ where the operators are defined in (7) and (8) respectively. Assimilating the $\mu/\|\mathbf{h}_i\|^2$ into the constant c_i, the above optimization problem can be written in the following generic form

$$\min_{\mathbf{w}\ge\mathbf{0}} \quad J(\mathbf{w}) \qquad \text{where}$$

$$J(\mathbf{w}) = \sum_i \max(0, c_i(\nu_i - \langle\mathbf{w}, \mathbf{x}_i\rangle))^2 + \frac{\lambda}{2}\|\mathbf{w} - \mathbf{w}_0\|^2 \tag{12}$$

where $\mathbf{w}_0$ refers to the term $R\mathbf{h}_i/\|\mathbf{h}_i\|^2$. Note that this is the same as the L2-SVM optimization problem with additional non-negativity constraints on $\mathbf{w}_i$ and the regularizer measuring distance from the vector $\mathbf{w}_0$ instead of the origin.

This objective is minimized using a projected gradient algorithm (due to the non-negativity constraint) [19] on the primal objective directly, as it is smooth and therefore the gradient is well defined. The update is given by

$$\mathbf{w}^{(k+1)} = \prod(\mathbf{w}^{(k)} - \eta_k \nabla J(\mathbf{w}^{(k)}))$$

where $\prod$ is the projection operator $\prod(s) = \max(s, 0)$ and

$$\nabla J(\mathbf{w}^{(k)}) = -2\sum_i \max\left[c_i\left(\nu_i - \left\langle\mathbf{w}^{(k)}, \mathbf{x}_i\right\rangle\right), 0\right]\mathbf{x}_i + \lambda(\mathbf{w}^{(k)} - \mathbf{w}_0)$$

The main trick lies in choosing the best rate η_k at the k^{th} step. Following [19], we start with $\eta_0 = 1$ and at every step hot start $\eta_k = \eta_{k-1}$. If η_k satisfies

$$J(\mathbf{w}^{(k+1)}) - J(\mathbf{w}^{(k)}) \leq \sigma \left\langle \nabla J(\mathbf{w}^{(k)}), \mathbf{w}^{(k+1)} - \mathbf{w}^{(k)} \right\rangle \quad (13)$$

we continuously increase $\eta_k \leftarrow \eta_k/\beta$ as long it satisfies (13). If η_k initially does not satisfy (13), we continuously decrease $\eta_k \leftarrow \eta_k\beta$ until the condition is satisfied. For our experiments, we choose $\beta = 0.05$ and $\sigma = 0.01$.

Our online learning algorithm can now be composed as in the table labeled Algorithm 1.

Initializations: Note that in the algorithm above, $\mathbf{H}_{\text{init}} = \begin{pmatrix} \mathbf{H}^{ev} \\ \mathbf{H}^{em} \end{pmatrix}$ where $\mathbf{H}^{ev}$, the evolving set is initialized from $\mathbf{H}(t-1)$ and $\mathbf{H}^{em}$ is initialized to random distributions.

The corresponding parts of $\mathbf{W}_{\text{init}} = \begin{pmatrix} \mathbf{W}_{11} & \mathbf{W}_{12} \\ \mathbf{W}_{21} & \mathbf{W}_{22} \end{pmatrix}$ are initialized in the following way: $\mathbf{W}_{11}$ is extracted from previous runs (previous association of old documents with existing topics), $\mathbf{W}_{12} = 0$ (old documents have weak association with emerging topics), $\mathbf{W}_{21}$ optimizes

$$\underset{\hat{\mathbf{W}} \geq 0}{\text{argmin}} \|\mathbf{X}(t) - \hat{\mathbf{W}}\mathbf{H}^{ev}\|_{fro}^2$$

In other words, existing topics are allowed to first reconstruct new documents. Finally, $\mathbf{W}_{22}$ is chosen to be the columns of $\mathbf{W}$ corresponding to emerging topics where $\mathbf{W}$ is chosen randomly and scaled appropriately with

$$\alpha = \underset{\beta > 0}{\text{argmin}} \|\mathbf{X}(t) - \beta\mathbf{W}\mathbf{H}_{init}\|_{fro}^2$$

(see [14] for simple expressions for α).

Convergence: Using a general result on convergence of Block Coordinate Descent, from [3], we can show that the limit point $(\mathbf{W}^*, \mathbf{H}(t))$ generated by algorithm 1 is a stationary point of the objective function (1). This follows from the uniqueness of the projection onto a closed convex set (simplex, or simplex with box constraints) and the strict convexity of Eq. 12.

6. PERFORMANCE EVALUATION

We conducted a comprehensive empirical evaluation of our system on both traditional topic detection and tracking datasets comprising of streaming news stories, as well as a twitter stream filtered for tweets relevant to IBM that was collected over a span of 6 weeks.

It should be noted that the goal of our experiments is to empirically understand the effectiveness of the temporal regularizers. As a result, most of our experiments try to demonstrate the role of these temporal regularizers on traditional news datasets as well as twitter streams as opposed to exhaustive comparison with existing TDT algorithms. However we still perform comparisons with a simple baseline TDT model to put our models into perspective. We begin with a discussion on appropriate evaluation metrics, some of which are new to the best of our knowledge.

6.1 Metrics and Methodology

For performance evaluation, we assume that documents in the corpus have been manually identified with a set of E events. For simplicity, we assume that each document i is tagged with a single, most dominant event that it associates with $\pi_{true}(i) \in \{1 \ldots E\}$. In the description below, we call these human labeled topics as events or "true topics" and assume them to be the ground truth.

Microaveraged F1: This measure has been commonly reported in topic detection and tracking (TDT) literature (see, e.g., [6]). Let us assume that the system generates S topics where in general $S \neq E$ i.e., the system is allowed to generate any number of topics. We first construct the $E \times S$ confusion matrix CM between events and system topics, i.e., $CM_{e,s}$ is the number of documents that were tagged e by the human and tagged s by the system. From this matrix, for each event e, we identify $top_k(e)$ – the set of top-k most frequently co-occuring system topics. We can then compute the *microaveraged* F1 measure as follows:

$$Precision_k = \frac{\sum_{e=1}^{E} |D_{true}(e) \cap D_{system}(e)|}{\sum_{e=1}^{E} |D_{system}(e)|} \quad (14)$$

$$Recall_k = \frac{\sum_{e=1}^{E} |D_{true}(e) \cap D_{system}(e)|}{\sum_{e=1}^{E} |D_{true}(e)|} \quad (15)$$

$$F1_k = \frac{Precision * Recall}{Precision + Recall} \quad (16)$$

where $D_{true}(e)$ is the set of documents human-tagged e and $D_{system}(e)$ is the set of documents system-tagged with a topic in $top_k(e)$, $|\cdot|$ denotes the cardinality and $\cap$ denotes intersection operators on sets. Note that while k in top_k is typically chosen to be 1, higher values may also be meaningful to study particularly when multiple system topics attempt to capture the same semantic event. Also note that the F1 score also implicitly measures topic continuity which is important in providing a stable, consistent association across time between events and system topics.

Miss Rate @ First Detection: This measure attempts to intuitively capture the following notion: *how much of an event has been "missed" before the system is able to communicate its existence to the user via the word-distribution of a topic.* It is a direct measure of the quality of an online topic detection model to serve as an early warning system for emerging themes. To compute this measure, we find the *first detection timepoint*: the earliest timepoint when the word-distribution of a system generated topic gets *sufficiently near* the word distribution of the event e (which is itself evolving temporally). We then compute the percentage of documents tagged with e that have streamed away prior to first detection. This fraction is called the *miss rate @ first detection*. To operationalize this measure, we define the following:

(a) *Word distribution of an event* e at time t: the normalized centroid of all documents on event e upto time t. Note that this distribution is also obtained from the solution to the optimization problem

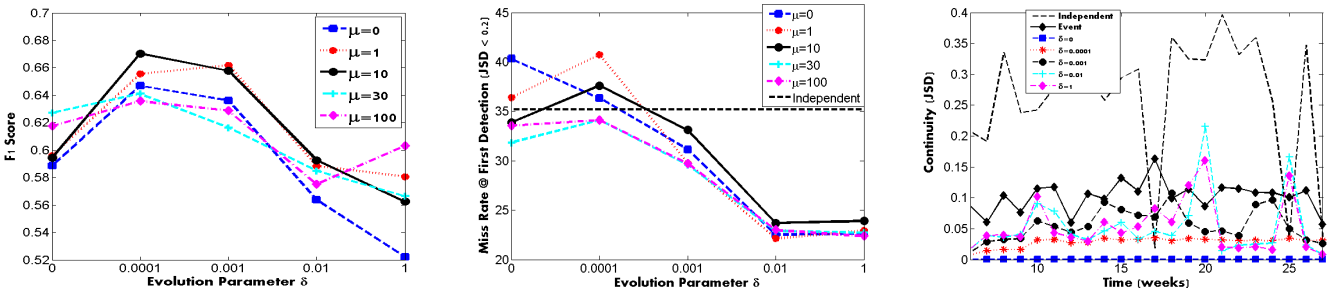
$$\mathbf{H}_{true}(t) = \underset{\mathbf{H} \geq 0, \mathbf{H}\mathbf{1}_D = \mathbf{1}_E}{\text{argmin}} \|\mathbf{X}(1:t) - \mathbf{W}_{true}\mathbf{H}\|_{fro}^2$$

where $\mathbf{W}_{true}$ is simply the matrix encoding for π_{true}, i.e., $\mathbf{W}_{true}(i,e) = 1$ *if* $\pi_{true}(i) = e$ and 0 otherwise. $\mathbf{H}_{true}(t)$ serves as a proxy for the word distributions of the true topics.

(b) *Similarity measure for Topic nearness*: Given two discrete distributions $\mathbf{p}$ and $\mathbf{q}$, we use the Jensen-Shannon Divergence (JSD) as a measure of topic proximity,

$$JSD(\mathbf{p}, \mathbf{q}) = \frac{1}{2}\left(KL(\mathbf{p}||\mathbf{m}) + KL(\mathbf{q}||\mathbf{m})\right)$$

Figure 3: Metrics for TDT2 (note that F1 score for independent models (not plotted) is 0.21).



where $\mathbf{m} = \frac{1}{2}(\mathbf{p}+\mathbf{q})$. Other measures (KL-divergence, Symmetric KL-divergence, Overlap in top words etc) are also possible. However, we chose JSD because it is always numerically well defined and bounded between 0 and 1.
(c) *First detection Time* for an event e is defined as:

$$t_{detect}(e) = \underset{t}{\text{argmin}} \left[t : \min_{s} JSD\left(h_{true}(e;t), h_{sys}(s;t)\right) < \theta \right]$$

where θ is a detection threshold and $h_{true}(e;t)$ is the word distribution for event e at time t *i.e.* the row corresponding to event e in $\mathbf{H}_{true}(t)$, and $h_{sys}(s;t)$ is a row of $\mathbf{H}(t)$, i.e., a system topic s at time t. Hence, t_{detect} simply measures the first timepoint at which a system topic comes θ-close to an event as measured by JSD. In our experiments, we either set θ to a small value (0.2) or study variation with respect to its choice. The miss rate for event e can now be defined as:

$$\frac{|d : d \in D_{true}(e), timestamp(d) < t_{detect}(e)| * 100}{|D_{true}(e)|} \quad (17)$$

We study miss rates for individual events or report an average across all events.

Topic Continuity: We adapt a direct measure of topic continuity suggested in [6]. This measure essentially reports the JSD between word-distributions in consecutive timepoints both for events (true topics) as well as for system-topics. It therefore measures temporal smoothness in topic distributions.

Emerging Bandwidth: The topics in the emerging set can a) either all become part of the evolving set going forward in time, b) can be manually selected with some of them being discarded as noise by the user or c) can be selected using criteria such as net current strength (sum of the document weights of a topic from the corresponding column of $\mathbf{W}$). In our experiments, we choose (a) and retain all topics in the emerging set.

6.2 Threshold Based TDT model

We adopt a simple baseline model (Threshold Based TDT) based on a body of work in topic detection [1, 8] which corresponds to one of the classic topic detection frameworks. We adapt these body of algorithms for comparison to our experimental setup by using analogous concepts like adhoc period and emerging topics. In particular, we learn an initial set of topics by k-means clustering of the documents over an adhoc period. New documents then stream in one time unit at a time and if their similarity to the existing topics exceed a certain threshold (**Yes/No threshold**), they are allocated to those topics. We also incorporate adaptation [8] so that if the similarity of an incoming document to a topic exceeds a certain **adaptation threshold** τ, we modify the corresponding cluster (topic) center to incorporate the effect of the new document. The threshold values are treated as tunable parameters. The remaining documents which are not similar to the existing topics are then considered as belonging to emerging topics. We perform another k-means clustering to cluster these documents into a set of emerging topics. The documents which are very far from these cluster centers are assigned to singleton topics. Similar to our temporal model, the emerging topics are absorbed into the evolving set going forward in time.

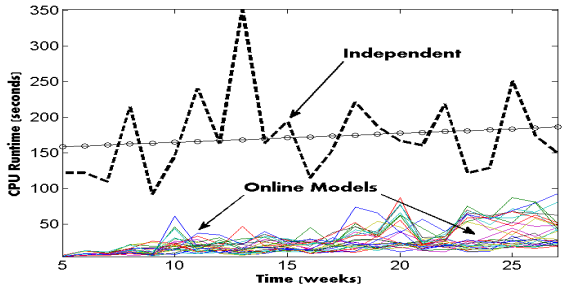
6.3 TDT2 Dataset

Our first dataset is drawn from the NIST Topic Detection and Tracking (**TDT2**) corpus[2] which consists of news stories in the first half of 1998. In our evaluation, we used a set of 9394 documents represented over 19528 terms and distributed into the top 30 TDT2 human-labeled topics over a period of 27 weeks. We choose the hinge parameter to be $\nu = 20$ and emerging bandwidth of 2 per week for this dataset. In our experiments, we use a sliding window of $\omega = 4$ weeks. The left panel of Figure 3 shows tracking performance ($F1$) as a function of the evolution parameter δ for various values of μ. When $\delta = 0$, the system freezes a topic as soon as it is detected not allowing the word distributions to change as the underlying topic drifts over time. When $\delta = 1$, the system has complete freedom in retraining topic distributions causing no single channel to remain consistently associated with an underlying topic. It can be seen that both these extremes are suboptimal. Tracking is much more effective when topic distributions are allowed to evolve under sufficient constraints in response to the statistics of incoming data. Moreover, the presence of emergence regularizer $\mu > 0$ tends to lead to higher F1-score also. In the middle panel of Figure 3 we turn to the miss rate at first detection. Higher values of μ typically reduces miss rates helping emerging topics to be detected early. As δ is increased, topics become less constrained and therefore provide additional bandwidth to drift towards emerging topics, therefore lowering the miss rate curves. However, this comes at the price of reduced tracking performance. Thus, for a fixed amount of available topic bandwidth (which also corresponds to user information overload), there is a tradeoff between tracking and early detection that can be navigated

[2]http://www.nist.gov/speech/tests/tdt/tdt98/index.htm

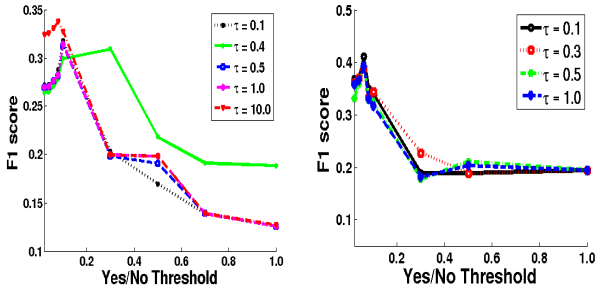
with the choice of μ and δ. The rightmost panel of Figure 3 shows topic continuity. Choosing $\delta = 0$ leads to no evolution of word distributions while $\delta = 1$ can lead to abrupt changes than can cause user confusion. For other values of δ we see that the system allows drift that is of the same scale as the drift in the true word distributions. Figure 3 also plots the metrics for a simple online methodology where a fresh NMF is learnt at every timepoint over all the data seen so far, independent of previous runs. This model is labeled *independent* in the Figure. As can be seen, independent runs lead to high miss rate (see middle panel) since small emerging topics can be swamped by the goal of reconstructing the ever expanding set of historical documents. Due to complete temporal independence, no system topic can maintain a stable association with an event leading to low F1 score (0.20, not plotted in the left panel) and high temporal discontinuity (see right panel). In Figure 4, we see that, as expected, training online models across short sliding windows takes much lesser time overall than training fresh batch models at every timepoint.

Figure 4: Training time on TDT2



We perform baseline experiments with the Threshold Based TDT model over the TDT2 dataset. Due to space constraints, we just report the tracking performance using F1 score (left panel on Figure 5) as a function of the Yes/No threshold for various values of the Adaptation threshold, τ. The F1 scores obtained by our temporal model are clearly much better than the simple baseline.

Figure 5: Metrics for TDT-Adapt: F1 score on TDT2 and Twitter



6.4 IBM Twitter Archive

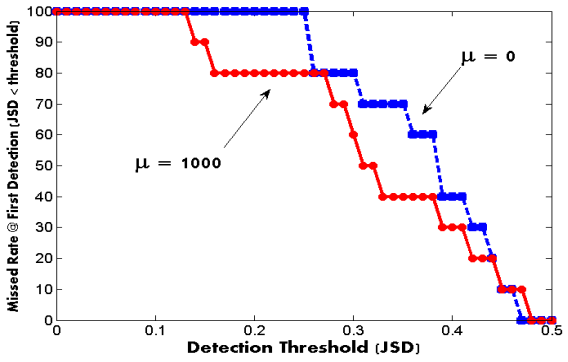
We used the Twitter Search API to collect all tweets mentioning "IBM" in the time period Dec 21, 2010 to Feb 2, 2011. Our dataset comprises of 198029 tweets spread over 43 days. The first 7 days are used to build an initial adhoc model after which the data is streamed into our models for online analysis. In order to facilitate quantitative benchmarking, we labeled approximately 23% of this data in a semi-automatic fashion. First, we learnt a batch NMF model over the entire dataset with 100 topics. From this model, we identified 10 salient events in this time period concerning IBM, listed in the table below together with an estimate of date of peak strength from the model.

Table 1: 10 events comprising roughly 23% of 198029 tweets between Dec 21, 2010 and Feb 2, 2011.

ID	Event	Size	Peak Date
1	Lotusphere 2011 Conference	5020	Jan 31, 2011
2	Watson Jeopardy Contest	16388	Jan 14, 2011
3	Graphene Transistors	874	Jan 24, 2011
4	IBM Patents	2459	Jan 10, 2011
5	IBM Virtual Desktop Offering	1889	Jan 24, 2011
6	IBM Cloud Computing Data Center	2823	Jan 26, 2011
7	IBM Centennial	2558	Jan 22, 2011
8	IBM Quarterly Earnings	6944	Jan 18, 2011
9	IBM Analytics Study	3413	Jan 14, 2011
10	IBM ARM Partnership	3168	Jan 20, 2011

Several topics in the batch model were found to cover these events and induced a partial clustering of tweets which was treated as ground truth. Note, the online model needs to make a real-time judgement of emerging topics and online tracking of evolving ones, and does not have the benefit of retrospective hindsight that the batch model does. It is important to note that our online models are trained on the entire dataset, and only the evaluation is restricted to these labeled events. In other words, these events need to each be teased apart from the entire data collection as they emerge. These 10 events are quite diverse and cover the victory of the IBM Watson supercomputer in a practice Jeopardy! round in January, IBM's quarterly earnings statement, announcements such as creation of the largest cloud computing data center in Asia and a new virtual desktop offering, new partnerships, market studies, patent leadership, centennial celebrations and super fast Graphene transistors. In our empirical study, we fixed the hinge parameter $\nu = 20$, evolution parameter $\delta = 0.001$, sliding window of one week ($\omega = 7$) and emerging topic bandwidth to be 4. regularization. Figure 6 shows that the miss rate at first detection

Figure 6: Miss Rate at First Detection as a function of Detection Threshold on IBM Twitter data



curve as a function of detection threshold (θ in Equation 17) is significantly lower when the emergence regularizer is used.

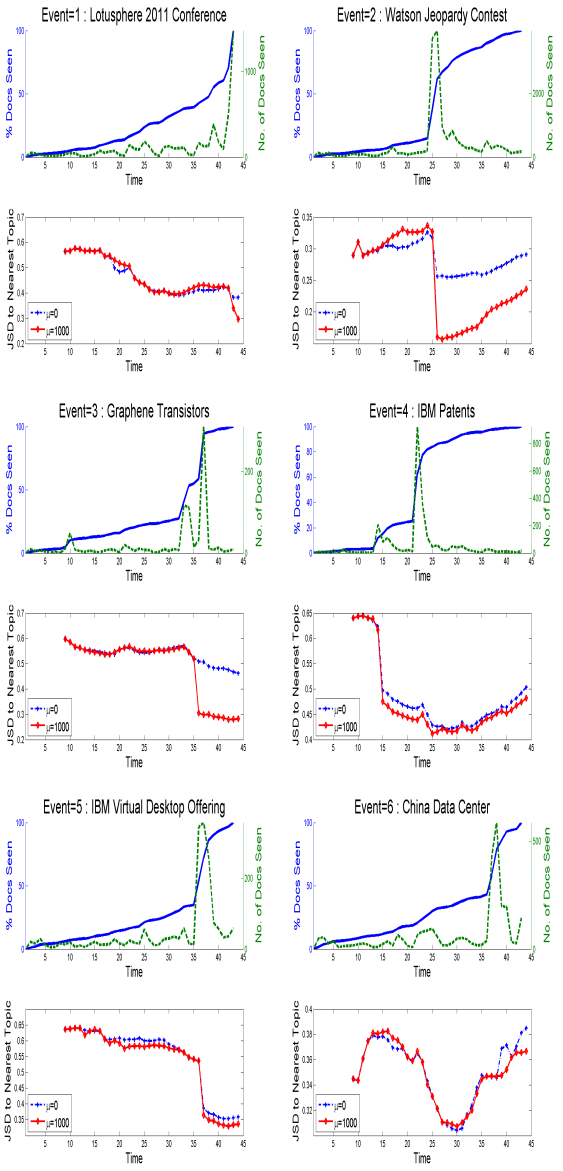
Table 2: Tracking Performance on Twitter

	top_1		top_5	
Measure	$\mu = 0$	$\mu = 1000$	$\mu = 0$	$\mu = 1000$
Precision	92.10	91.50	71.03	82.09
Recall	36.72	30.73	65.05	62.01
F1	52.51	46.00	67.91	70.65

In Figures 7, 8 we see the effect of emergence regularization: on several events, the online model with $\mu = 1000$ shows a much sharper dip (see bottom panel for each event) as compared to $\mu = 0$ in terms of JSD with respect to the true word distribution, around the time the event emerges (see top panel for each event). These results clearly show the effectiveness and potential value of emergence regularization. The training time is 3.8 minutes per day for $\mu = 0$ and 4.5 minutes per day for $\mu = 1000$ on a commodity desktop running MATLAB. In the Table 2, we report $F1$ scores at $k = 1$ and $k = 5$ respectively. We see that the models tend to have high precision and in particular, the presence of the emergence regularizer also significantly improves precision when upto 5 best matched system topics are associated with each event. Finally, in Figure 9, we show a visualization of topic temporal rivers [24] populated by keyword clouds (the words with highest mass) associated with system topics generated by our model. We show two different time points each spanning 1 week that has some coverage for most of the events. We see that the system is able to communicate the semantic essence (in terms of words) as well as the temporal profile of each event very effectively.

We perform baseline experiments with Threshold Based TDT as described in section 6.2 for both $k = 1$ and $k = 5$ just like our model. For $k = 1$, the maximum F1 score for the thresholds considered is 41.12 obtained for precision and recall values of 71.65 and 28.83 respectively. The corresponding numbers for $k = 5$ is F1 of 48.16 with precision 39.30 and recall 62.19 respectively. We plot the F1 score in the same way as for the TDT2 dataset for the $k = 1$ case(Figure 5 right panel).

Figure 7: Emerging Topics in IBM Tweets



7. CONCLUSION AND FUTURE WORK

We have developed a new framework for modeling the evolution of topics and aiding the fast discovery of emerging themes in streaming social media content. We have shown the effectiveness and value of novel temporal regularizers in analyzing twitter streams. There are several avenues for future work including a detailed large-scale empirical study of the interplay between the model parameters as well as the tradeoff between evolution and emergence, coming up with convex formulations to avoid local minima problems possibly using nuclear norm regularization[17], and effective means to do model selection in the online setting. Other fascinating directions include incorporating topic volatility into the evolution constraints, and building sparser models using l_0/l_1-regularizers.

8. REFERENCES

[1] J. Allan, editor. *Topic Detection and Tracking: Event-based Information Organization*. Kluwer Academic Publ, 2002.

[2] L. AlSumait, D. Barbara, and C. Domeniconi. On-line lda: Adaptive topic models for mining text streams. In *ICDM*, 2008.

[3] D. Bertsekas. *Non-linear Programming*. Athena Scientific, 1999.

[4] D. Blei and J. Lafferty. Dynamic topic models. In *ICML*, 2006.

[5] D. Blei and M.Jordan. Latent dirichlet allocation. *JMLR*, 3:993–1022, 2003.

[6] Tzu-Chuan Chou and Meng Chang Chen. Using Incremental PLSI for Treshhold-Resilient Online Event Analysis. *IEEE transactions on Knowledge and Data Engineering*, 2008.

[7] A. Cichocki, R. Zdunek, A. H. Phan, and S. Amari. *Non-negative and Tensor Factorizations: Applications to Exploratory Multiway Data Analysis and Blind Source Separation*. Wiley, 2009.

Figure 8: Emerging Topics in IBM Tweets

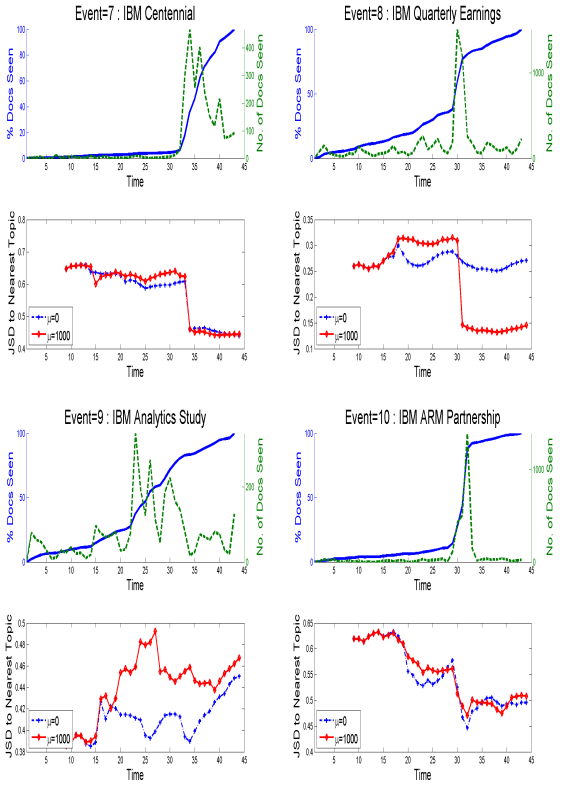
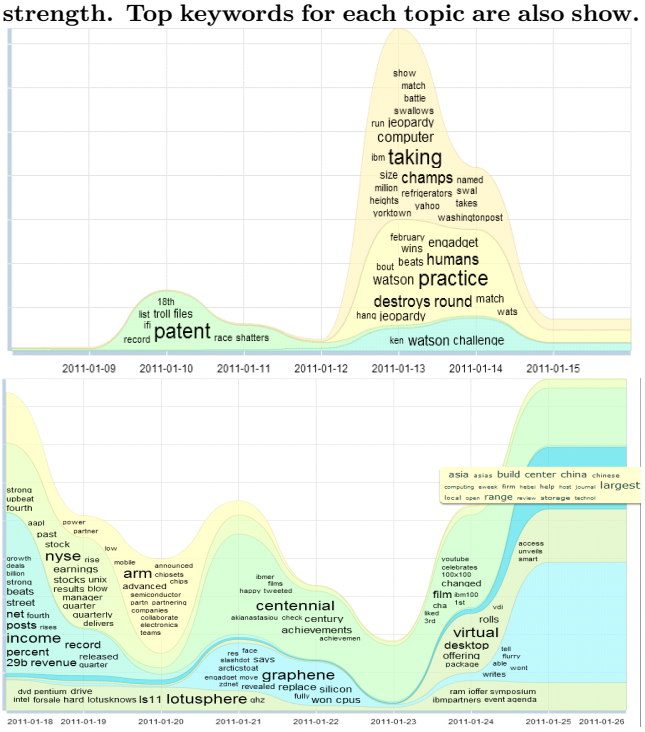


Figure 9: TIARA visualizations [24]: Topics are organized as stacked rivers with width proportional to strength. Top keywords for each topic are also show.



[8] Margaret Connell, Ao Feng, Giridhar Kumaran, Hema Raghavan, Chirag Shah, and James Allan. UMass at TDT 2004. 2004.

[9] Aron Culotta. Towards detecting influenza epidemics by analyzing twitter messages, 2010.

[10] C. Ding, T. Li, and W. Peng. On the equivalence between non-negative matrix factorizations and probabilistic latent semantic analysis. *Computational Statistics and Data Analysis*, 2008.

[11] M. Elad. *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. Springer, 2010.

[12] Mark Girolami and A. Kaban. On an equivalence between plsi and lda. *SIGIR*.

[13] A. Gohr, A. Hinneburg, R. Schult, and M. Spiliopoulou. Topic evolution in a stream of documents. In *SDM*, 2009.

[14] Ngoc-Diep Ho, Paul Van Dooren, and Vincent D. Blondel. Descent methods for nonnegative matrix factorization. *Numerical Linear Algebra in Signals*, abs/0801.3199, 2007.

[15] Matthew D. Hoffman, David M. Blei, and Frances Bach. Online learning for latent dirichlet allocation. In *NIPS*, 2010.

[16] T. Hoffman. Probabilistic latent semantic analysis. In *UAI*, 1999.

[17] M. Jaggi and M. Sulovský. A simple algorithm for nuclear norm regularized problems. In *ICML*, 2010.

[18] D. Lee and H.S. Seung. Learning the parts of objects using non-negative matrix factorizations. *Nature*, 1999.

[19] C.J. Lin. Projected gradient methods for non-negative matrix factorization. In *Neural Computation*, 2007.

[20] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online learning for matrix factorization and sparse coding. *JMLR*, 2010.

[21] P. Melville, V. Sindhwani, and R. Lawrence. Social media analytics: Channeling the power of the blogosphere for marketing insight. *Workshop on Information in Networks*, 2009.

[22] P. M. Pardalos and N. Kovoor. An algorithm for singly constrained class of quadratic programs subject to upper and lower bounds. *Mathematical Programming*, 46:321–328, 1990.

[23] G. Salton. *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*. Addison-Wesley, 1989.

[24] Furu Wei Shimie Pan Michelle X. Zhou Weihong Qian Lei Shi Li Tan Qiang Zhang Shixia Liu, Yangqiu Song. Tiara: Visually analyzing topic evolution in large text collections. In *KDD*, 2010.

[25] Wei Xu, Xin Liu, and Yihong Gong. Document clustering based on non-negative matrix factorization. In *SIGIR*, 2003.

[26] Yiming Yang, Tom Pierce, and James Carbonell. A Study on Retrospective and Online Event Detection. In *SIGIR*, 1998.