Mining Named Entities with Temporally Correlated Bursts from Multilingual Web News Streams

Alexander Kotov
akotov2@illinois.edu

ChengXiang Zhai
czhai@cs.uiuc.edu

Richard Sproat
rws@illinois.edu

Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, IL 61801

ABSTRACT

In this work, we study a new text mining problem of discovering named entities with temporally correlated bursts of mention counts in multiple multilingual Web news streams. Mining named entities with temporally correlated bursts of mention counts in multilingual text streams has many interesting and important applications, such as identification of the latent events that attracted the attention of on-line media in different countries, and valuable linguistic knowledge in the form of transliterations. While mining "bursty" terms in a single text stream has been studied before, the problem of detecting terms with temporally correlated bursts in multilingual Web streams raises two new challenges: (i) correlated terms in multiple streams may have bursts that are of different orders of magnitude in their intensity and (ii) bursts of correlated terms may be separated by time gaps. We propose a two-stage method for mining items with temporally correlated bursts from multiple data streams, which addresses both challenges. In the first stage of the method, the temporal behavior of different entities is normalized by modeling them with the Markov-Modulated Poisson Process. In the second stage, a dynamic programming algorithm is used to discover correlated bursts of different items that can be potentially separated by time gaps. We evaluated our method with the task of discovering transliterations of named entities from multilingual Web news streams. Experimental results indicate that our method can not only effectively discover named entities with correlated bursts in multilingual Web news streams, but also outperforms two state-of-the-art baseline methods for unsupervised discovery of transliterations in static text collections.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications—*Data mining*; I.2.7 [**Artificial Intelligence**]: Natural Language Processing—*Text analysis*

General Terms

Algorithms, Experimentation

Keywords

Text streams, correlated burst detection, probabilistic modeling, dynamic programming

1. INTRODUCTION

The vast amounts and availability of textual data constantly generated on the Web (in the form of news, blog articles, newsgroup posts, consumer reviews, etc.) open up many possibilities for exploring new interesting data mining problems. Most existing research on text stream mining has focused on mining single text streams (e.g., [9, 18]). Web data, however, is naturally generated by a large number of streams, and hence there exist many unique Web-specific problems, which require taking into account complex interactions and dependencies in the behavior of multiple streams.

One such problem is, given a collection of multilingual textual streams in the form of the on-line news documents provided by the RSS feeds of news agencies in different countries, discover named entities (such as names of people, organizations or geographic locations), which exhibit similar pattern in the form of temporally correlated bursts of their mentions in the documents produced by the streams. Such patterns are often very meaningful. We define the burst of an entity as a sudden and sharp increase in the total number of its occurrences in the documents generated by all the Web news streams in the same language. Temporally correlated bursts are simultaneous bursts of (potentially different) terms in different text streams which consistently occur at similar points or intervals of time.

In general, "bursty" terms in text streams are interesting to study, since they often signal changes in some latent variable. For example, a sudden increase of mentions of a particular named entity in the news streams is generally correlated with the happening of a particular event. Specifically, when a major international event happens, news streams tend to frequently mention certain named entities (e.g., people, locations, and organizations) associated with it, leading to temporally correlated bursts of related entities. If we can discover named entities with correlated bursts from textual streams in different natural languages, we will be able to group semantically related named entities in different languages together and potentially discover

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
WSDM'11, February 9–12, 2011, Hong Kong, China.
Copyright 2011 ACM 978-1-4503-0493-1/11/02 ...$10.00.

transliteration relations of proper names in an unsupervised way. Since proper names grow in an open-ended way, it is hard to manually create an exhaustive list of transliterations for all possible proper names. Therefore, methods to automatically mine such transliterations can be very useful, particularly for cross-language information retrieval and machine translation. In addition to that, the vastness of the Web data ensures the coverage of transliterations in all possible domains, including the new and emerging ones, that no dictionary can guarantee. Therefore, terms with temporally correlated bursts of mention counts not only constitute valuable knowledge by themselves, but can also be used to identify latent events that cause the change in the behavior of multiple text streams.

Moreover, entities with temporally correlated bursts can also reveal how particular events are represented in different languages (or countries). For example, by knowing the entities that occur in the news wires of different countries much more frequently than usual during a particular time point or interval, one can not only discover the major events happening at that moment, but also differentiate between the local events, which are important only to one country, and global events, which attract the interest of the media from different countries. In addition to that, the social impact of a real life event can be estimated by the strength and duration of the bursts of certain named entities that it causes. As a specific example, the nomination of Sarah Palin as a republican vice-presidential candidate during the 2008 presidential election campaign has caused intense and long-term bursts of mention counts of the term “Palin” in the U.S. news streams, but in other countries this event was only briefly mentioned in the news wires.

While mining “bursty” terms in a single text stream has been studied previously (e.g., [9]), mining terms with correlated bursts from multiple Web text streams raises three interesting new challenges:

1. **Difference in burst magnitude:** correlated entities in multiple multilingual streams may have bursts that are of different orders of magnitude in their intensity in streams, corresponding to different languages. For example, the bursts of terms related to a major U.S. event (e.g., “Katrina”) are likely to be several orders of magnitude higher in the U.S. news media than the bursts of the corresponding entities in, for example, the Russian media. Thus, using raw mention counts of terms will likely produce inaccurate results.

2. **Temporal lag:** it is often the case that there exists a *temporal lag between the time points*, at which the on-line media in different countries start covering the same event. Therefore, our method cannot assume the temporal alignment of bursts for the same entities in different languages and needs to account for the fact that there may be irregular time gaps between the bursts, corresponding to the related entities in multilingual news streams.

3. **Entities are much more fine-grained units, than documents or topics:** topics are (possibly unbounded) sets of terms with associated probabilities. Major events may cause long-term correlated bursts of a large number of entities, whereas bursts corresponding to minor events may last for a very short time and involve only a few entities. For this reason, although topic-based event detection methods can discover a small number of topics reflecting major, long-term and very influential events, they are likely to miss many minor and short-term correlated bursts of individual terms.

In this work, we propose a two-stage method for mining items with temporally correlated bursts from multiple data streams. In the first stage of the method, the temporal behavior of individual entities in news streams is normalized by modeling them with the Markov-Modulated Poisson Process (MMPP). The MMPP provides a necessary level of abstraction over the raw stream data, which ensures robustness of the approach against the differences in the magnitude of bursts of individual entities, thus addressing the first issue outlined above. In the second stage, we propose to use a dynamic programming algorithm to discover entities with correlated bursts that can be potentially temporally separated by irregular gaps, thus addressing the second issue.

In summary, the main contributions of this work are as follows:

- We formulated a novel multi-stream text mining problem of detecting named entities with correlated bursts in on-line news streams. To the best of our knowledge, the present work is the first attempt at solving this problem;

- We proposed a two-stage solution to the formulated problem, which combines a theoretically justified probabilistic modeling method with a dynamic programming algorithm to address the unique challenges of the proposed problem, which are specific to the Web. Although in this paper we focus on the textual domain, the proposed method is essentially general and data independent. Thus, it can also be potentially applied to mining items with temporally correlated bursts in any type of data streams;

- To the best of our knowledge, this is the first work that proposed to formalize the problem of burst detection as modeling the stream behavior with the discrete-time Markov Modulated Poisson Process;

- We empirically demonstrated that the proposed method can effectively discover named entities, corresponding to major and minor real life events from multiple real multilingual Web news streams. In addition to that, we presented experimental results, indicating that our method has comparable performance to the two state-of-the-art methods for automatic detection of transliterations in parallel static corpora, and, thus, can be applied to solving this important practical problem as well.

The rest of the paper is organized as follows. In the next Section, we briefly discuss the major lines of research related to the present work. In Section 3, we formally define the problem of mining named entities with correlated bursts from multiple text streams, followed by a high-level overview of our two-stage method in Section 4. We discuss how MMPP can be used for modeling the behavior of data streams and present an EM algorithm for estimating its parameters in Section 5. A dynamic programming algorithm to discover entities with temporally correlated bursts is discussed in detail in Section 6. We present and discuss the evaluation results in Section 7 and conclude with the summary and directions for future work.

2. RELATED WORK

In this section, we provide a brief overview of three major research lines, related to different aspects of the present work: burst detection, multi-stream text mining and automatic transliteration. Burst detection is an important problem in stream data management. Previous research has clearly demonstrated that different problems, involving streams of various types and data volumes, require different approaches to burst detection. Kleinberg [9] proposed an infinite-state automaton to model complex hierarchical structure of nested bursts in a stream of emails. In topic detection and tracking (TDT), Swan et al. [18] proposed to use the χ^2-test and Krause et al. [11] a combination of Factorial HMMs and exponential order statistics to identify periods of topical bursts in various static text collections. Zhu et al. [23] used sliding window smoothing and wavelet transformations for detecting bursts in large-scale astrophysical and stock trading data streams. Parikh et al. [14] proposed a method for detecting and classifying bursts in users queries to a large scale e-commerce system. Although smoothing-based burst detection methods can be used for large volume data streams, they are not suitable for precise alignment of bursts across multiple data streams, due to potential distortion of shape, duration and magnitude of bursts after smoothing. In addition to that, distance-based measures for similarity of bursts generally have a common disadvantage in that they require data-dependent tuning of threshold parameters.

In *general multi-stream mining*, several methods [16] [22] [8] have been proposed to detect correlations between entire data streams. Most of the proposed methods, however, relied on geometric distance measures on the raw stream data and disregarded potentially useful characteristic features of stream behavior, such as bursts, thus trading accuracy for efficiency and applicability to large-volume streams. Ide at al. [8] introduced singular spectrum transformations to detect correlations between time series based on change-point scores. Streams of mention counts of particular named entities in news wires, however, are not high-volume streams and the task of aligning them favors precision over efficiency. A few existing works on *mining multiple textual streams* adopted a document-level view of streams and proposed extensions to existing probabilistic methods for topic modeling, such as PLSA in Wang et al. [20] and LDA in Wang et al. [21] and Blei et al. [2], to detect the common topics, shared across multiple textual streams. These methods can detect only 10-15 major topics, shared across the entire textual streams. Although it has been demonstrated that topic modeling-based methods can potentially capture some major long-term events and the terms, describing those events, they are likely to skip many short-term events and an even larger number of entities and terms, corresponding to those events. Moreover, since topics define distribution over a large number of general terms and can be quite vague, it is hard to accurately and automatically align them, without explicit semantic labeling. Even if such alignment is performed, there still remains a problem of extracting related pairs of entities from the aligned topics. Therefore, topic-based approaches are not applicable to the fine-grained, low-level task of alignment of individual named entities.

Machine transliteration is the process of matching words in the source language with their approximate phonetic or spelling equivalents in the target language. Existing approaches to automatic transliteration mostly require linguistic knowledge to construct phonetic similarity models for particular pairs of languages such as English-Arabic [1], English-Chinese [12] or English-Japanese [10] and usually require supervision. Unsupervised methods have the advantage that they can work with any pair of languages and require less effort to implement. Previous work on *unsupervised* automatic transliteration includes approaches, that involve computing simple distance measures, such as the Pearson correlation coefficient [19] over the raw streams of entity mention counts, or the normalized cosine similarity [17] over the raw streams smoothed by a sliding window. The Pearson correlation coefficient was also used in [3] to find semantically related queries. Since these methods disregard other strong similarity signals and rely only on general correlation measures between the time series of mentions counts, one can envision that as the number of entities increases with the amount of data generated by textual streams, there may be a fairly large number of entities, whose entire time series of mention counts are correlated, according to simple distance measures. It has been experimentally demonstrated that these methods can discover related entities in parallel static corpora of small size. However, it is interesting to know whether these methods can be accurate in detecting transliterations in the Web news streams as well. Although our method is designed to address the specific challenges of Web textual streams, it is interesting to compare its performance relative to other methods for similar tasks, such as automatic transliteration, which we do in Section 7, using both of the above-mentioned methods as baselines.

As follows from the above discussion, our proposed approach for detection of entities with correlated bursts can also provide a novel solution to another important interdisciplinary problem. Although evaluation of the proposed method focused on discovering correlated bursts of named entities, we would like to note that our method is data independent in nature and can be used to detect items with correlated bursts in any type of data streams. We now move on to a high-level overview of the proposed method.

3. PROBLEM DEFINITION

Informally, the problem addressed in the present work can be formulated as follows. Given a collection of text streams as input, identify all pairs of named entities in different languages that have temporally correlated bursts. Before moving on to the high-level overview and more in-depth discussion of our approach, we need to define the key concepts behind it.

DEFINITION 1. (TEXT STREAM) *A textual data stream $\mathcal{S}$ of length M is a temporally ordered sequence of documents $\{D_1, D_2, \ldots, D_M\}$ over T discrete, non-overlapping time intervals $1, 2, \ldots, T$, such that each document in the sequence has an associated time stamp.*

Since each document in a sequence has a time stamp, documents can be grouped into sets, in such a way that each document in a set belongs to a certain time interval $(t-1, t]$, $1 \leq t \leq T$. According to the above definition, documents produced by news agencies naturally form temporarily ordered textual streams.

Definition 2. *(*Stream of Entity Mention Counts*) A stream of mention counts $\mathcal{C}$ for a named entity E is a temporarily ordered numerical sequence $\{c_1, c_2, \ldots, c_T\}$ of the number of times a named entity E occurs in the documents of the text stream S within each of the $T-1$ time intervals.*

Definition 3. *(*Burst*) Given a named entity E and its stream of mention counts $\mathcal{C}$, if $\exists t_1, t_2 \in [1, T]$, such that $t_2 - t_1 = \tau$ and $\forall t \in [t_1, t_2]$, $c_t \geq \sigma$, then an entity E has a burst of duration $t_2 - t_1$.*

Bursts correspond to one or several adjacent time intervals in a stream of entity mention counts, in which the counts are greater or equal than the threshold σ. An entity and its corresponding stream of mention counts may have several bursts. Next we extend the definition of a burst to the case of multiple streams.

Definition 4. *(*Correlated Burst*) Let E_1 and E_2 be a pair of entities that have bursts at the intervals $[t_{11}, t_{12}]$ and $[t_{21}, t_{22}]$ in the streams of their mention counts $\mathcal{C}_1$ and $\mathcal{C}_2$ respectively. If the boundaries of $[t_{11}, t_{12}]$ and $[t_{21}, t_{22}]$ are within the τ time intervals, then the named entities E_1 and E_2 have a correlated burst.*

In other words, correlated bursts occur when the mention counts of two entities are both above the threshold during the temporally close time intervals. We now move on to an overview of our method.

4. TWO-STAGE METHOD

As discussed in the introduction, existing methods for discovering correlated bursts cannot address the two potential variations of bursts across the different streams: (i) bursts may be of different orders of magnitude (ii) bursts may be separated by time gaps. These two limitations are illustrated by an example in Figure 1. In this example, the two entities have clearly correlated bursts in their streams of mention counts, despite the relative difference in the magnitude and the time lag between them.

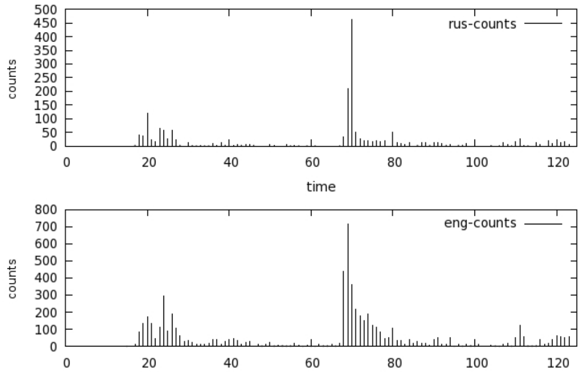

Figure 1: Example of the two streams, corresponding to mention counts of the same entity in Russian and English streams. The temporal behavior with respect to bursts is clearly correlated, despite the relative difference in the magnitude of bursts and the time gap between them.

Our method consists of the two stages. In the first stage, we assume that the temporal behavior of named entities in textual data streams can be characterized by a discrete stochastic process, whose latent parameters can be estimated in a well-defined and precise way. As such probabilistic model, we propose to use the Markov Modulated Poisson Process (MMPP) [6]. MMPP is formally defined as a doubly stochastic Poisson process [7], whose intensity is time-varying according to a finite, non-observable Markov chain. Since in the context of our problem the time intervals are strictly bounded, we are using MMPP with discrete-time Markov chain. In general, a discrete-time MMPP is a Hidden Markov Model (HMM) [5, 13, 15], in which the Poisson distribution is used as an emission distribution. Given an observation sequence, HMM can explain it in terms of unobserved sequence of model state changes and probability density functions associated with model states. In case of MMPP, each state is associated with a Poisson process, generating mention counts according to the Poisson rate parameter. MMPP starts from a certain state according to the initial state probability distribution and undergoes state changes over time, according to the matrix of state transition probabilities. Fitting MMPP for a given sequence of observations involves estimating the matrix of transition probabilities for the Markov chain and the intensity parameters of the Poisson processes, associated with the states in the model.

Thus, in the first stage the streams of entity mention counts $\mathcal{C} = \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_K\}$ are extracted for any number K of named entities discovered in the collection of multiple raw textual streams $\mathcal{S} = \{\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_M\}$ of length T. Then the parameters $\mathbf{M}_1, \mathbf{M}_2, \ldots, \mathbf{M}_K$ of MMPPs, corresponding to each stream of mention counts in $\mathcal{C}$, are estimated by using an EM algorithm, described in Section 5.3, after which each stream of entity mention counts $\mathcal{C}_i$ in $\mathcal{C}$ is mapped into a stream of temporal behavior (or "burstiness") coefficients Φ_i. As can be seen in Figure 1, the bursts in the first and second streams are of different orders of magnitude, therefore using non-normalized distance measures may indicate low correlation of the streams. Mapping the raw mention counts to the "burstiness" coefficients allows to abstract away from the raw data and achieve uniform normalization.

In the second stage, we run a dynamic programming alignment algorithm, described in Section 6, on the space of all pairs of streams of temporal behavior coefficients $\Phi \times \Phi$ to detect the pairs of streams with temporally correlated bursts.

5. MODELING BURSTS WITH MMPP

5.1 General Idea

Given $\mathcal{S} = \{\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_P\}$, a collection of P text streams over T discrete time intervals, our method first creates a set of K streams of entity mention counts $\mathcal{C} = \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_K\}$ over the same number of T time intervals by extracting the number of mentions of each named entity during each time interval from all streams in $\mathcal{S}$. At the next step, each stream of entity mention counts $\mathcal{C}_i$ is modeled with the Markov Modulated Poisson Process $\mathbf{M}_i$. In particular, the behavior of each named entity $E_i \in V$ at each time interval $(t-1, t]$, $1 \leq t \leq T$ of $\mathcal{C}_i$ is characterized by the value of the expectation λ of the Poisson distribution, associated with the state that the MMPP is in at that time interval (i.e., λ is the expectation of the number of mention counts of the entity during the corresponding time interval). The vector of expectations of the Poisson distribution can be sorted in ascending order and each time interval can be labeled with the rank of λ corresponding to it, instead of the λ itself. Therefore, ϕ_{it}, the rank of λ, can be viewed as an entity "burstiness" coefficient, with larger values of ϕ_{it} corresponding to the states with larger values of expectation for the number of entity mentions (i.e. more "bursty" states). It follows that there exists a unique and natural mapping from a stream of entity mention counts $\mathcal{C}$ into a stream of "burstiness" coefficients $\phi = \{\phi_1 = r_{\lambda_1}, \phi_2 = r_{\lambda_2}, \ldots, \phi_T = r_{\lambda_T}\}$, where $r_{\lambda_1}, r_{\lambda_2}, \ldots, r_{\lambda_T}$ are the ranks of $\lambda_1, \lambda_2, \ldots, \lambda_T$ in the sorted vector of expectations of the Poisson distribution associated with each state of MMPP. In other words, given a fully specified MMPP and a sequence of observed mention counts for a named entity over a set of discrete time intervals, we can model the observation sequence by labeling each interval with the state that the hidden Markov chain of MMPP is in at that interval, in such a way that states with larger numbers, correspond to more "bursty" states. An example of such labeling is shown in Figure 2.

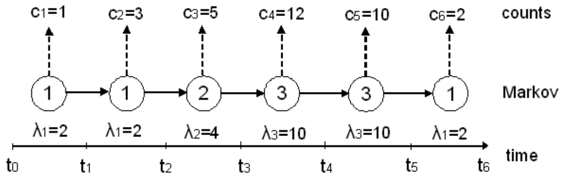

Figure 2: Example of a stream, in which the time intervals are labeled by its activity level, with larger numbers corresponding to more "bursty" states.

5.2 Formal Definition

We now define the MMPP more formally. With MMPP, each stream is modeled as a finite-state, discrete-time Markov chain with N distinct states $S = \{S_1, S_2, \ldots, S_N\}$, which at any given time point t can be in one of the N states. The distribution of observations (mention counts) at time t is determined by the intensity parameter of the Poisson distribution, which is associated with the state S_t, that the Markov chain is in at time t. At the end of each time stamp $t = 1, 2, \ldots, T$, the Markov chain undergoes a state change, from state i to state j (where j can be the same as i) with the probability $\mathbf{A}_{ij}$, according to an $N \times N$ matrix $\mathbf{A} = \{a_{i,j}\}$, $i = 1 \leq i, j \leq N$, of state transition probabilities. Let $Q = (q_1, q_2, \ldots, q_t, \ldots, q_T)$, where $q_i \in \{S_1, \ldots, S_N\}$ for $i = 1, 2, \ldots, T$ be an unobservable random vector, whose elements follow an N-state first-order Markov chain over the state space S with unknown matrix of state transition probabilities $\mathbf{A}$ and unknown distribution of initial state probabilities $\pi = \{\pi_i, \ldots, \pi_N\}$. The number of mention counts c_t of any given entity during the time interval $(t-1, t]$ can be characterized by the following Poisson probability distribution, depending on the hidden state q_t that the Markov chain is in at time t:

$$Pr(c_t|q_t) = \frac{e^{-\lambda_{q_t}} \lambda_{q_t}^{c_t}}{c_t!}$$

where λ_{q_t} is the expectation of the number of mention counts of an entity at time t, given that MMPP is in state q_t. Therefore, in order to model temporal behavior of any named entity with respect to its "burstiness" at any given time point, we need to estimate the following triplet of parameters of an N-state ergodic MMPP $\mathbf{M} = (\mathbf{A}, \pi, \lambda)$, where $\lambda = \{\lambda_1, \lambda_2, \ldots, \lambda_N\}$ is the vector of Poisson rates, associated with each state of the Markov chain.

Given a stream of observations $\mathcal{C}$, we need a computationally efficient procedure to estimate the parameters $\mathbf{M} = (\mathbf{A}, \pi, \lambda)$ of MMPP and determine the optimal state sequence of the unobservable Markov chain Q, corresponding to $\mathcal{C}$. This can be viewed as a process of finding a model $\mathbf{M}$ in a space of all possible models, such that it maximizes the probability of observing $\mathcal{C}$:

$$\mathbf{M} = \underset{\mathbf{A},\pi,\lambda}{\operatorname{argmax}}\ P(\mathcal{C}|\mathbf{A}, \pi, \lambda)$$

In principle, one can compute $P(\mathcal{C}|\mathbf{A}, \pi, \lambda)$ by computing the joint probability $P(\mathcal{C}, q_1, q_2, \ldots, q_T|\mathbf{A}, \pi, \lambda)$ for all possible hidden state sequences of length T $q_1, q_2, \ldots, q_T$, and marginalize over all state sequences:

$$P(\mathcal{C}|\mathbf{A}, \pi, \lambda) = \sum_{q_1=S_1}^{S_N} \cdots \sum_{q_T=S_1}^{S_N} P(\mathcal{C}, q_1, q_2, \ldots, q_T|\mathbf{A}, \pi, \lambda) \tag{1}$$

where

$$P(\mathcal{C}, q_1, q_2, \ldots, q_T|\mathbf{A}, \pi, \lambda) = \pi_{q_1} \prod_{t=2}^{T} \left(\frac{e^{-\lambda_{q_t}} \lambda_{q_t}^{c_t}}{c_t!} \mathbf{A}_{q_{t-1},q_t} \right)$$

Since the amount of computation using the above formulas quickly becomes intractable, as the length of observation sequences grows, we use an EM algorithm [4], described in the next section, to maximize (1).

5.3 EM algorithm

For the purpose of clarity, we define two conditional probabilities, given the observation sequence $\mathcal{C}$ and $\mathbf{M}$, the parameters of MMPP:

- $\xi_t(i,j)$, the conditional probability of the Markov chain being in state S_i at time t and in state S_j at time $t+1$:

$$\xi_t(i,j) = P(q_t = S_i, q_{t+1} = S_j|\mathcal{C}, \mathbf{M})$$

- $\gamma_t(i)$, the conditional probability of the Markov chain being in state S_i at time t, given observation sequence $\mathcal{C}$:

$$\gamma_t(i) = P(q_t = S_i|\mathcal{C}, \mathbf{M})$$

From the above definitions, it follows that the expected number of transitions from the state S_i is:

$$E[\gamma(i)] = \sum_{t=1}^{T} \gamma_t(i) \tag{2}$$

and the expected number of transitions from the state S_i to the state S_j is:

$$E[\xi(i,j)] = \sum_{t=1}^{T-1} \xi_t(i,j) \tag{3}$$

At the M-step of the EM algorithm we maximize

$$E[P(\mathcal{C}, q_1, q_2, \ldots, q_T|\mathbf{A}, \pi, \lambda)|\mathcal{C}] \tag{4}$$

We denote the vector of model parameter estimates that maximize (4) at the p-th iteration of the EM algorithm as $\hat{\mathbf{M}}^p$. At the E-step, we compute $\gamma^{p+1}(i) = E(\gamma(i)|\hat{\mathbf{M}}^p)$ and $\xi^{p+1}(i,j) = E(\xi(i,j)|\hat{\mathbf{M}}^p)$ for $1 \leq i,j \leq N$, using the estimate $\hat{\mathbf{M}}^p$ from the previous iteration of the algorithm. Expressions for the expectations of the number of times the Markov chain is in state i and the number of transitions from state i to state j for the E-step of an EM algorithm are shown in Figure 3.

Closed form expressions for the estimates of model parameters $\hat{\mathbf{M}}^p = (\hat{\mathbf{A}}^p, \hat{\pi}^p, \hat{\lambda}^p)$ at the M-step are as follows. Poisson rate parameters $\hat{\lambda}^p$ at the M-step of the pth iteration are estimated as:

$$\hat{\lambda}_i^p = \frac{\sum_{t=1}^{T} \hat{\gamma}_t^p(i) c_t}{\hat{\gamma}^p(i)}$$

The probability of transition between state S_i and S_j is estimated as:

$$\hat{\mathbf{A}}_{i,j}^p = \frac{\hat{\xi}^p(i,j)}{\hat{\gamma}^p(i)}$$

And, finally, initial state probabilities are estimated as:

$$\hat{\pi}_i^p = \gamma_1^p(i)$$

In order to make an EM algorithm more efficient in terms of the number of calculations required to compute $\hat{\gamma}^p(i)$ and $\hat{\xi}^p(i,j)$, we can use the Forward-Backward Algorithm. The vector of "burstiness" coefficients ϕ, corresponding to a given observation sequence, can be obtained by labeling the observation sequence with the states of the Markov chain by using the Viterbi algorithm.

5.4 Discussion

MMPP is an inherently unsupervised model. It, however, requires one parameter, the number of states of the Markov chain, to be specified a priori. In this work we experimentally determine the optimal number of states of MMPP, as described in Section 7.2.

Using MMPP over statistical approaches for detection of bursts in multiple streams environment provides a uniform level of abstraction over the raw stream data. In particular, simultaneous bursts in multiple streams can be detected regardless of their magnitude relative to each other. This is achieved by labeling observations from streams with ordered states of MMPP. The complexity of labeling and training on an entire observation sequence with MMPP is $O(N^2T)$, where N is the number of states in MMPP and T is the length of the sequence.

6. DETECTING CORRELATED BURSTS

6.1 Formal definition

The algorithm for detecting correlated bursts is based on the general idea of constructing an alignment table for a pair of streams of "burstiness" coefficients, which keeps track of the alignment score, as the algorithm simultaneously processes the two streams. If both streams are in "bursty" states within the same time interval, the score is incremented by a reward. If the bursts in streams are within the maximum allowable temporal gap, the score is also increased, however the reward in this case is less. If one of the streams is in a "bursty" state and the other is not, the score is decreased by the penalty, equal to the amount of reward. Note that the algorithm allows to incrementally align the streams, as the new data is arriving. Formally, given a set of streams of temporal behavior coefficients (MMPP states) $\Phi = \{\Phi_1, \Phi_2, \ldots, \Phi_K\}$ for K named entities, where each Φ spans a period of T time intervals, we align each pair of streams $\Phi_1 = \{\phi_{1,1}, \phi_{1,2}, \ldots, \phi_{1,T}\}$ and $\Phi_2 = \{\phi_{2,1}, \phi_{2,2}, \ldots, \phi_{2,T}\}$ by constructing a dynamic programming table S of size $T \times T$, in which the element $s_{i,j}, 1 \leq i,j \leq T$ is a correlation score of the two streams at the time points i and j with respect to their "burstiness". The table is constructed according to the following formulas:

$$s_{i,j} = \max(s_{i-1,j}, s_{i,j-1}, s_{i-1,j-1}) + r_{i,j} \tag{5}$$

$$r_{i,j} = \begin{cases} 0, \text{ if } \phi_{1,i} < \sigma \text{ and } \phi_{2,j} < \sigma \\ \rho - p(i,j), \text{ if } \phi_{1,i} > \sigma \text{ and } \phi_{2,j} > \sigma \\ -\rho + p(i,j), \begin{array}{l} \text{if } \phi_{1,i} < \sigma \text{ and } \phi_{2,j} > \sigma \text{ or} \\ \text{if } \phi_{1,i} > \sigma \text{ and } \phi_{2,j} < \sigma \end{array} \end{cases} \tag{6}$$

If both streams are in "bursty" states (above the threshold σ), $r_{i,j}$ is equal to the reward constant ρ, decreased by the penalty function p. A penalty function is a function mapping the length of the time gap $|i-j|$ between the time points i and j to the value of the penalty. Its purpose is to penalize the correlation of bursts shifted in time relative to each other. A penalty function can take any algebraic form, but should be equal to 0, if there is no gap between i and j. The maximum possible size of the gap between the bursts, for which the streams are still rewarded, depends on ρ and the algebraic form of the penalty function. An exponential $p = e^{|i-j|}$ and quadratic $p = (i-j)^2$ penalty functions force to avoid larger temporal gaps between the bursts. Logarithmic function $p = \lfloor \log(|i-j|) \rfloor$, on the other hand, favors longer lags between the bursts. Linear function $p = |i-j|$ decreases the reward by 1 with each day of the gap between two bursts. For example if $\rho = 3$ and the penalty function is linear, the maximum gap would be 2 time intervals. If one of the streams is in a "bursty" state and the other one is not, the "burstiness" score is equal to the negated reward, which is decreased, depending on the gap between i and j. The final correlation score is equal to $s_{T-1,T-1}$. It can be normalized by diving it by $(2\rho - 1)(b-1) + \rho$, the maximum possible alignment score, where b is the maximum number of bursts (time stamps with MMPP states above the threshold) among the two streams. Algorithm 1 is a complete dynamic programming algorithm for determining the similarity between the streams with respect to the temporal correlation of bursts.

The running time of Algorithm 1 is $O(T^2)$. Therefore, the total running time of the two-stage approach is $O(N^2T + T^2)$, which is orders of magnitude less than the running time of topic-based approaches [20] [21].

7. EXPERIMENTS

7.1 Data Set

All the experiments in this work have been performed using the data set consisting of documents from multiple English and Russian news wires that have been crawled over 4 months from October of 2007 to February of 2008. Statistics of the experimental data set are presented in Table 1. Since the proposed method works at the level of individ-

$$
\begin{aligned}
\hat{\gamma}^{p+1}(i) &= E(\gamma(i)) = \frac{P(q_1,\ldots,q_t=S_i,\ldots,q_T|O,\hat{\mathbf{M}})}{P(q_1,q_2,\ldots,q_T|O,\hat{\mathbf{M}})} = \\
&= \frac{\sum_{q_1=S_1}^{S_N}\sum_{q_2=S_1}^{S_N}\cdots\sum_{q_{t-1}=S_1}^{S_N}\sum_{q_{t+1}=S_1}^{S_N}\cdots\sum_{q_T=S_1}^{S_N} P(O,Q=(q_1,\ldots,q_t=i,\ldots,q_T))P(Q|\hat{\mathbf{M}})}{\sum_{q_1=S_1}^{S_N}\sum_{q_2=S_1}^{S_N}\cdots\sum_{q_T=S_1}^{S_N} P(O|Q=(q_1,\ldots,q_T),\hat{\mathbf{M}})P(Q|\hat{\mathbf{M}})}
\end{aligned}
$$

$$
\begin{aligned}
\hat{\xi}^{p+1}(i,j) &= E(\xi(i,j)) = \frac{P(q_1,\ldots,q_t=S_i,q_{t+1}=S_j,\ldots,q_T|O,\hat{\mathbf{M}})}{P(q_1,\ldots,q_t=S_i,\ldots,q_T|O,\hat{\mathbf{M}})} = \\
&= \frac{\sum_{q_1=S_1}^{S_N}\cdots\sum_{q_{t-1}=S_1}^{S_N}\sum_{q_{t+2}=S_1}^{S_N}\cdots\sum_{q_T=S_1}^{S_N} P(O,Q=(q_1,\ldots,q_t=i,q_{t+1}=j,\ldots,q_T))P(Q|\hat{\mathbf{M}})}{\sum_{q_1=S_1}^{S_N}\sum_{q_2=S_1}^{S_N}\cdots\sum_{q_T=S_1}^{S_N} P(O|Q=(q_1,\ldots,q_T),\hat{\mathbf{M}})P(Q|\hat{\mathbf{M}})}
\end{aligned}
$$

$$P(O|\hat{\mathbf{M}}) = \sum_{q_1=S_1}^{S_N}\sum_{q_2=S_1}^{S_N}\cdots\sum_{q_T=S_1}^{S_N} P(O|Q=(q_1,\ldots,q_T))P(Q|\hat{\mathbf{M}})$$

$$P(Q=(q_1,\ldots,q_T)|\hat{\mathbf{M}}) = \pi_{\mathbf{q_1}}\prod_{t=2}^{T}\mathbf{A}_{q_{t-1},q_t}$$

Figure 3: EM updating formulas for the Markov Modulated Poisson Process

Algorithm 1 Algorithm for detecting streams with temporally correlated bursts

```
Require: $\phi_1, \phi_2$, streams of “burstiness” coefficients of length $T$
Require: $p$, penalty function
Require: $\sigma$, threshold for “bursty” states
Require: $\rho$, reward constant
$S \Leftarrow \mathbf{0}$
for $i = 0$ to $T - 1$ do
  for $j = 0$ to $T - 1$ do
    if $\phi_{1,i} > \sigma$ and $\phi_{2,i} > \sigma$ then
      $S[i][j] \Leftarrow \rho - p(i,j)$
    else if $\phi_{1,i} < \sigma$ and $\phi_{2,i} > \sigma$ then
      $S[i][j] \Leftarrow -\rho + p(i,j)$
    else if $\phi_{1,i} > \sigma$ and $\phi_{2,i} < \sigma$ then
      $S[i][j] \Leftarrow -\rho + p(i,j)$
    end if
    if $i > 0$ and $j > 0$ then
      $S[i][j] \Leftarrow S[i][j] + max(S[i-1][j], S[i][j-1], S[i-1][j-1])$
    else if $j > 0$ then
      $S[i][j] \Leftarrow S[i][j] + S[i][j-1]$
    else if $i > 0$ then
      $S[i][j] \Leftarrow S[i][j] + S[i-1][j]$
    end if
  end for
end for
return $S[T-1][T-1]$
```

Language	# Streams	# Docs	# Entities
english	16	53189	2048
russian	4	84610	1338

Table 1: Statistics of the data streams used for evaluation

ual entities, rather than entire documents in the stream, after crawling the data we preprocessed it by constructing a multidimensional temporal index, which allows to easily determine the number of times a given entity occurred in all the stream documents in the same language within a given period if time. We did not use named entity recognizers for detection of named entities. Instead, we extracted all capitalized phrases and split them into individual lexemes (e.g. “Barack Obama” was split into “Barack” and “Obama”). We also applied basic morphological normalization to the Russian named entities by removing inflectional endings (i.e. Обама, Обаме, Обаме, Обамой etc. were converted to the same stem “Обам”). After that, for each extracted entity we extracted a stream of the number of times it was mentioned in all the documents generated by the streams in the same language during each of the observation days. For experiments we discarded the entities that are too rare (entities, which have less than 50 mentions in total or are mentioned in less than 3 days over the 4 month observation period) and too common (entities, which occurred at least once during more than 80% of the days in the observation period). The final number of entities (and their associated mention count streams) in each language after such filtering is shown in Table 1

7.2 Parameter setting

Our method has several parameters to tune. We experimentally determined the sensitivity of performance to these parameters, the effectiveness of normalization of mention counts using MMPP and the necessity of accommodating the time gaps.

The first parameter we examine is the number of states in the MMPP. In Figure 4, we show the performance variation when changing the number of states of MMPP. In the same figure, we also compare MMPP with a binning-based baseline normalization method. The binning normalization method searches for the minimum and the maximum element in the time series and partitions the range between the minimum and the maximum into a given number of intervals (bins). Labels are then assigned to each element of the time series, depending on the interval that this element belongs to. Modeling a stream with N bins is conceptually similar to modeling it with an N-state MMPP. In order to evaluate the performance of parameter setting, from the index we randomly selected 40 pairs of entities from both languages, which denote the same object. For each of those entities, we ran our method to detect the entities with correlated bursts among the selected entities in the other language. The recall measure takes into account the presence of the correct transliteration in the top 5 candidate transliterations, ranked according to the “burstiness” correlation score computed by our method. As can be seen in Figure 4, MMPP consistently outperforms the binning-based normalization baseline, indicating that MMPP can take the advantage of global model fitting to achieve better normalization. Another important conclusion is that a 3-state MMPP is sufficient for achieving the optimal performance.

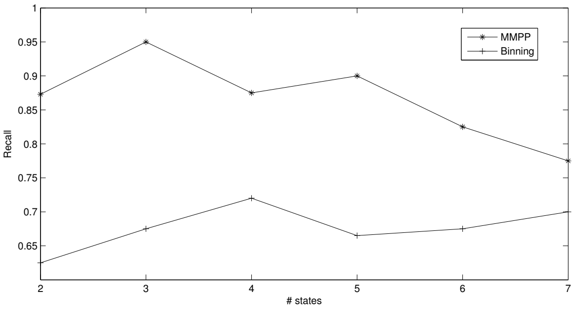

Figure 4: Performance of MMPP versus binning-based smoothing

Next, we experimented with different penalty functions and values of reward in the dynamic programming algorithm used in the second stage of the method. We varied the value of reward from 1 to 5 and used quadratic, linear, logarithmic and zero (i.e. no penalty) functions.

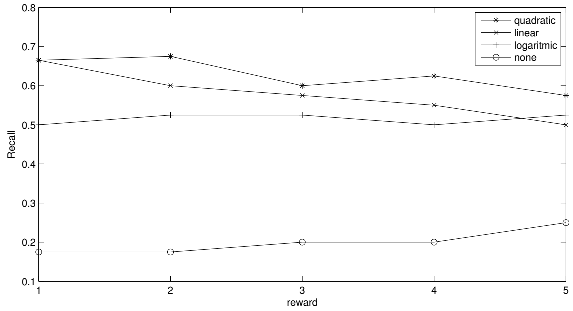

Figure 5: Performance of different penalty functions

Figure 5 demonstrates that the optimal performance is achieved by using a quadratic penalty function in conjunction with the reward value of 4, which indicates that it is necessary to accommodate the temporal gaps in order to achieve the optimal performance.

7.3 Results

In order to evaluate the proposed two-stage approach, we conducted two sets of experiments. The first set was aimed at checking the ability of our method to solve its primary task: detect entities with temporally correlated bursts in text streams. The second set of experiments was focused on estimating the performance of our method with respect to a different, but conceptually related task of detecting transliterations. For both tasks we used the optimal configuration of parameters, empirically determined in Section 7.2. In the first stage, we used a 3-state MMPP, which was run for a maximum of 500 iterations or until convergence. In the second stage, the threshold for “burstiness” was set to 3 (i.e., only state 3 is considered as “bursty”), the reward was set to 4 and quadratic penalty function was used, thus allowing the maximum possible gap between the correlated bursts to be one day.

7.4 Detection of entities with correlated bursts

In order to evaluate the performance of our method with respect to the primary task of detecting entities with correlated bursts, for each entity in the index in one language we ran our method to determine the top 10 entities with correlated bursts among all entities in the other language. Examples of interesting mined patterns are presented in Table 2. The source entities are presented in the top row and the most similar entities along with the burst correlation scores are below them. Russian named entities are presented in Cyrillic transcription along with their English equivalents.

Patterns in Table 2 have interesting interpretations. Pattern 1 corresponds to the annual world economic forum, taking place during the last week of January in Davos, Switzerland, which usually attracts politicians, such as Italian prime minister Silvio Berlusconi. In 2008 it coincided with a tragic death of an actor Heath Ledger on January 22, 2008. The second pattern corresponds to the death of a famous chess player Robert Fischer in January of 2008. Fischer spent the last part of his life in Iceland and his main chess rival back in the days was Boris Spassky. Pattern 3 corresponds to the assassination of Benazir Bhutto in Pakistan in December of 2007. Bhutto was critically wounded and rushed to Rawalpindi General Hospital. Pattern 4 corresponds to Jerome Kerviel, a French trader, who caused considerable financial loss to the bank Societe Generale. This story coincided with the death of George Habash. As can be seen from this example, our method can mine meaningful and interesting patterns from the news streams.

7.5 Transliteration

In addition to evaluating the performance of our method with respect to the primary task of detecting entities with correlated bursts, we also compared its performance to other methods, solving the different but conceptually close task of automatically detecting transliterations. For this task we randomly selected 200 entities, in such a way that they represent the three regions of low, medium and high entropy of their associated streams of mention counts. The entropy $H(\mathcal{C}_i)$ of a stream of mention counts $\mathcal{C}_i = \{c_{i,1}, c_{i,2}, \dots, c_{i,T}\}$ is defined as:

$$H(\mathcal{C}_i) = -\sum_{t=1}^{T} c_{i,t}/N \log(c_{i,t}/N)$$

where $N = \sum_{t=1}^{T} c_{i,t}$. The entropy of a stream of length T ranges from 0 to $\log T$ and can be a good indicator of the nature of a stream. Streams with low entropy are likely to be sparse and exhibit highly non-regular behavior with possibly multiple bursts. High-entropy streams are dense and less “bursty”. Finally, for each entity in one language we

pattern 1	pattern 2	pattern 3	pattern 4
давос (davos)	спасск (spassk)	мушараф (musharraf)	kerviel
switzerland 0.72	spassky 0.5455	musharraf 0.7508	кервьель (kerviel) 0.8803
berlusconi 0.6667	fischer 0.52	pakistan 0.6199	хабаш (khabash) 0.7217
forum 0.6250	soldiers 0.367	rawalpindi 0.5006	сосьете (societe) 0.5652
ledger 0.4615	iceland 0.2051	bhutto 0.4771	женераль (generale) 0.5652

Table 2: Examples of mined entities with correlated bursts

used our method to identify and rank the top 20 most similar entities in the other language, according to the score for the temporal correlation of bursts. We performed the same procedure with the two state-of-the-art methods for mining transliterations (referred to as PC [19] and CS [17], which for each selected entity determined the top 20 ranked candidate transliterations as well. We then evaluated the accuracy of our method and the baselines with respect to the task of finding transliterations. The accuracy of transliteration is measured by the Mean Reciprocal Rank (MRR) measure, which takes into account the rank of the correct transliteration in the list of candidate transliterations. For a set of n named entities $E_1, E_2, \ldots, E_n$ and their transliteration candidates, MRR is defined according to the rank of the correct transliteration r_i, $i = 1, \ldots, n$ in the list of candidates:

$$MRR = 1/n \sum_{i=1}^{n} 1/r_i$$

The upper bound of MRR is 1, which corresponds to the case when for each named entity its correct transliteration is consistently top-ranked in the candidate list. Comparison of the performance of baselines and our method for the transliteration task is presented in Table 3.

In order to determine its relative influence on the overall performance of the method, we evaluated the first stage of the method (no temporal gaps allowed) separately (column 5 in Table 3) and in combination with the second stage (column 2 in Table 3).

Several important observations can be made based on the analysis of Table 3.

1. A combination of MMPP and Dynamic Programming has better performance than using MMPP without allowing any temporal gaps, which indicates the importance of allowing temporal flexibility in detecting correlated bursts of the Web textual streams;

2. Our method (MMPP+DP) outperforms both baselines in case of low-entropy (more "bursty") entities. However, the performance of our method decreases in other entropy categories. In general, our method performs better than CS in all entropy categories and better than PC for low-entropy entities. It can be expected that PC performs better for transliteration of less "bursty" entities, since it focuses on the overall correlation as a similarity feature, whereas our method uses bursts as primary signals;

3. Overall, for the task of finding transliterations, our method clearly outperforms one baseline and has comparable performance to the other baseline.

Thus, the experimental results support our intuition that for low-entropy entities the temporal correlation of bursts is a more effective signal for discovering transliterations than the overall correlation coefficient. Therefore, for the task of unsupervised detection of transliteration pairs, depending on the entropy of the stream of entity mention counts, our method can be used in conjunction with the Pearson correlation coefficient to achieve the best overall performance.

8. CONCLUSIONS AND FUTURE WORK

In this work, we proposed a new two-stage method for mining named entities with temporally correlated bursts from Web news streams, which can solve two unique, Web-specific challenges of this new text mining problem, i.e., the difference in magnitude of bursts and possible temporal lag between them. In the first stage of the method, the temporal behavior of different terms is normalized by modeling them with the Markov-Modulated Poisson Process, thus addressing the first challenge. In the second stage, we propose a dynamic programming algorithm to discover correlated bursts of different terms that can be potentially separated by time gaps, thus addressing the second challenge. We evaluated our method with the task of discovering transliterations of named entities from the Web news streams in different natural languages. Experimental results indicate that the proposed method can effectively discover named entities with correlated bursts that are associated with meaningful real-world events and outperforms two state-of-the-art baseline methods for mining transliterations of "bursty" named entities.

Although in this work we focused on the study of mining temporally correlated bursts of named entities in textual streams, we would like to emphasize that the proposed method is in fact data independent and general. Hence it can be potentially applied to the detection of temporally correlated bursts in any type of data streams. We believe that a very interesting direction for extending the present work would be to explore temporal correlations between the "bursty" entities mined from different types of Web 2.0 data, such as news and blogs or news and tag data.

9. ACKNOWLEDGEMENTS

This work is supported in part by DOI under the contract number NBCHC040176 and by the National Science Foundation under grants IIS-0713581 and IIS-0713571. We would like to thank the anonymous reviewers for their helpful comments, that helped to improve this work.

10. REFERENCES

[1] Y. Al-Onaizan and K. Knight. Machine transliteration of names in arabic text. In Proceedings of the ACL'02 Workshop on Computational Approaches to Semitic Languages, pages 1–13, 2002.

[2] D. Blei and J. Lafferty. Correlated topic models. Advances in Neural Information Processing Systems (NIPS), 18:147–154, 2005.

Direction	MMPP+DP		PC		CS		MMPP	
	Entropy	MRR	Entropy	MRR	Entropy	MRR	Entropy	MRR
rus→eng	all	0.3572	all	0.3497	all	0.2905	all	0.2762
	low	0.3694	low	0.3070	low	0.2843	low	0.3031
	med	0.3427	med	0.3438	med	0.3065	med	0.2851
	high	0.3084	high	0.3617	high	0.2809	high	0.2456
eng→rus	all	0.3782	all	0.3958	all	0.2838	all	0.3069
	low	0.3846	low	0.3540	low	0.2553	low	0.3283
	med	0.3375	med	0.3429	med	0.3763	med	0.2797
	high	0.3192	high	0.4061	high	0.2270	high	0.2418

Table 3: Comparison of the performance of our method with the baselines for transliteration task

[3] S. Chien and N. Immorlica. Semantic similarity between search engine queries using temporal correlation. In Proceedings of the 14th International Conference on World Wide Web, pages 2–11, 2005.

[4] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the em algorithm. Journal of the Royal Statistical Society, Series B (Methodological), 39(1):1–38, 1977.

[5] Y. Ephraim and N. Merhav. Hidden markov processes. IEEE Transactions on Information Theory, 48(6), 2002.

[6] W. Fischer and K. Meier-Hellstern. The markov-modulated poisson process cookbook. Performance Evaluation, 18(2):149–171, 1993.

[7] R. G. Gallager. Discrete Stochastic Processes. Springer, 1995.

[8] T. Idé and K. Inoue. Knowledge discovery from heterogeneous dynamic systems using change-point correlations. In Proceedings of 2005 SIAM International Conference on Data Mining (SDM'05), 2005.

[9] J. Kleinberg. Bursty and hierarchical structure in streams. In Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'02), pages 91–101, 2002.

[10] K. Knight and J. Graehl. Machine transliteration. Computational Linguistics, 24(4):599–612, 1998.

[11] A. Krause, J. Leskovec, and C. Guestrin. Data association for topic intensity tracking. In Proceedings of the 23rd International Conference on Machine Learning (ICDM'06), pages 497–504, 2006.

[12] J.-S. Kuo, H. Li, and Y.-K. Yang. A phonetic similarity model for automatic extraction of transliteration pairs. ACM Transactions on Asian Language Information Processing, 6(2), 2007.

[13] I. L. MacDonald and W. Zucchini. Hidden Markov and Other Models for Discrete-valued Time Series. Chapman and Hall, 1997.

[14] N. Parikh and N. Sundaresan. Scalable and near real-time burst detection from ecommerce queries. In Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'08), pages 972–980, 2008.

[15] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. Proceedings of the IEEE, 77(2):257–286, 1989.

[16] M. Sayal. Detecting time correlations in time-series data streams. Technical Report HPL-2004-103, HP Laboratories Palo Alto, 2004.

[17] Y. Shinyama and S. Sekine. Named entity discovery using comparable news articles. In Proceedings of the 20th International Conference on Computational Linguistics (COLING'04), 2004.

[18] R. Swan and J. Allan. Automatic generation of overview timelines. In Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'00), pages 49–56, 2000.

[19] T. Tao and C. Zhai. Mining comparable bilingual text corpora for cross-language information integration. In Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'05), pages 691–696, 2005.

[20] X. Wang, C. Zhai, X. Hu, and R. Sproat. Mining correlated bursty topic patterns from coordinated text streams. In Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'07), pages 784–793, 2007.

[21] X. Wang, K. Zhang, X. Jin, and D. Shen. Mining common topics from multiple asynchronous text streams. In Proceedings of the 2nd ACM International Conference on Web Search and Data Mining (WSDM'09), pages 192–201, 2009.

[22] T. Zhang, D. Yue, Y. Gu, and G. Yu. Boolean representation based data-adaptive correlation analysis over time series streams. In Proceedings of the 16th International Conference on Information and Knowledge Management (CIKM'07), pages 203–212, 2007.

[23] Y. Zhu and D. Shasha. Efficient elastic burst detection in data streams. In Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'03), pages 336–345, 2003.