Online News Topic Detection and Tracking via Localized Feature Selection

Ola Amayri and Nizar Bouguila

***Abstract*— The detection of topic trends has increasingly attracted interest over the past decades, fueled in particular by the revolution of internet and the emergence of social media. However, manual topic detection and tracking (TDT) is not efficient, this has become possible thanks to the development of modern data mining techniques and their flexibility to model potential issues. A critical challenge in this context is the representation choices of news stories along with adequate detection of new topics. To this end, we propose a unified statistical framework that allows simultaneous topic clustering and feature (word) selection in online settings based on spherical mixtures. Through empirical experiments, the proposed framework demonstrates the ability to learn new topics incrementally and improve detection quality within a reasonable time framework on diverse high-dimensional datasets.**

I. Introduction

IN combination with drastic entailment of using internet, the emergence of social media has supplemented the instant dissemination of news. This revolution entitles each user to play the role of stories producer and stories consumer and in turn poses several personal and technical challenges. Therefore, automated topic detection and tracking (TDT) [1] plays an important role in advancing the state of the art in different types of media to determine informative data. The original research for TDT was initiated in Defense Advanced Research Projects Agency (DARPA) [1] and driven by the demand of deep insight into tremendous amounts of news. Subsequently, TDT has been extensively utilized for different purposes in various technologies due to its importance and practical implications. For example, TDT techniques have been employed in search engines to facilitate the process of finding desired knowledge [2], [3], [4]. Google News alert, for instance, might be used by individuals to receive updates on a given topic in a timely manner [2]. Yet, further enhancement is still needed since recent research has reported that Google news alert reported high false alarm [5], [3].

A. Motivation

TDT models, characterized by the absence/presence of knowledge, can be divided into two main categories: batch (retrospective) topic detection and online topic detection. Batch topic detection, is the unsupervised task that detects the topics associated with relevant stories from the story dataset at hand. On the other hand, online topic detection is defined to immediately examine the news stories as they arise, whether they are related to the existing topics or they form new topics. Both approaches can be cast as clustering problems [6], [7] in the context of machine learning. Generally two main clustering approaches have been considered in TDT, namely, hierarchal clustering and non-hierarchal (flat) clustering. Hierarchal clustering focuses in organizing the news stories in a directed acyclic graph, where topics are presented as nodes in multiple levels that help to reflect the granularity of topics. On the other hand, flat clustering assumes that all topics have the same importance and define different topics as clusters of associated stories [7], [6].
Nevertheless, TDT is a multifaceted issue which requires in-depth analysis, and hence, clustering algorithms should be able to construct flexible TDT models to adapt to real-time demands. For example, the dynamic anatomy of topics makes it difficult to keep pace; while new stories constantly continue to appear, outdated stories may disappear [6]. This in turn stresses the need to adequately update the detection model (clusters) through online learning. Indeed, it also highlights the deficiency of supervised TDT and directs the detection to unsupervised fashion [7]. However, online learning in unsupervised manner (online clustering) confronts several challenges on error constraints, restoring objects labels and finding the number of model's topics (clusters) at each time step [8].
Another challenging fact is that each story tends to discuss several topics rather than a single one and distributed over wide spectrum of areas (e.g. sports, science breakthrough, politics, etc.). Such an assumption is often restrictive as stories in clustering typically represented by features with varying degrees of membership. Thus, it is expected to substantially benefit the application of feature selection in defining the relevancy of stories to different topics (clusters). In unsupervised online learning, however, the main obstacle resides in defining a subset of relevant and irrelevant features at each time step which maintains a good tradeoff between speed and accuracy and yields comparably to better optimal performance rate.

B. Contributions

Motivated by these problems, in this paper, we propose compact and accurate TDT statistical frameworks that are automatically adjustable to dynamic changes. In particular, we formulate TDT issue as a clustering problem of partitioning stories (from different topics) distributed on unit sphere. We prefer clustering as it elegantly deals with high-dimensional data, allows neat mathematical representation of topics, can easily be extended to online settings, and

Ola Amayri and Nizar Bouguila are with the Engineering and Computer Science Faculty, Concordia University, Montreal, Qc, Canada (email: o_amayri@ece.concordia.ca,nizar.bouguila@concordia.ca).

The completion of this research was made possible thanks to the Natural Sciences and Engineering Research Council of Canada (NSERC).

entertains a reasonable computational complexity given its efficiency [9], [8]. Finite mixture models are among the most applied approaches to data clustering. A crucial aspect when employing mixture model is the choice of probability density function that best describes the data [10], [9], [11]. Thus, in this paper spherical distribution, namely, Langevin [12] is adopted as we are dealing with high-dimensional, sparse spherical (L_2-normalized) vectors [11]. Indeed, Langevin mixture model implicitly uses cosine similarity that is easy to interpret and simple to compute for sparse vectors, and has been widely used in text mining [13], spam filtering [10] and topic detection [7], [6]. The estimation of topic distributions (Langevin mixture parameters) can be refined using the common approach of Expectation-Maximization (EM) algorithm in which the determination of the story memberships to a given topic is tackled. Furthermore, we tackle the problem of automatic determination of the number of topics (i.e. model selection) of Langevin mixture model by developing a minimum message length (MML) criterion [10].

As time plays an important role in TDT application, incremental updates of the model is highly desirable. In [9], an algorithm for online learning of von Mises Fisher mixture was proposed based on online spherical K-means for text clustering and was used in [7] to the problem of TDT. The model proposed in [9], however, updates clusters centroids up on the arrival of new documents while keeping the number of clusters fixed which is impractical in realtime TDT models. And hence, we propose to extend the unsupervised online learning framework proposed in [9] to address the problem of creating new topics clusters as their associated stories arise to the detection model with time. To achieve this goal, in this paper we adopt recursive version of EM (RSEM) proposed in [14] to incrementally update TDT model. The approach proposed in [14] updates mixture model parameters based on stochastic gradient descent in the case of Gaussian mixture model (GMM). This method facilitates model calculation, minimizes the misclassification error and maintains fast convergence as compared to other approaches [15].

The majority of research on TDT have stressed the importance of words in identifying the topic of the story [16] and hence have applied feature selection as a preprocessing step [6], [1], [16]. Thus, we enhance further our online TDT model by engaging feature selection in the detection process. Unlike previous research works, we propose a concrete online framework that allows simultaneous feature selection in a unified manner. We approach this by extending feature saliency in [17], [18] to online settings. Thus, our main contributions are:

- We introduce a novel online TDT framework that simultaneously considers feature relevancy, number of topics while gradually updates given model.
- We compare the performance of different online and offline TDT models, and demonstrate the effectiveness and efficiency of using feature selection at assigning stories clusters.
- We present detailed comparison of the proposed frameworks using Langevin mixture (and Von Mises mixture) with the widely used Gaussian mixture model (GMM). Furthermore, we discuss the properties of proposed frameworks on abundant, high-dimensional, directional and challenging data.

The rest of this paper is organized as follows. Section II serves as an introduction to Langevin mixture model and describes proposed design choices for TDT problem. In Section III, we describe our experiments conducted to evaluate the performance of the proposed framework while comparing it to state of the art frameworks. Finally, Section IV discusses the pros and cons of the proposed framework and concludes the paper.

II. TDT Models

In this section, we start by illustrating the procedure of constructing feature vectors from news stories using a common vector space model. Next, we present our proposed TDT models. Table I shows the computational costs for proposed frameworks. Note that the consideration of feature selection in batch and online frameworks doesn't add further computational costs.

TABLE I
COMPUTATIONAL COSTS FOR TDT FRAMEWORKS. N IS THE NUMBER OF STORIES, M IS THE NUMBER OF TOPICS, D IS THE NUMBER OF FEATURES AT TIME T AND D* IS THE NUMBER OF FEATURES AT TIME T+1.

TDT Framework	Computational Cost
Batch TDT	$\mathcal{O}$(NMD)
Batch TDT with Feature Selection	$\mathcal{O}$(NMD)
Online TDT	$\mathcal{O}$(NMD+MD*)
Online TDT with Feature Selection	$\mathcal{O}$(NMD+MD*)

A. Topic Representation

Let $\mathcal{S} = \{\vec{S}_1, \ldots, \vec{S}_N\}$ be a set of N stories and $\mathcal{F}$ be a set of unique features (terms) that affects topics dictionary. In particular, each story $\vec{S}_i = (Y_{i1}, \ldots, Y_{iD})$, where D denotes the total number of features, consists of independent terms in the dictionary $\mathcal{F}$ and then can be modeled as a count vector using TF-IDF (term frequency-inverse document frequency) which is an efficient vector space representation that has been used over the past years in text mining applications [11]. Typically, in TDT models, count vectors are L_2-normalized[1] in an attempt to avoid burstiness phenomenon [16], [6], [7].

B. Batch TDT Model

Generally, each topic consists of many news stories related to it and can be described by a mixture model. Let $p(\vec{S}_i|\Theta)$ be a mixture of M Langevin distributions:

$$p(\vec{S}_i|\Theta) = \sum_{j=1}^{M} p(\vec{S}_i|\theta_j)p_j \quad (1)$$

[1]It is noteworthy that if we consider L_2-normalized vectors, then each topic can be modeled accurately using Langevin distributions (as we shall see in this paper) [10], [9], [11].

where $\Theta = \{\vec{P} = (p_1, \ldots, p_M), \vec{\theta} = (\theta_1, \ldots, \theta_M)\}$ denotes all the parameters of the mixture model such that $\theta_j = (\mu_j, \kappa_j)$ where μ_j is the mean of topic cluster and κ_j is the variance of topics. In addition, $\vec{P}$ represents the vector of mixing weights which must be positive and sum to one and $p(\vec{S}_i|\theta_j) = \frac{\kappa_j^{\frac{D}{2}-1}}{(2\pi)^{\frac{D}{2}} I_{\frac{D}{2}-1}(\kappa_j)} \exp\{\kappa_j \vec{\mu}_j^T \vec{S}_i\}$ is a D-variate Langevin distribution [12] on the $(D-1)$-dimensional unit sphere $\mathbb{S}^{D-1} = \{\vec{S}_i | \vec{S}_i \in \mathbb{R}^D : ||\vec{S}_i|| = (\vec{S}_i^T \vec{S}_i) = 1\}$, with mean direction unit vector $\vec{\mu}_j \in \mathbb{S}^{D-1}$, where $\vec{\mu}_j^T$ denotes the transpose of $\vec{\mu}$, and non-negative real concentration parameter $\kappa_j \geq 0$. Furthermore, $I_D(\kappa)$ denotes the modified Bessel function of first kind and order D [12]. Note that when $D = 2$ we find von Mises model (vM) which is a probability distribution in which the data are concentrated on the circumference of a unit circle.

The underlying probability topic model and its parameters can be estimated using EM algorithm [11] that emphasizes topic-specific words by assigning higher probabilities to words that are more likely to be in the topic and permits the addition of new words to the distributions. Indeed, apart from parameter estimation, another important problem is the automatic selection of the appropriate number of topics M. Among the well-established selection criteria, MML has been repeatedly shown to demonstrate good performance in case of Langevin, Gaussian, Gamma, Poisson, Dirichlet and generalized Dirichlet mixtures models [10], [19], [20], [21], [22], [**?**] by outperforming AIC and MDL (and their variants) approaches. Thus, we propose here the consideration of MML criterion to find the optimal number of mixture components by minimizing the following objective function [23], [10], [24]:

$$\begin{aligned} MessLen(M) \simeq \frac{1}{2}\log\left(\frac{N^{M-1}}{\prod_{j=1}^{M} p(j)} \prod_{j=1}^{M} n_j^D u^2(\kappa_j) v^2(\vec{\mu}_{j,0})\right) \\ - \log\left(\frac{(M-1)!}{S_D^M} \prod_{j=1}^{M}\left[\frac{\kappa_j^{D-1}}{(1+\kappa_j^2)^{\frac{D+1}{2}}}\right]\right) \\ - \log p(\mathcal{S}|\Theta) + \frac{M(D+1)-1}{2}(1 + \log\frac{1}{12}) \end{aligned}$$

where $p(\mathcal{S}|\Theta)$ is the likelihood, with $u(\kappa_j) = \kappa_j^{\frac{1}{2}(D-2)} A_D(\kappa_j)^{\frac{1}{2}(D-1)} \left(\kappa_j - A_D(\kappa_j) - \kappa_j A_D(\kappa_j)^2\right)^{\frac{1}{2}}$ and $v(\vec{\mu}_{j,0}) = \prod_{p=1}^{P-1} \sin^{D-2}\mu_{j,0,p-1}$ where $\vec{\mu}_{j,0} = (\mu_{j,0,1}, \ldots, \mu_{j,0,D})$ denotes the spherical polar coordinates of $\vec{\mu}_j$, $A_D(\kappa_j) = \frac{I_{\frac{D}{2}}(\kappa_j)}{I_{\frac{D}{2}-1}(\kappa_j)}$.

C. Batch TDT Model with Feature Selection

The TDT model based on finite mixture models is explored by grouping similar stories into homogenous topic clusters, where this similarity depends basically on the features (i.e. words) that represent each story. Indeed, researchers have proven over the years the fallacy assumption that the more features representing the document the better discrimination capability the classifier has [18]. This can be due to the presence of noisy and non-informative (i.e. irrelevant) features that generally highly drop the performance. In order to overcome this problem, we adopt the approach proposed in [17], [18], in the case of the Gaussian mixture and von Mises mixture (movM) (i.e. is a 2D special case of Langevin distribution), respectively, that assigns smaller weights to irrelevant features by introducing the notion of feature saliency. It assumes that a given feature is irrelevant if it follows a common density $p(\vec{Y}_{id}|\lambda_{jd})$ across clusters while maintaining the independency of class labels [17].

Following [18], each $d^{th} \in [1, D]$ feature $\vec{Y}_{id} = (Y_{id1}, Y_{id2})$ (such that $Y_{id1} = Y_{id}$) in story $\vec{S}_i$ is represented by movM of two components $p(\vec{Y}_{id}|\theta_{jd})$ and $p(\vec{Y}_{id}|\lambda_{jd}))$ governed by ρ_{jd} that denotes the weight of the d^{th} feature on cluster j. In fact, if ρ_{jd} is very high, then there is no significant difference with the classical movM model $p(\vec{Y}_{id}|\theta_{jd})$ without any saliency. Thus, our model, to take feature selection into account, can be written as:

$$p(\vec{S}_i|\Omega) = \sum_{j=1}^{M} p_j \prod_{d=1}^{D}\left(\rho_{jd} p(\vec{Y}_{id}|\theta_{jd}) + (1-\rho_{jd}) p(\vec{Y}_{id}|\lambda_{jd})\right)$$

where $\Omega = \{\vec{P}, \vec{\mu}_{jd}, \vec{\kappa}_{jd}, \{\rho_{jd}\}, \{\lambda_{jd}\}\}$, and $\lambda_{jd} = (\vec{\mu}_{jd|\lambda}, \kappa_{jd|\lambda})$ are the parameters of vM from which the irrelevant feature is drawn. And $p(\vec{Y}_{id}|\theta_d) = \frac{1}{2\pi I_0(\kappa_d)} \exp\{\kappa_d \vec{\mu}_d^T \vec{Y}_{id}\}$ is vM distribution, I_0 is the modified Bessel function of the first kind and order zero [12], $\theta_d = (\vec{\mu}_d, \kappa_d)$, $\vec{\mu} = (\vec{\mu}_1, \ldots, \vec{\mu}_D)$, $\vec{\mu}_d = (\mu_{d1}, \mu_{d2})$ is the mean direction, $\vec{\kappa} = (\kappa_1, \ldots, \kappa_D)$, κ_d is the concentration parameter and $(\vec{Y}_{id})^T \vec{Y}_{id} = 1$.

Our intention is to develop TDT model that allows to learn topic mixture model while considering simultaneously the relevancy of features. To achieve this goal, we need to minimize the following objective function to simultaneously find the optimal number of topics (clusters) and estimate features saliencies:

$$\begin{aligned} S(\Theta, \mathcal{S}) &= -MessLen_{FS}(M) + \xi(1 - \sum_{j=1}^{M} p_j) \\ &+ \sum_{d=1}^{D} \nu_{jd}(1 - \rho_{jd1} - \rho_{jd2}) \end{aligned} \quad (2)$$

where

$$\begin{aligned} MessLen_{FS}(M) &= \frac{1}{2}(M + 5MD - 1)\log N \\ &+ \frac{D}{2}\sum_{j=1}^{M}\log p_j + \frac{M}{2}\sum_{d=1}^{D}\log\rho_{jd} + \frac{M}{2}\sum_{d=1}^{D}\log(1-\rho_{jd}) \\ &+ \frac{1}{2}\sum_{j=1}^{M}\sum_{d=1}^{D}\log(|F_1(\theta_{jd})|) + \frac{1}{2}\sum_{j=1}^{M}\sum_{d=1}^{D}\log(|F_1(\lambda_{jd})|) \\ &+ \frac{N_p}{2}(1 + \log\frac{1}{12}) - \log p(\mathcal{S}|\Omega) \\ &+ \sum_{j=1}^{M}\sum_{d=1}^{D}[4\log\pi + \log(1+\kappa_{jd}^2) + \log(1+\kappa_{jd|\lambda}^2)] \end{aligned}$$

$\rho_{jd1} = \rho_{jd}$, $\rho_{jd2} = 1 - \rho_{jd}$, ξ and ν_{jd} are Lagrange multipliers to satisfy the constraints $\sum_{j=1}^{M} p_j = 1$ and $\rho_{jd1} + \rho_{jd2} = 1$, respectively. Next, we use common EM approach to learn model parameters using Eq.(2) (interested readers can see [18] and references therein). A full batch TDT approach with feature selection is summarized in Algorithm 1.

Algorithm 1 Batch TDT with feature selection

INPUT: Set of N D-dimensional stories $\mathcal{S}$ on $\mathbb{S}^{D-1}$.
OUTPUT: Topics clusters of $\mathcal{S}$ with salient features.

1: Choose initial number of topics.
2: **for** each candidate value of topics **do**
3: Apply spherical K-means [25] to obtain the initial parameters for each component.
4: **repeat**
5: **E-Step**: Update model posterior [18].
6: **M-Step**: Update p_j, ρ_{jd}, $\vec{\mu}_{jd}$, $\vec{\mu}_{jd|\lambda}$, $\vec{\kappa}_{jd}$ and $\vec{\kappa}_{jd|\lambda}$ [18].
7: **until** Convergence
8: Select the optimal model M^* such that $M^* = \arg\min MessLen_{FS}(M)$.
9: **end for**

D. Online TDT Model

As we mentioned in the introduction, in real-time news appear in stream of sequences and hence online learning is favored over off-line counterpart. The main idea is to update topic (mixture) model parameters incrementally as stories are presented to the classifier. Formally, assume that at time t we have a set of N stories $\mathcal{S} = \{\vec{S}_1, \ldots, \vec{S}_N\}$ which is represented by M topics of Langevin mixture with parameters Θ_N^t. At time $t+1$ a new story $\vec{S}_{N+1}$ is introduced and the model should be updated considering the new story. Indeed, the new story might be assigned to already defined topics or it might be *First Story* creating a new cluster. In this section, we develop flexible online TDT model that automatically finds the optimal number of topics (model components) at each time step. To achieve this goal, in the following we adopt RSEM approach proposed in [14]. The same way as EM, the RSEM is mainly obtained by computing the posterior probability $\hat{Z}_{N+1,j} = \frac{p_j^t p(\vec{S}_{N+1}|\Theta_N^t)}{\sum_{j=1}^{M} p_j^t p(\vec{S}_{N+1}|\Theta_N^t)}$ of the complete available data $(\mathcal{S}_{N+1}, \mathcal{Z}_{N+1})$ at time $t+1$ in the E-step, where $\mathcal{S}_{N+1} = \{\vec{S}_1, \ldots, \vec{S}_{N+1}\}$ is a finite set of $N+1$ stories and $\mathcal{Z}_{N+1} = \{Z_{N+1,1}, \ldots, Z_{N+1,M}\}$ is a random vector of missing data that indicates if the story is associated with topic j, such as $Z_{N+1,j} = 1$ if the new story $\vec{S}_{N+1}$ belongs to topic j, 0 otherwise. Using RSEM, in the M-step we update the model parameters with respect to $\Theta^t = \{p_j, \Psi = \{\vec{\mu}_j, \kappa_j\}\}$ and with the constraints $0 < p_j \leq 1$ and $\sum_{j=1}^{M} p_j = 1$:

$$\Psi_{N+1}^{t+1} = \Psi_N^t + \gamma_N \frac{\partial \log(p(\mathcal{S}_{N+1}, \mathcal{Z}_{N+1}|\Psi_N^t))}{\partial \Psi_N^t}, \quad 1 \leq j \leq M$$

where γ_N represents any sequence of positive numbers which decreases to zero or positive definite matrix such that $\sum|\gamma_N| = \infty$ and $\sum|\gamma_N|^2 < \infty$. In our case we have chosen $\gamma_N = \frac{1}{N+1}$ [8], [14]. Note that to ensure the unity of the mixing proportion p_j we introduce the Logit transform $w(j) = \log\frac{p_j}{p_M}$ such that $w_M = 0$, where:

$$w(j)^{t+1} = w(j)^t + \gamma_N(\hat{Z}_{N+1,j} - p_j^t), \quad 1 \leq j < M$$

$$p_j^{t+1} = \frac{\exp(w(j)^{t+1})}{1 + \sum_{j=1}^{M-1}\exp(w(j)^{t+1})}, \quad j = 1, ..., M-1$$

$$p_M^{t+1} = \frac{1}{1 + \sum_{j=1}^{M-1}\exp(w(j)^{t+1})}$$

E. Online TDT with Feature Selection

In this section, our intention is to develop flexible and accurate online mixture model that lets us to simultaneously choose relevant features and optimal number of model components. To achieve this goal, in the following we extend our proposed batch TDT with feature selection in section II-C to online settings. We assume that the feature saliences are mutually independent and also independent of the hidden component label for any story. Thus,

$$\begin{aligned} p(\mathcal{S}_{N+1}, \mathcal{Z}_{N+1}, \vec{\Lambda}_{N+1}) &= p(\mathcal{S}_{N+1}|\mathcal{Z}_{N+1}, \vec{\Lambda}_{N+1}) \\ &\times\ p(\mathcal{Z}_{N+1})p(\vec{\Lambda}_{N+1}) \end{aligned}$$

and hence, in the E-step we compute the complete data $(\mathcal{S}_{N+1}, \mathcal{Z}_{N+1}, \vec{\Lambda}_{N+1})$ log-likelihood based on Θ_N^t as:

$$\begin{aligned} &E[\log p(\mathcal{S}_{N+1}, \mathcal{Z}_{N+1}, \vec{\Lambda}_{N+1})] \\ &= \sum_{i=1}^{N}\sum_{j=1}^{M} p(Z_{N+1,j} = j|S_{N+1,d}) \times \log p_j \\ &+ \sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{d=1}^{D}\sum_{\phi=0}^{1} p(Z_{N+1,j} = j, \phi_{jd}|S_{N+1,d}) \\ &\times \Big[\phi_{jd}(\log p(Y_{N+1,d}|\theta_{jd}) + \log\rho_{jd}) \\ &+ (1-\phi_{jd})(\log p(Y_{N+1,d}|\lambda_{jd}) + \log(1-\rho_{jd}))\Big] \end{aligned}$$

where $\mathcal{Z}_{N+1} = (Z_{N+1,1}, \ldots, Z_{N+1,M})$ is a random vector of missing data that indicates if the new document is associated with component j, such as $Z_{N+1,j} = 1$ if the new story $\vec{S}_{N+1}$ belongs to class j, 0 otherwise. $\vec{\Lambda}_{N+1} = \{\phi_{11}, \ldots, \phi_{MD}\}$ is a random vector of missing data that indicates if feature d is relevant to cluster j, where $\rho_{jd} = p(\phi_{jd} = 1)$.

In the M-step we update the model parameters with respect to Φ^t and with the constraints $0 < p_j \leq 1$ and $\sum_{j=1}^{M} p_j = 1$:

$$\Phi^{(t+1)} = \Phi^{(t)} + \gamma_N \frac{\partial \log(p(\mathcal{S}_{N+1}, \mathcal{Z}_{N+1}, \vec{\Lambda}_{N+1}|\Phi^{(t)}))}{\partial\Phi} \quad (3)$$

$$w(j)^{t+1} = w(j)^t + \gamma_N(\hat{Z}_{N+1,j} - p_j^t), \quad 1 \leq j < M \quad (4)$$

with $p_j^{t+1} = \frac{\exp(w(j)^{t+1})}{1+\sum_{j=1}^{M-1}\exp(w(j)^{t+1})}$ such that $1 \leq j \leq M-1$ and $p_M^{t+1} = \frac{1}{1+\sum_{j=1}^{M-1}\exp(w(j)^{t+1})}$, where $\hat{Z}_{N+1,j} = \frac{p_j^t p(\vec{S}_{N+1}|\Phi^t)}{\sum_{j=1}^{M} p_j^t p(\vec{S}_{N+1}|\Phi^t)}$ is the posterior probability, $\Phi^t =$

$\{\mu_{jd}^t, \kappa_{jd}^t, \mu_{jd|\lambda}^t, \kappa_{jd|\lambda}^t\}$, and $\gamma_N = \frac{1}{N+1}$ [14]. In order to figure out the relevancy of features for the new stories, we need to update ρ_{jd} such that $\rho_{jd} \in [0,1]$. Let $\rho_{jd1} = \rho_{jd}$ and $\rho_{jd2} = 1 - \rho_{jd}$ such that $\rho_{jd1} + \rho_{jd2} = 1 \ \forall d = 1, \ldots, D$. Hence, we propose to use parametrization based on Logit transform $h_{jd} = \log(\rho_{jd}) = \log \frac{\rho_{jd}}{1-\rho_{jd}}$, and we obtain:

$$h_{jd}^{t+1} = h_{jd}^t + \frac{\hat{\Lambda}_{jd}}{N+1}\left[\frac{\partial \log(p(\mathcal{S}_{N+1}, \mathcal{Z}_{N+1}, \vec{\Lambda}_{N+1}|\rho_{jd}^{(t)}))}{\partial \rho_{jd}}\right] \quad (5)$$

$$\rho_{jd1}^{t+1} = \frac{\exp(h_{jd}^{t+1})}{1+\exp(h_{jd}^{t+1})}, \quad d = 1, \ldots, D \quad (6)$$

with $\hat{Z}_{N+1,j} \frac{\rho_{jd}^t p(Y_{N+1,d}|\theta_{jd}^t)}{\rho_{jd}^t p(\vec{Y}_{N+1,d}|\theta_{jd}^t) + (1-\rho_{jd}^t) p(\vec{Y}_{N+1,d}|\lambda_{jd}^t)}$

If feature d is relevant, and $\hat{\Lambda}_{jd} = \hat{Z}_{N+1,j} \frac{(1-\rho_{jd}^t) p(Y_{N+1,d}|\lambda_{jd}^t)}{\rho_{jd}^t p(\vec{Y}_{N+1,d}|\theta_{jd}^t) + (1-\rho_{jd}^t) p(\vec{Y}_{N+1,d}|\lambda_{jd}^t)}$ otherwise. Finally, the online TDT algorithm considering the feature selection is summarized in Algorithm 2. Note that in our model, we find the optimal number of clusters by running MML model concurrently for models $\{M_{min}, \ldots, M_{max}\}$ and select the solution which minimize message length. For computational complexity sake, when the number of components is large and the estimation of the candidate model is slow, we keep all the candidate model fitting with $\{\Theta_{M_{min}}, \ldots, \Theta_{M_{max}}\}$.

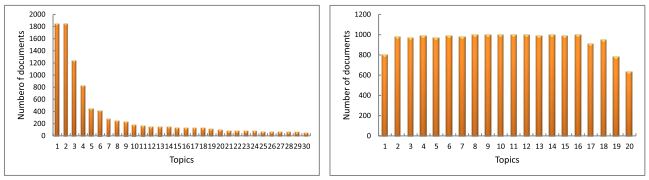
Algorithm 2 Online TDT with feature selection

INPUT: $\vec{S}_{N+1}$, $\{M_{min}, \ldots, M_{max}\}_N^t$, $\{\rho jd\}_N^t$ and Θ_N^t
OUTPUT: Θ_{N+1}^{t+1}, $\{\rho jd\}_{N+1}^{t+1}$ and M_{N+1}^*

1: **for** each candidate value of topics **do**
2: Compute the posterior probabilities $\hat{Z}_{N+1,j}$.
3: Assign $\vec{S}_{N+1}$ to cluster which maximizes $\hat{Z}_{N+1,j}$
4: Update the different mixtures models using Eqs.(3)-(6) for $j \in \{M_{min}, \ldots, M_{max}\}$.
5: if ρ_{jd}^{t+1} approaches to zero we can discard feature d.
6: Select the optimal model M^* such that $M^* = \arg\min MessLen_{FS}(M)$ for $M^* \in \{M_{min}, \ldots, M_{max}\}$.
7: **end for**

III. Experimental Results

In this section, experiments have been carried out to evaluate the effectiveness of the proposed learning frameworks. The goal of the experiments is twofold. First, we investigate the impact of feature selection on improving the overall TDT performance. Second, we determine how online learning is desirable with dynamic data. To achieve this goal, we run experiments to compare: batch TDT model based on Langevin mixture model with (LMMFS) and without feature selection (LMM) along with Online TDT model with and without feature selection that were proposed in sections II-B, II-C, II-D and II-E, respectively. We compare our results with other mixture and non-mixture models, which are:



(a) TDT-2 (b) 20 Newsgroup

Fig. 1. Analysis of datasets.

Gaussian mixture model (GMM)[2] (where previous scenarios were applied, also), K-means, Spherical K-means, online vMF [6] and latent Dirichlet allocation (LDA). We illustrate our results on high-dimensional, sparse and challenging real-world datasets. In our experiments, we initialize the number of components to $M_{max} = 40$, and the feature saliency values are initialized at ($\rho = 0.5$). We run all the tested algorithms 5 times for evaluation.

A. Datasets

TDT results are presented on two public news datasets, which are: The Topic Detection and Tracking (TDT-2) dataset [26] and 20 Newsgroup dataset[3]. TDT-2 dataset contains news stories classified into 96 topics and has been collected in 1998 from six sources: two newswires (Associated Presss World Stream and New York Times), two radio programs (Voice of America and Public Radio Internationals The World) and two television programs (CNN and ABC). The TDT-2 corpus is subdivided into three two-month sets: a training set (Jan-Feb), a development test set (Mar-Apr), and an evaluation set (May-Jun). In preprocessing step, we removed the documents that belong to several topics, and hence, only 30 topics were left, resulting in 9394 patterns over 36771 dimensions. On the other hand, 20 Newsgroup was collected from UseNet postings over several months in 1993. In our experiments, we find 26214 patterns over 24876 dimensions classified into 20 categories.
Figure 1 shows the distribution of documents over topics in both datasets. It is clear that TDT-2 is unbalanced where some topics have less than 60 documents while others have more than 18000 document. On the other hand, in 20 Newsgroup dataset documents are fairly distributed over different topics.

B. Evaluation Metrics

We evaluated the proposed frameworks for TDT problem using typical evaluation criteria that have been used, for instance, in the context of text clustering. We reported the execution time of the batch framework on an Intel(R) Core(TM) 64 Processor PC with the Windows XP Service Pack 3 operating system and a 4 GB main memory. While in the online, we reported the average time to assign new

[2]Details about the learning of GMM can be found in [17], [14].
[3]http://kdd.ics.uci.edu/databases/20newsgroups

story to a topic. Moreover, we calculated F_1(micro-averaged) measure as follows:

$$F_1(\text{micro-averaged}) = \frac{2\times\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}} \quad (7)$$

where

$$\text{Precision} = \frac{\text{number of documents correctly predicted in class}i}{\text{number of documents in class}i}$$

$$\text{Recall} = \frac{\text{number of documents correctly predicted in class}i}{\text{number of correct prediction of class}i}$$

It is worth mentioning that the larger values of $F_1 \in (0,1)$ represent higher classification quality. In addition, we calculated normalized mutual information (NMI) [7] criterion as an external measure of how well the clustering results conform to existing class labels:

$$NMI = \frac{\sum_{j,c} N_{j,c} \log \frac{N_{j,c}}{N_j N_c}}{\sqrt{(\sum_j N_j \log \frac{N_j}{N})(\sum_c N_c \log \frac{N_c}{N})}}$$

where N_j is the number of stories in cluster j, N_c is the number of stories in true classes labels c, and $N_{j,c}$ is the number of stories that are in cluster j and class c. The larger value of NMI reflects better clustering. It is noteworthy that NMI, is unbiased towards high number of clusters M as purity and entropy criteria. Moreover, in online TDT framework we used $Cost$ function[4] that has been used as a standard evaluation criterion in TDT-2 [1].

C. Results

We start by preparing our data by extracting the text of the news articles. Next, we applied stemming and removed stop words. This gives us vocabulary of words for each dataset. Each article is then described as a L_2-normalized frequency vector as described in Section II-A.

1) Batch TDT Experiments: we compare the performance of the batch TDT algorithm we proposed in Section II-B with batch TDT algorithm with feature selection (Section II-C) on both datasets. Tables II and III illustrate the average normalized mutual information (NMI), the estimated number of components (M^*) and run time results averaged over 5 runs. All results in these tables are shown in the format of $average \pm standarddeviation$. According to these tables, it is clear that the best results were obtained when TDT framework was applied using feature selection. Note that engaging feature selection has not only improved detection but also accelerated the clustering. In particular, using LMM shows a better detection over the rest of the models in all experiments. Indeed, we clearly see slight improvement of LMM detection over GMM which has been extensively used in the past. This result is expected since we are modeling vector of stories that are defined on the unit hypershpere (i.e. non-Gaussian vectors) and hence Langevin mixture (spherical distribution) provides a better clustering.

Figure 2 shows also how the choice of the number of features influences the accuracy of detection. In all the

[4]Note that $Cost$ evaluation and its visual tool Detection Error Tradeoff curves (DET) are biased to the value of user-defined parameters [1]. Indeed, authors in [7] have shown that Langevin mixture fails to perform well on DET curve and achieved minimal cost of 70%.

TABLE II

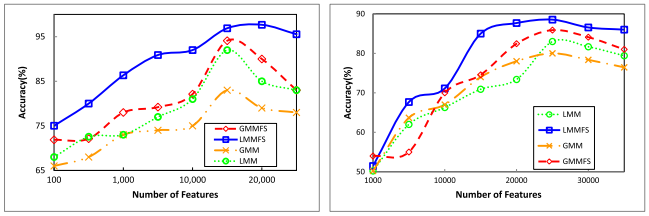
PERFORMANCE OF BATCH TDT FRAMEWORK FOR TDT-2 DATASET BASED ON DIFFERENT MODELS AVERAGED OVER 5 RUNS

	Evaluation Criteria		
	NMI	M^*	Run Time (sec)
LMM	0.71	30±1.90	390
GMM	0.70	30±2.10	410
LMM+FS	0.89	30±0.14	230
GMM+FS	0.79	30±1.36	294
Spherical k-means	0.67	29±1.56	488
k-means	0.66	27±0.20	491
LDA	0.60	30±4.01	400

TABLE III

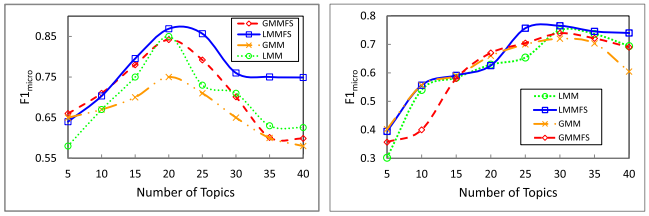
PERFORMANCE OF BATCH TDT FRAMEWORK FOR 20 NEWSGROUP DATASET BASED ON DIFFERENT MODELS AVERAGED OVER 5 RUNS

	Evaluation Criteria		
	NMI	M^*	Run Time (sec)
LMM	0.68	20±2.56	320
GMM	0.67	20±2.09	356
LMM+FS	0.76	20±0.49	197
GMM+FS	0.74	20±0.01	211
Spherical k-means	0.58	21±2.00	500
k-means	0.54	18±5.41	510
LDA	0.60	20±0.11	350
EDCM [6]	0.54	-	934



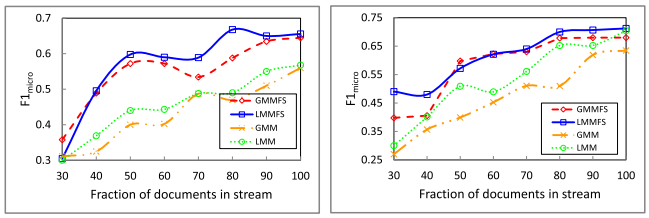
(a) 20 Newsgroup (b) TDT-2

Fig. 2. Detection accuracy vs. the number of features based on Langevin mixture model (LMM) & Gaussian mixture model (GMM).



(a) 20 Newsgroup (b) TDT-2

Fig. 3. F_1(micro-averaged) vs number of topics in Batch TDT framework with and without feature selection based on Langevin mixture model (LMM) & Gaussian mixture model (GMM).



(a) 20 Newsgroup (b) TDT-2

Fig. 4. F_1(micro-averaged) vs fraction of documents in Online TDT framework with and without feature selection based on Langevin mixture model (LMM) & Gaussian mixture model (GMM)

experiments, both Langevin mixture (LMM) and Gaussian mixture (GMM) models achieve their higher value at $M = 20$ and $M = 30$ for 20 newgroup and TDT-2 datasets, respectively, which is the correct number of topics.

Indeed, we can clearly see that we obtained significantly better results, using TDT feature selection, in terms of NMI, and running time than in [6], where vMF, LDA and EDCM were used for 20 Newsgroup. In another interesting work authors in [27] have evaluated TDT-2 dataset using five approaches, namely, Canonical K-means, K-means clustering in the principle components subspace, Normalized Cut, Graph Regularized Nonnegative Matrix Factorization (GNMF), Nonnegative Matrix Factorization (NMF). It is noteworthy though that our proposed TDT framework has a comparable performance with GNMF framework, in terms of NMI, which achieved the best results in all their experiments.

2) Online TDT Experiments: Here, we compare the performance of the online TDT algorithm on both datasets. We first arranged stories in both datasets in chronological order. Then, we selected the oldest 2000 articles from both datasets to initialize, where we clustered using the algorithms proposed in Section II-B and II-C. Later, we used online algorithms (Section II-D and Section II-E) each time we insert new story until the end. We initialize cluster components to $M_{min} = 5$ and $M_{max} = 40$ for all our experiments.

Tables IV and V present the results on both datasets. In terms of running time, in all the experiments, both Langevin and Gaussian mixtures show very similar speed. However, GMM shows worse performance in terms of NMI and detection of optimal number of components. Moreover, we can observe that the online algorithms give worse NMI results than their batch counterparts, which is expected since the online algorithms can only update the cluster statistics incrementally. In addition, Langevin mixture model appeared to be more effective in terms of TDT-2 cost than Gaussian mixture (see Table VI).

Figure 4 shows that the F_1(micro-averaged) over different number of stories fed to the system over time. Note that both Langevin and Gaussian achieve best F_1(micro-averaged) when we insert the whole set of documents in both datasets. According to those figures we notice again that using feature selection as apart of online learning has improved the quality of the clusters.

The performance improvement in the case of online TDT framework is rather promising, comparing to the results in [6] our proposed framework has not only showed improved clustering quality $NMI = 0.54$ to $NMI = 0.58$ but also afforded faster clustering, where in [6] run time equals 0.011 as compared to 0.006 per story in our framework.

TABLE IV

PERFORMANCE OF ONLINE TDT FRAMEWORK FOR TDT-2 DATASET BASED ON DIFFERENT MODELS AVERAGED OVER 5 RUNS

	Evaluation Criteria		
	NMI	M^*	Run Time (sec)
LMM	0.59	28±2.2	0.012
LMM+FS	0.61	30±3.09	0.007
GMM	0.42	28±2.61	0.020
GMM+FS	0.56	29±2.03	0.011
Spherical k-means	0.53	27±1.5	0.013
k-means	0.41	27±3.11	0.021
Online vMF	0.58	28±0.98	0.014
LDA	0.58	28±0.21	0.012

TABLE V

PERFORMANCE OF ONLINE TDT FRAMEWORK FOR 20 NEWSGROUP DATASET BASED ON DIFFERENT MODELS AVERAGED OVER 5 RUNS

	Evaluation Criteria		
	NMI	M^*	Run Time (sec)
LMM	0.56	19±1.52	0.009
LMM+FS	0.58	20±1.04	0.006
GMM	0.49	19±1.32	0.011
GMM+FS	0.51	18±2.04	0.009
Spherical k-means	0.52	18±2.61	0.011
k-means	0.48	18±0.45	0.015
Online vMF	0.56	19±0.01	0.010
LDA	0.35	18±0.02	0.014
EDCM [6]	0.39	-	0.565

D. Previous work

An alternative model for TDT has been presented by a family of probabilistic models that exploits latent structure in stories. Under this approach, topics were described as hidden variables using multinomial distribution over all words rather than clusters of stories [28], [6], [29], [7]. Examples include latent Dirichlet allocation (LDA) model, probabilistic latent semantic indexing (pLSI), Dirichlet compound multinomial (DCM) models and their variants. However, this approach suffers from high computational complexity, for instance,

TABLE VI

EVALUATION OF THE PERFORMANCE *Cost* OF ONLINE TDT FRAMEWORK WITH AND WITHOUT FEATURE SELECTION BASED ON LANGEVIN MIXTURE MODEL (LMM) & GAUSSIAN MIXTURE MODEL (GMM) ON TDT-2 DATASET.

Dataset	LMM	GMM
Online	0.0216	0.0499
Online with Feature Selection	0.0059	0.0077

using pLSI models the number of parameters was increased linearly with the size of documents [29]. Moreover, authors in [6] have shown that von Mises Fisher mixture (a.k.a Langevin mixture [10]) indeed performs better than LDA and DCM in several TDT experiments for offline and online scenarios. Furthermore, authors in [30] have provided an extensive comparison of TDT models using spherical mixtures and non-mixture models.

IV. Conclusion

In this paper, we proposed statistical TDT frameworks based on Langevin mixture model that facilitate topic detection in both offline and online scenarios. Our framework is motivated by the desire to construct flexible and accurate TDT that simultaneously determines relevant features, number of clusters while being able to incrementally update model's parameters. To this aim, the unsupervised learning of proposed model was handled through EM and RSEM in off-line and online scenarios, respectively. The automatic discovery of topics number was taken into account via MML approach. Empirical experiments on challenging data sets have proven that the proposed algorithms have yielded to good performance and improved the quality of clustering. Proposed frameworks are well-justified and can be adjusted to different problems. A promising future work, that we are working on, is the extension of the proposed model to the infinite case using Dirichlet processes. Moreover, instead of the restrictive assumption of assigning each story to one topic we can extend our work to assign each story to multiple topics (mixed membership).

References

[1] J. Allan, J. Carbonell, G. Doddington, J. Yamron, and Y. Yang, "Topic detection and tracking pilot study final report," in *Proc. of the DARPA Broadcast News Transcription and Understanding Workshop*, 1998.

[2] V. Hatzivassiloglou, L. Gravano, and A. Maganti, "An investigation of linguistic features and clustering algorithms for topical document clustering," in *Proc. of the 23rd ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, 2000, pp. 224–231.

[3] Q. He, K. Chang, and E.-P. Lim, "Analyzing feature trajectories for event detection," in *Proc. of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, ser. SIGIR, 2007, pp. 207–214.

[4] X.-Y. Dai, Q. Chen, X. Wang, and J. Xu, "Online topic detection and tracking of financial news based on hierarchical clustering," in *Proc. of International Conference on Machine Learning and Cybernetics (ICMLC)*, 2010, pp. 3341–3346.

[5] J. Cheng, J. Zhou, and S. Qiu, "Fine-grained topic detection in news search results," in *Proc. of the 27th Annual ACM Symposium on Applied Computing (SAC)*, 2012, pp. 912–917.

[6] A. Banerjee and S. Basu, "Topic models over text streams: A study of batch and online unsupervised learning," in *Proc. of the SIAM International Conference on Data Mining (SDM)*, 2007, pp. 437–442.

[7] Q. He, K. Chang, E.-P. Lim, and A. Banerjee, "Keep it simple with time: A re-examination of probabilistic topic detection models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 10, pp. 1795–1808, 2010.

[8] N. Bouguila and D. Ziou, "Online clustering via finite mixtures of dirichlet and minimum message length," *Engineering Applications of Artificial Intelligence*, vol. 19, pp. 371–379, 2006.

[9] A. Banerjee and J. Ghosh, "Frequency-sensitive competitive learning for scalable balanced clustering on high-dimensional hyperspheres," *IEEE Transactions on Neural Networks*, vol. 15, no. 3, pp. 702–719, may 2004.

[10] O. Amayri and N. Bouguila, "Probabilistic clustering based on langevin mixture," in *Proc. of the 10th International Conference on Machine Learning and Applications (ICMLA)*, vol. 2, 2011, pp. 388–391.

[11] A. Banerjee, I. S. Dhillon, J. Ghosh, and S. Sra, "Clustering on the unit hypersphere using von mises-fisher distributions," *Journal of Machine Learning Research*, vol. 6, pp. 1345– 1382, 2005.

[12] K. V. Mardia, *Statistics of directional data*. Academic Press, 1972.

[13] G. Salton and M. J. McGill, *Introduction to Modern Retrieval*. McGraw-Hill Book Company, 1983.

[14] J. F. Yao, "On recursive estimation in incomplete data models," *Statistics*, vol. 34, no. 1, pp. 27–51, 2000.

[15] J. Ratsaby, "A stochastic gradient descent algorithm for structural risk minimisation," in *Algorithmic Learning Theory*, ser. Lecture Notes in Computer Science, R. Gavalda, K. Jantke, and E. Takimoto, Eds. Springer Berlin, Heidelberg, 2003, vol. 2842, pp. 205–220.

[16] Q. He, K. Chang, and E.-P. Lim, "Using burstiness to improve clustering of topics in news streams," in *Proc. of the 7th IEEE International Conference on Data Mining (ICDM)*, 2007, pp. 493–498.

[17] M. H. C. Law, M. A. T. Figueiredo, and A. K. Jain, "Simultaneous feature selection and clustering using mixture models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp. 1154–1166, 2004.

[18] O. Amayri and N. Bouguila, "Unsupervised feature selection for spherical data modeling: Application to image-based spam filtering," in *Proc. of the 5th International Conference on Multimedia Communications, Services and Security (MCSS)*, ser. Communications in Computer and Information Science, vol. 287. Springer Berlin Heidelberg, 2012, pp. 13–23.

[19] W. P. Bouberima, M. Nadif, and Y. K. Bencheikh, "Assessing the number of clusters from a mixture of von mises-fisher," in *Proc. of the World Congress on Engineering (WCE)*, 2010.

[20] D. Ziou and N. Bouguila, "Unsupervised learning of a finite gamma mixture using MML: Application to SAR image analysis," in *Proc. of 17th International Conference on Pattern Recognition (ICPR)*, 2004, pp. II.68–II.71.

[21] N. Bouguila and D. Ziou, "MML-based approach for finite dirichlet mixture estimation and selection," in *Proc. of the 4th International Conference on Machine Learning and Data Mining in Pattern Recognition*, ser. Lecture Notes in Computer Science, vol. 3587. Springer, 2005, pp. 42–51.

[22] ——, "MML-based approach for high-dimensional learning using the generalized dirichlet mixture," in *CVPR '05: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Workshops*. IEEE Computer Society, 2005, p. 53.

[23] ——, "On fitting finite dirichlet mixture using ECM and MML," in *ICAPR (1)*, ser. Lecture Notes in Computer Science, S. Singh, M. Singh, C. Apté, and P. Perner, Eds., vol. 3686. Springer, 2005, pp. 172–182.

[24] R. Baxter and J. Oliver, "Finding Overlapping Components with MML," *Statistics and Computing*, vol. 10, no. 1, pp. 5–16, 2000.

[25] O. Amayri and N. Bouguila, "Beyond hybrid generative discriminative learning: Spherical data classification," *Pattern Analysis and Applications*, 2013, in press.

[26] I. S. Dhillon and D. S. Modha, "Concept Decompositions for Large Sparse Text Data Using Clustering," *Machine Learning*, vol. 42, no. 1-2, pp. 143–175, 2001.

[27] D. Cai, X. He, and J. Han, "Document clustering using locality preserving indexing," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 12, pp. 1624–1637, 2005.

[28] D. Cai, X. He, J. Han, and T. S. Huang, "Graph regularized nonnegative matrix factorization for data representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1548–1560, 2011.

[29] T.-C. Chou and M. C. Chen, "Using incremental plsi for threshold-resilient online event analysis," *IEEE Transaction on Knowledge and Data Engineering*, vol. 20, no. 3, pp. 289–299, 2008.

[30] J.-T. Chien and C.-H. Chueh, "Topic-based hierarchical segmentation," *IEEE Transactions on Audio, Speech & Language Processing*, vol. 20, no. 1, pp. 55–66, 2012.

[31] J. Reisinger, A. Waters, B. Silverthorn, and R. J. Mooney, "Spherical topic models," in *Proc. of the 27th International Conference on Machine Learning (ICML)*, 2010.