Feature Constrained Multi-Task Learning Models for Spatiotemporal Event Forecasting

Liang Zhao, Qian Sun, Jieping Ye, Feng Chen, Chang-Tien Lu, and Naren Ramakrishnan

Abstract—Spatial event forecasting from social media is potentially extremely useful but suffers from critical challenges, such as the dynamic patterns of features (keywords) and geographic heterogeneity (e.g., spatial correlations, imbalanced samples, and different populations in different locations). Most existing approaches (e.g., LASSO regression, dynamic query expansion, and burst detection) address some, but not all, of these challenges. Here we propose a novel multi-task learning framework that aims to concurrently address all the challenges involved. Specifically, given a collection of locations (e.g., cities), forecasting models are built for all the locations simultaneously by extracting and utilizing appropriate shared information that effectively increases the sample size for each location, thus improving the forecasting performance. The new model combines both static features derived from a predefined vocabulary by domain experts and dynamic features generated from dynamic query expansion in a multi-task feature learning framework. Different strategies to balance homogeneity and diversity between static and dynamic terms are also investigated. And efficient algorithms based on Iterative Group Hard Thresholding are developed to achieve efficient and effective model training and prediction. Extensive experimental evaluations on Twitter data from civil unrest and influenza outbreak datasets demonstrate the effectiveness and efficiency of our proposed approach.

Index Terms—Event forecasting; Multi-task learning; LASSO; Dynamic query expansion; Hard thresholding

1 INTRODUCTION

Microblogs such as Twitter and Weibo are experiencing an explosive level of growth. Millions of microblog users across the world broadcast their daily observations on an enormous variety of topics, such as crime, sports, and politics. This paper focuses on the problem of spatial event forecasting from microblogs, for events such as civil unrest, disease outbreaks, and crime hotspots. Our new approach searches for subtle patterns in specific cities that serve as indicators of ongoing or future events, where each pattern is defined as a burst of context features (keywords) relevant to a specific event. For instance, expressions of discontent about gas price increases could be a potential precursor to a protest about government policies.

Three technical challenges must be overcome when addressing this problem: 1) **Dynamic features**. The language used in microblogs is highly informal, ungrammatical, and dynamic. Most existing methods treat fixed keywords as features [23], [24], but expressions in tweets may dynamically evolve, rendering the use of fixed features and historical training data insufficient. For example, the most significant Twitter keyword for the Mexican protests in Aug 2012 was "#YoSoy132" (i.e., the hashtag of an organization protesting against electoral fraud), alluding to the protests against the Mexican presidential election, but "#CNTE" (i.e., a hashtag denoting the national teacher's association of Mexico) had become the most popular term by the beginning of 2013 due to the growing resistance to Mexican education reform. Ideally, an event forecasting system must combine the judicious use of static (fixed) features with an awareness of subtle changes involving dynamic features. 2) **Geographic heterogeneity.** Existing models usually build a single predictive model for all the different locations [24], [28]. However, different cities have different characteristics, such as population, weather (e.g., humidity, temperature), and administrative structures (e.g., capital cities versus non-capital cities). As a result, it is difficult to impute basal levels of occurrence uniformly. Considering civil unrest as an example, finding 1000 tweets mentioning the keyword "protest" is not likely to be a strong indicator of an upcoming civil unrest event in a city with a population of a few million users but could be a strong signal in a much smaller city with a population of only 10,000. To consider the geographical heterogeneity, some works propose to establish the corresponding model for each different location separately [21]. But because each model only utilizes the data of its corresponding location, the data scarcity problem (especially for non-large locations) is a serious challenge that degrades the model performance and generalization. 3) **Scalability.** Spatiotemporal event forecasting in social media streams prefers real-time (or near real-time) framework and hence has emphasis on computation efficiency. However, the efficiency is challenged by the huge scale of the data, including (1) High-dimension features (e.g., keywords) to characterize the rich text and network information; (2) large number of time points; and (3) heterogeneity in enormous geo-locations (e.g., counties and cities). This means that even a medium-scale problem that contains 1000 keywords, 1000 dates and 1000 locations will involve at least 1 billion data points in the optimization computation. Therefore, some scalable forecasting methods are desired for this problem.

In order to concurrently address all these technical challenges, this work presents a novel computational approach in the form of a framework of multi-task learning (MTL) that combines the strengths of methods that use static features (e.g., LASSO regression [21]) and those that use dynamic features (e.g., dynamic query expansion (DQE) [32]). In

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2017.2657624, IEEE Transactions on Knowledge and Data Engineering

our previous work we have utilized these methods, individually, for event forecasting, but this paper tackles the challenges involved in unifying these contrasting approaches in a single framework. Learning multiple related tasks simultaneously effectively increases the sample size for each location (e.g., city, state), thus potentially also improving the forecasting performance, especially when the sample size for each task (i.e., location) is small. One critical issue in multi-task learning is how to define and exploit the commonality among different tasks. Intuitively, events that occur around the same time may involve similar topics, and therefore tweets from different cities may share many common keywords that are related to the event(s). We address this issue by presenting four multi-task feature learning (MTFL) formulations for event forecasting that differ in the specifics of how common features are extracted.

The main contributions of our study can be summarized as follows:

1) **Formulation of a multi-task learning framework for event forecasting.** Here, event forecasting for multiple cities/states in the same country are treated as a multi-task learning problem. In the proposed model, we build event forecasting models for different cities/states simultaneously by restricting all cities/states to select a common set of features with different weights exclusive to corresponding tasks. We explore both penalized and constrained MTL formulations, applying 4 different strategies to control the common set of features selected.
2) **Concurrent modeling of static and dynamic terms.** The existing models (LASSO and DQE) use different but complementary information: LASSO uses static terms, while DQE identifies dynamic terms. Our proposed MTL formulations make use of both types of information by integrating the strengths of LASSO (a supervised approach) into DQE (an unsupervised approach). To the best of our knowledge, there is little if any prior work that combines supervised and unsupervised approaches for event forecasting.
3) **Development of efficient algorithms.** In this paper we explore both convex and non-convex optimization formulations. For convex problems, we employ proximal methods, such as FISTA [6] that have been shown to be efficient for solving sparse and multi-task learning problems. For non-convex problems, we apply the iterative Group Hard Thresholding (IGHT) [8] framework, which is guaranteed to converge to a local solution.
4) **Comprehensive experiments to validate the effectiveness and efficiency of the proposed techniques.** We evaluated the proposed methods using two different Twitter datasets: the Latin America civil unrest dataset and the United States influenza outbreaks dataset. For comparison, we implemented a broad range of other algorithms. The results demonstrated that the proposed methods consistently outperformed the competing methods, namely LASSO, DQE, traditional multitask learning models, and their variants. We also performed sensitivity analyses to reveal the impact of the parameters on the performance of the proposed methods. Multiple case studies are provided to demonstrate the utility of the proposed method in practical applications.

The rest of this paper is organized as follows. Section 2 reviews the background to this research and related work, and Section 3 introduces the problem. Section 4 presents our proposed multi-task feature learning models, and Section 5 presents two efficient algorithms based on IGHT. The experiments on real Twitter datasets are presented in Section 6, and the paper concludes with a summary of the research in Section 7.

2 Related Work

This sections introduces the related work in the areas of 1) temporal mining of social media; 2) event detection and forecasting; 3) supervised and unsupervised learning; and 4) multitask learning.

Temporal mining of social media: In recent years, much attention has been paid to this area, which focuses on modeling the temporal pattern such as evolutional publich sentiment [25], dynamic topic [33], online collabrative environments [16], and information diffusion [31]. Tan et al. [25] proposed two topic models that leverage lexicon-based knowledge to characterize the variations of the public sentiment. Zhao et al. [33] developed a framework that can track themes of targeted domain dynamically utilizing the heterogeneous links such as co-occurrence, friendship, authorship, and replying. Guan et al. [16] proposed a method for locating appropriate expert on relevant knowledge by modeling and identifying people's knowledge based on their web activities. Zhang et al. [31] leverage triadic structures to investigate the formation of other neighboring links triggered by "following" links.

Event detection: A large body of work focuses on the identification of ongoing events, including earthquakes [23], disease outbreaks [24], and other types of events [3], [18], [29]. In general, these researchers use either classification or clustering to extract tweets of interest and then examine the spatial [23], temporal [29], or spatiotemporal burstiness [18] of the extracted tweets. However, instead of forecasting events in the future, these approaches typically uncover them only after their occurrence.

Event forecasting: Most research in this area focuses on temporal events and ignores the underlying geographical information. This approach is generally used for events such as the forecasting of elections [20], stock market movements [9], disease outbreaks [22], and crimes [28]. These studies can be grouped into three categorizes: 1) Linear regression model, where simple features, such as tweet volumes, are utilized to predict the occurrence time of future events [9], [20]; 2) Nonlinear models, where more sophisticated features such as topic-related keywords are used as the input to build forecasting models using existing methods such as support vector machines or LASSO [28]; and 3) Time series-based methods, where methods such as autoregressive models are used to model the temporal evolution of event-related indicators (e.g., tweet volume) [2]. However, few existing approaches can provide true spatiotemporal resolution for predicted events. Wang et al. [28] developed a spatiotemporal generalized additive model to characterize and predict spatio-temporal criminal incidents, but their model requires demographic data. Ramakrishnan et al. [21]

built separate LASSO models for different locations to predict the occurrence of civil unrest events. Zhao et al. [34] also designed a new predictive model based on topic models that jointly characterize the temporal evolution for both the semantics and geographical burstiness of social media content.

Supervised approaches: They involve considering a set of stationary terms whose distribution can be learned from historical data. For example, LASSO regression methods estimate a sparse predictive model based on a predefined set of keyword terms (vocabulary) for each location that predicts the probability of an ongoing event in this location in each predefined time interval (e.g., hourly or daily) [21]. Similarly, burst detection methods search for geographic regions (cities) where the aggregated counts of certain predefined terms are abnormally high compared with the counts for the same terms outside those cities. For example, Sakaki et al. utilize spatiotemporal Kalman filtering, which is similar to space-time burst detection, to track the geographical trajectory of hot spots of tweets related to earthquakes [23].

Unsupervised approaches: They utilize a set of dynamic terms that could be different in different time intervals, and apply unsupervised learning techniques for event detection. Here, the dynamic query expansion method (DQE) iteratively expands a predefined set of seed terms (e.g., protest, strike, march) by using current tweets to identify and rank new terms that are relevant to ongoing events, then retain the top terms and tweets containing these terms for further modeling [32]. Clustering-based methods search for novel spatial clusters of documents or terms using predefined similarity metrics, such as cosine similarity and social similarity for documents [3], or auto-correlations [2] and co-occurrences [29] for terms.

Multi-task learning: Multi-task learning (MTL) models multiple related tasks simultaneously to improve generalization performance [10], [26]. Many MTL approaches have been proposed in the past [36]. In [14], Evgeniou et al. proposed a regularized MTL that constrained the models of all tasks to be close to each other. The task relatedness can also be modeled by constraining multiple tasks to share a common underlying structure, e.g., a common set of features [5], or a common subspace [4]. MTL approaches have been applied in many domains, including computer vision and biomedical informatics. To the best of our knowledge, however, ours is the first work that applies MTL for civil unrest forecasting.

3 Problem Setup

Suppose there are m locations (e.g., cities, states) in the country of interest, and each location l has $n_{l,t} \in \mathbb{Z}$ tweets in each time interval t (e.g., hour, day). Define a matrix $C_{l,t} \in \mathbb{Z}^{p \times n_{l,t}}$, whose (i,j)-th entry, denoted as $C_{l,t}(i,j)$, refers to the frequency of the i-th term in the j-th tweet. Here p refers to the size of the vocabulary V. We are also given a binary variable $Y_{l,\tau} \in \{0,1\}$ for each location l at time τ, which indicates the occurrence ('yes' or 'no') of a future event. Therefore, given the input data $C_{l,t}$, the goal is to predict the future event occurrence $Y_{l,\tau}$ for a specific location l at a future time interval $\tau = t + \delta$ based on the tweets data collected, where δ is called the lead time of forecasting.

This work is built upon two of our previous predictive models, namely LASSO [21] and dynamic query expansion (DQE) [32]. Suppose we have a subset of keywords of size d in V that are relevant to the domain of interest and predefined by the domain experts, and denote A as the corresponding incidence matrix, $A \in [0,1]^{d \times p}$. Define a matrix $K_{l,t}$ as follows: $K_{l,t} = A \cdot C_{l,t} \cdot \mathbf{1}$, where $\mathbf{1}$ refers to a vector of all ones. It is clear that $K_{l,t} \in \mathbb{Z}^{d \times 1}$ is the vector of keywords frequencies in location l at time t. The LASSO model learns a separate sparse linear regression model for each location l:

$$\arg\min_{w_l} \left\| {w_l}^T K_{l,t} - Y_{l,\tau} \right\|_2^2 + \rho_1 \|w_l\|_1,$$

where the regularization parameter ρ_1 controls the sparsity, and $w_l \in \mathbb{R}^{d \times 1}$ is the vector of regression coefficients that need to be estimated. We need to estimate $m \cdot d$ parameters in total for the m separate LASSO regression models.

DQE is based on the idea that the specific topics of events under targeted domain could be quite varied and hence we must seek to grow our vocabularies of interest on the fly. Term co-occurrence is generally deemed to be an indicator of semantic proximity. A tweet and its replying tweets are causal in context, similar in semantics, and consistent in theme. Given a short seed query (e.g., "protest" and "march" for civil unrest domain), DQE adopts a query expansion strategy to expand the new keywords (e.g., "#OccupyWallSt" and "corruption") that appear with seed query in the same tweets or replying tweets. The volume and pattern of tweets containing these keywords are then utilized for event detection or forecasting. Denote $I(\cdot)$ as the indicator function. For each location l and time t, define the number of tweets containing any of the k dynamic keywords $S_t^{(k)}$ as $D_{l,t,k}$. Then, the DQE-based event forecasting can be formulated as a function $Y_{l,t} = I(D_{l,t,k} > \gamma)$, that is, $Y_{l,\tau} = 1$ if $D_{l,t,k}$ is larger than the threshold γ; $Y_{l,\tau} = 0$, otherwise. The dynamic keywords are expanded and ranked from the seed query based on the tweets data $C_{\cdot,t} = \{C_{l,t}\}_l$, where the seed query S_0 is an initial set of few semantically coherent keywords that characterize the concept of the targeted domain. Specifically, the keyword expansion process is formulated as follows:

$$P_t = F_t(B_t^T \cdot B_t + B_t^T R_t B_t) \cdot P_0$$

where $P_0 \in \mathbb{R}^{|V| \times 1}$ is the initial weight vector of all the words in V, $[P_0]_{i,1} = I(V_i \in S_0)$, and V_i is the ith word. B_t is the adjacency matrix between tweets and words. $R \in \mathbb{R}^{|C_t| \times |C_t|}$ is the tweet-replying matrix, and $[R_t]_{ij} = 1$ means there is replying relationship between tweet i and tweet j; $[R_t]_{ij} = 0$, otherwise. $F \in \mathbb{R}^{|V| \times |V|}$ is the inverse document frequency (IDF) matrix of F, which is a diagonal matrix such that $[F]_{ii}$ refers to the IDF of the word V_i. $P_t \in \mathbb{R}^{|V| \times 1}$ is the updated weight vector. Finally, the dynamic keyword set $S_t^{(k)}$ is defined as the top k words with the largest weights according to P_t.

There are three main challenges when using either LASSO or DQE individually: (1) The LASSO model only uses a set of predefined fixed keywords, called "static features," which may not capture fast-evolving expressions in Twitter, thus making it difficult to predict future events that are related to a small set of new keywords that are not included in the fixed keyword set. (2) The LASSO model trains an individual model for each location, but many small cities may have insufficient information in the training set to build an accurate forecasting model. (3) DQE requires two types of thresholds, namely 1) k, the number of dynamic keywords expanded from the seed query, and 2) γ, the least number of tweets, each of which can contain any of the dynamic keywords, to indicate the occurrence of an event. However, it is difficult to set these two thresholds based on domain experience. In the next section, we present a novel computational approach based on multi-task learning that addresses all three of these challenges.

4 MODELS

As defined above, LASSO uses the "static feature" set $K_{l,t}$, which is the count of predefined keywords in location l at time t. DQE uses the "dynamic feature" set $D_{l,t,k}$, which is the number of tweets containing the top k dynamic keywords at location l at time t. Because it is difficult to predefine an optimal k, we propose to make use of multiple k values in the range of $[1, s]$ (here s is a user-specified parameter; our experiments show that using a set of $s = 20$ values is sufficient), and then learn the optimal k automatically within the proposed multi-task learning framework. This results in $D_{l,t} = \{D_{l,t,k}\}_{k=1}^{s}, D_{l,t} \in \mathbb{R}^{s\times 1}$, which corresponds to the "dynamic feature" set for location l and time t. We combine the information used in LASSO and DQE by forming a new data matrix $X_{l,t} = [K_{l,t}; D_{l,t}] \in \mathbb{R}^{(d+s)\times n_{l,t}}$. For notational simplicity, we will remove the subscript t throughout the rest of this paper.

We aim to build m models $\{w_i | i = 1, \ldots, m\}$ to predict the occurrence of events for the m locations. A simple approach is to learn these m models (tasks) independently, ignoring the task relatedness. However, such an approach does not consider the intrinsic relationships among different locations (e.g., cities, states), and the resulting models may not be accurate as some locations may not have sufficient information in the training set. To address this issue, we propose to build the forecasting models for all m locations simultaneously by extracting and utilizing appropriate shared information across tasks while retaining their heterogeneity [36]. Figure 1 illustrates the proposed multi-task learning framework. Learning multiple related tasks simultaneously effectively increases the sample size for each location, since when we learn a model for a specific location, we also use information from all other locations.

Intuitively, the events that occur at different locations around the same time could well involve similar topics, thus the tweets from different locations may share many common keywords that are related to the events. This led us to explore multi-task feature learning (MTFL) models that constrain multiple related models to select a common set of features. Note that the heterogeneity among tasks is characterized as the difference in the weights of features for different tasks. For example, for two locations: a metropolis and a village, the importance of 1000 protest tweets to them differs, which can be characterized by the difference in the value scales of their models' feature weights. Specifically, we chose to explore four multi-task feature learning models:

- Regularized multi-task feature learning model,
- Constrained multi-task feature learning model I,
- Constrained multi-task feature learning model II,
- Constrained multi-task feature learning model III.

Each of these four models formulates the multi-task learning problem by following a general paradigm, i.e., to minimize a penalized empirical loss:

$$\min_{W} \mathcal{L}(W) + \lambda g(W) \tag{1}$$

or by implementing a constrained version:

$$\min_{W} \mathcal{L}(W) \text{ s.t.} \quad g(W) \leq l. \tag{2}$$

where $\mathcal{L}(W)$ is the empirical loss on the training set. Here we use a smooth and convex loss function, e.g., the least squares and logistics loss. $g(W)$ is the regularization term that encodes task relatedness, which is typically non-smooth or even non-convex. Therefore, $\mathcal{L}(W)$ tries to tailor each model to its specific task while $g(W)$ tends to find shared patterns across different tasks. λ (or l) is a tuning parameter to balance the tradeoff between them.

Different regularization/constraint terms capture different types of task relatedness [1], [12], [14], [17]. In this paper, we adopt the logistic loss, and characterize the model relatedness by restricting all models to select a common set of features. We discuss each of the four models in turn below.

4.1 Regularized MTFL model

The j-th element in model w_i indicates the importance of the j-th feature for the i-th task. In the regularized MTFL model, we restrict all tasks to share a common set of top features, so the forecasting models for all cities are based on the same subset of features. This can be achieved by grouping the j-th elements of all tasks together and selecting the top groups. Specifically, we consider the m entries of the j-th row of the matrix W as a group and use the $l_{2,1}$-norm regularization to identify the top groups [5]. Thus, the j-th feature, which corresponds to the j-th element in the models, is likely to be selected or not by all the models simultaneously, achieving our desired goal. Mathematically, we employ the following multi-task feature learning model:

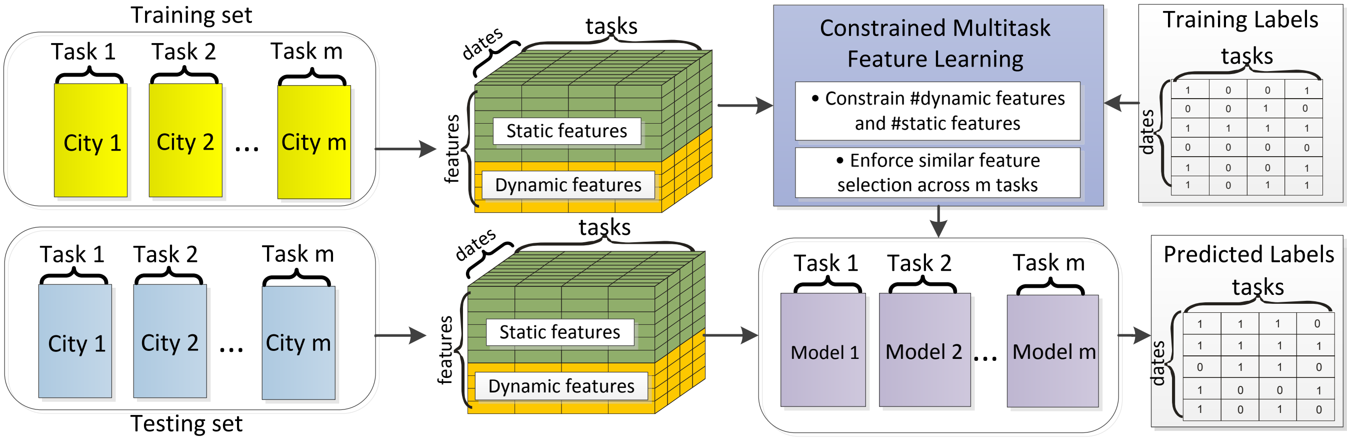
$$\min_{W} \mathcal{L}(W) + \rho_0 \|W\|_{2,1} + \rho_1 \|W\|_F^2, \tag{3}$$

where the first term is the data fitting term under logistic loss for all tasks such that $\mathcal{L}(W) = \sum_{i=1}^{m} \sum_{t} \log(1 + \exp(-Y_{i,\tau}(w_i \cdot X_{i,t}))$ and $\|W\|_{2,1}$ denotes the $l_{2,1}$ norm of matrix W which encourages all tasks to select a common set of features, and it can be computed as the summation of l_2-norm of each row in W. The regularization parameter ρ_0 controls the sparsity. We include a small multiple of the Frobenius-norm regularization, i.e., $\|W\|_F^2$, to enhance the robustness of the model. Problem (3) is a convex problem and can be solved by the FISTA algorithm [6].

4.2 Constrained MTFL model I

In the regularized MTFL model above, the model sparsity is controlled by the parameter ρ_1, which is less interpretable than the number of features selected. It is thus preferable to develop a model that directly controls the number of features to be selected. To this end, we introduce a constraint

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2017.2657624, IEEE Transactions on Knowledge and Data Engineering



1	0	0	1
0	0	1	0
1	1	1	1
0	0	0	0
1	0	0	1
1	0	1	1

1	1	1	0
1	1	1	1
0	1	1	0
1	0	0	1
1	0	1	0

Fig. 1: The flowchart of the proposed multi-task learning model

in the model that ensures that a specific number of rows of W will be non-zero, so we control the number of features included in the model. In particular, consider the following constrained multi-task feature learning model:

$$\begin{aligned} &\min_W \mathcal{L}(W) + \rho_1 \|W\|_F^2, \\ &\text{s.t. } \sum_j I(\|w^j\| > 0) \le r. \end{aligned} \tag{4}$$

Here w^j is the j-th row of W and $I(\cdot)$ is the indicator function. The constraint in (4) ensures that the number of nonzero rows of W is no larger than r, so no more than r features will be selected. Note that the convexity property no longer holds for Model (4). We will use the iterative Group Hard Thresholding framework to solve (4). More details are provided in the next section.

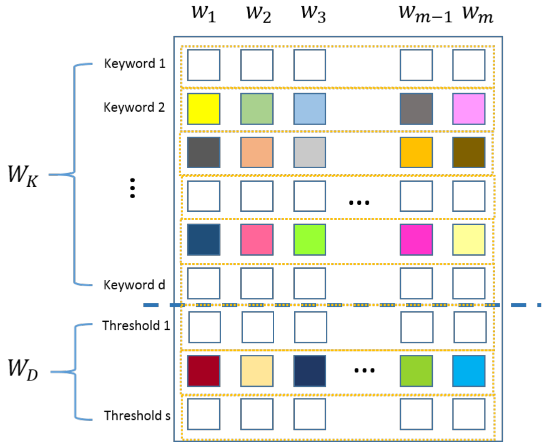


Fig. 2: Illustration of constrained MTFL model II. Each column represents the model for a specific location. The i-th row in W_K indicates the feature values for the i-th static feature (i.e., keyword), and the j-th row in W_D corresponds to the j-th dynamic feature (i.e., threshold value). Colored entries represent non-zero values in the model matrix, while white entries represent zeros.

4.3 Constrained MTFL model II

The constrained model above does not distinguish between the static and dynamic features. Recall that the first d features correspond to the d static features, while the last s features correspond to the use of s dynamic features. The feature values thus have very different meanings and in general, d is much larger than s. In our experiments, d is around 2000, while s is around 10 to 20. Thus, it is best to restrict the number of features selected from these two groups separately. In addition, in the current DQE model, only one dynamic feature is used and a common threshold value is applied for all cities in the same country. It is thus natural to restrict the number of dynamic features selected (out of the total s candidates) to be one. To achieve these goals, we propose the following model, which selects u features from the d static features, and selects v features from the s dynamic features:

$$\begin{aligned} &\min_W \mathcal{L}(W) + \rho_1 \|W\|_F^2, \\ &\text{s.t. } \sum_j I(\|w_K^j\| > 0) \le u, \\ &\quad\ \ \sum_j I(\|w_D^j\| > 0) \le v, \end{aligned} \tag{5}$$

where W_K is the model matrix corresponding to the set of static features, and W_D is the model matrix corresponding to the set of dynamic features. The structure of the model is depicted in Figure 2. As for Problem (4), u and v are user-specified parameters that control the number of features selected for the static feature set and dynamic feature set, respectively. We set $v = 1$ in our experiments, although our model is actually more general in that the user can select an arbitrary number of dynamic features.

Problem (5) is non-convex due to the use of nonconvex constraints. Similar to Problem (4), we can apply the Iterative Group Hard Thresholding algorithm to solve Problem (5). We show the details of our proposed algorithm for Problem (5) in Section 5.

4.4 Constrained MTFL model III

In the model CMTFL-II, the selection of static features in Equation (5) is known to be NP-hard and the existing efficient methods such as Iterative Hard Thresholding cannot guarantee a global optimization [7]. Additionally, the CMTFL-II model requires users to specify an appropriate number of static features. This target is difficult to accomplish by human labor and time consuming to achieve by cross-validation when the number of features is large and sensitive to the performance. To address these challenges, we propose Constrained MTFL model III (CMTFL-III), which automatically and globally optimizes the number of selected static features while still retaining the advantage of CMTFL-II, i.e., ensuring the selection of v dynamic features. CMTFL-III is formulated as below:

$$\min_{W} \mathcal{L}(W) + \rho_0 \|W_K\|_{2,1} + \rho_1 \|W\|_F^2, \\ s.t. \sum_j I(\|w_D^j\| > 0) \leq v, \tag{6}$$

where $W = \{W_K; W_D\}$ is the model matrix consisting of the set of static features W_K and dynamic features W_D. As for Problem (5), v is a user-specified parameter that controls the number of features selected for the set of dynamic features. We again set $v = 1$ in our experiments, although our model is more general in that the user can select an arbitrary number of dynamic features. As in Problem (5), Problem (6) is non-convex due to the use of nonconvex constraints. Problem (6) is solved by another proposed algorithm, which applies the Iterative Group Hard Thresholding algorithm and FISTA algorithm alternately until convergence is achieved. We show the details of our proposed algorithm for Problem (6) in Section 5.2.

4.4.1 Discussions

Based on multi-task learning framework, all of the proposed models rMTFL, CMTFL-I, CMTFL-II, and CMTFL-III can utilize the shared information among the event forecasting tasks of different spatial locations. The model rMTFL is the most basic one, which merely enforces the similar set of features to be selected across different tasks. However, in some situations that the user need to specify how many features to be selected, more sophisticated models are required that can constrain the number of selected features. This problem can be handled by the models CMTFL-I, CMTFL-II, and CMTFL-III. Among them, CMTFL-I can constrain the total number of selected features according to the user's need. rMTFL and CMTFL-I are unable to distinguish among different types of features. But in event forecasting in social media, in addition to traditional static features, the dynamic features are crucial and need to be ensured to be selected. Both CMTFL-II and CMTFL-III can address this problem by a constraint on the number of dynamic features to be selected. The difference between them lies in the strategy of static features selection. CMTFL-II utilizes a non-convex formulation for the static feature selection that cannot guarantee global optima, while CMTFL-III adopts a convex formulation which can be optimized exactly and efficiently.

5 Algorithm

The FISTA algorithm performs well for convex problems [6], [12], [36]. However, Problem (4), Problem (5), and Problem (6) are all non-convex. Even worse, they also involve discrete constraints, which make the problems particularly challenging to solve. Motivated by the success of the iterative hard thresholding algorithm for solving l_0-regularized problems [7] and recent advances in nonconvex iterative shrinkage algorithms [15], [30], we propose to employ the Iterative Group Hard Thresholding framework to solve both problems.

5.1 Algorithm for Models CMTFL-I and CMTFL-II

Note that Problem (4) is a special case of Problem (5) with $v = 0$. We thus focus on Problem (5) in the following discussion. The details are summarized in Algorithm 1. Here, data parallelism strategy is utilized to achieve the calculation of the gradient $\nabla f'(w_j^{i-1})$ in parallel for m different tasks. First, the variable H to store the array of gradients is defined. Then all of the tasks are evenly assigned onto multiple processors to calculate $\nabla f'(w_j^{i-1})$. After the calculation, the results from each processor are sent back to each $H_j \in H$. The detailed settings are specified in experiment section.

Algorithm 1 Algorithm for CMTFL-I and CMTFL-II

```
Require: $\boldsymbol{X}, \boldsymbol{Y}, \rho, \eta > 1$
Ensure: solution $\boldsymbol{W}$
Initialize $W^0$, $\eta \leftarrow 1$.
for $i \leftarrow 1, 2, \ldots$ do
    Initialize $L$
    for $j \leftarrow 1 \ldots m$ do in parallel
        $H_j \leftarrow \nabla f'(w_j^{i-1})$
    end for
    repeat
        $S^i \leftarrow W^i - \frac{1}{L}\nabla H$
        $W^i \leftarrow$ proj $(S^i)$ (defined in Lemma 1)
        $L \leftarrow \eta L$
    until line search criterion is satisfied
    if the objective stop criterion satisfied then
        return $W^i$
    end if
end for
```

Recall Problem (4), and denote $f(W) = \mathcal{L}(W) + \rho_1 \|W\|_F^2$. The key idea of IGHT is to first use the gradient information in the current iteration to provide the first-order approximation of the objective function, then apply the projection operators to ensure the next iteration satisfies the given constraints. Specifically, we use the combination of the linear approximation of the function $f(W)$ at a given point W^0 and a quadratic penalty term, and solve the following problem:

$$\min_{W} f(W^0) + \langle \nabla f(W^0), W - W^0 \rangle + \frac{\rho}{2} \|W - W^0\|_F^2, \\ \text{s.t.} \sum_j I(\|w_K^j\| > 0) \leq u, \\ \sum_j I(\|w_D^j\| > 0) \leq v, \tag{7}$$

where ρ is a positive constant that can be estimated by a line search scheme. By ignoring the constants and rearranging the terms in Problem (7), we obtain the following sub-problem:

$$\min_{W} \frac{1}{2} \|W - S\|_2^2 \\ \text{s.t.} \sum_j I(\|w_K^j\| > 0) \leq u \\ \sum_j I(\|w_D^j\| > 0) \leq v. \tag{8}$$

where $S = W^0 - \frac{1}{c}\nabla f(W^0)$. Problem (8) aims to find the optimal point satisfying the constraint set that is closest to a

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TKDE.2017.2657624, IEEE Transactions on Knowledge and Data Engineering

fixed point S. This can be treated as a Euclidean projection problem, denoted as proj$(\cdot)$, even though the constraint set is not convex. The key step in the IGHT framework solves the projection problem in (8). It is not hard to show that Problem (8) admits a closed-form solution as it can be decomposed into two independent problems, one for each block of features, as summarized in the following lemma.

Lemma 1. The projection Problem (8) admits a closed-form solution, given below:

$$w_K^j = \begin{cases} S_K^j, \text{if } j \in \Omega_K \\ 0, \text{otherwise} \end{cases} \tag{9}$$

and

$$w_D^j = \begin{cases} S_D^j, \text{if } j \in \Omega_D \\ 0, \text{otherwise} \end{cases} \tag{10}$$

where S_K consists of the first d rows of S, S_K^j is the j-th row of S_K, S_D consists of the last s rows of S, S_D^j is the j-th row of S_D, Ω_K is the index subset of $\{1, 2, \cdots, d\}$ of size u, including all rows of S_K that are among the top u rows of S_K in term of the length of the row vector, and Ω_D is the index subset of $\{1, 2, \cdots, s\}$ of size v, including all rows of S_D that are among the top v rows of S_D in term of the length of the row vector.

One remaining issue is how to estimate the step size, which determines the amount of movement made along a given search direction. In this paper, we apply the well-known Lipschitz criterion to select the step size. Finally, the time complexity of the proposed Algorithm 1 is $O(q \cdot r \cdot (s + d) \cdot m \cdot T)$, where q and r are the numbers of iterations for the outer and inner loops, respectively, T is the total number of the time intervals.

5.2 Algorithm for Model CMTFL-III

Note that Problem (6) encompasses an $l_{2,1}$-norm in the objective function similar to that in Problem (3) and utilizes a l_0-norm constraint similar to that in Problem (4). Accordingly, the solution to Problem (6) combines these notions from IGHT and FISTA. The details are summarized in Algorithm 2. Similar to Algorithm 1, data parallelism has been applied to different tasks in the loop in Line 4 and the loop in Line 10.

The key idea of the algorithm for CMTFL-III is as follows. First, we denote $f(W) = \sum_j^m f'(w_j)$, where $f'(w_j) = \sum_t \log(1 + \exp(-Y_{j,\tau}(w_j \cdot X_{j,t})) + \rho_1 \|w_j\|_F^2$. Applying a linear approximation, we get the first-order approximation to the original objective function in Problem 6, as shown in the following equation:

$$\begin{aligned} \min_W f\left(W^0\right) + \left\langle \nabla f\left(W^0\right), W - W^0 \right\rangle \\ + \frac{\rho}{2} \left\|W - W^0\right\|_F^2 + \rho_0 \|W_K\|_{2,1} \\ s.t. \ \sum_j \left\|w_D^j\right\|_0 \le v, \end{aligned} \tag{11}$$

where ρ is a positive constant that can be estimated using a line search scheme. By ignoring the constants and rearranging the terms in Problem 11, we obtain the following equivalent problem:

Algorithm 2 Algorithm for CMTFL-III

```
Require: $\boldsymbol{X}$, $\boldsymbol{Y}$, $\rho_0$, $\rho_1$, $\eta > 1$
Ensure: solution $\boldsymbol{W}$
 1: Initialize $W^0$, $\eta \leftarrow 1$.
 2: for $i \leftarrow 1, 2, \ldots$ do
 3:     Initialize $L$, $H$
 4:     for $j \leftarrow 1 \ldots m$ do in parallel
 5:         $H_j \leftarrow \nabla f'(w_j^{i-1})$
 6:     end for
 7:     repeat
 8:         $S \leftarrow W^{i-1} - \frac{1}{L}\nabla H$
 9:         $W_D^i \leftarrow \text{proj}\left(S_D\right)$
10:         for $j \leftarrow 1 \ldots d$ do in parallel
11:             $[W_K^i]_j \leftarrow \text{prox}_{\ell_{2,1}}([S_K]_j)$
12:         end for
13:         $L \leftarrow \eta L$
14:     until line search criterion is satisfied
15:     $W^i \leftarrow \left[W_K^i; W_D^i\right]$
16:     if the objective stop criterion satisfied then
17:         return $W^i$
18:     end if
19: end for
```

$$\begin{aligned} \min_W \tfrac{1}{2} \left\|W - S\right\|_F^2 + \rho_0 \|W_K\|_{2,1} \\ s.t. \ \sum_j \left\|w_D^j\right\|_0 \le v, \end{aligned} \tag{12}$$

where $S = W^0 - \frac{1}{L}\nabla f\left(W^0\right)$. Note that Problem (12) can be decomposed into the following two subproblems:

$$\begin{aligned} \min_{W_D} \tfrac{1}{2} \left\|W_D - S_D\right\|_F^2 \\ s.t. \ \sum_j \left\|w_D^j\right\|_0 \le v, \end{aligned} \tag{13}$$

and

$$\min_{W_K} \frac{1}{2} \left\|W_K - S_K\right\|_F^2 + \rho_0 \|W_K\|_{2,1} \tag{14}$$

where Problem (13) can be solved by applying the hard thresholding algorithm and Problem (14) can be solved using the FISTA algorithm.

The time complexity of the proposed Algorithm 2 is $O(q \cdot r \cdot (s + \gamma d) \cdot m \cdot T)$, which is composed of the computation of the dynamic features $O(q \cdot r \cdot s \cdot m \cdot T)$ and static features $O(q \cdot r \cdot \gamma \cdot d \cdot m \cdot T)$ where $\gamma \cdot d = O(d)$ is the computation time for a block soft thresholding on the weights of static features and γ is a constant.

6 Experiments

In this section, we evaluate the performance of the proposed multi-task learning formulations. First, we evaluate the effectiveness and efficiency of the methods using multiple real datasets and compare the results with those obtained using existing baseline methods on multiple event forecasting tasks. We then move on to study the parameter sensitivity of the methods. Finally, we provide several empirical case studies of event forecasting for civil unrest and influenza outbreaks to demonstrate the utility and practicality of these forecasting models.

TABLE 1: Twitter datasets and gold standards (GSR)

Country	Domain	Time Period	#Tweets (million)	Gold Standard[1]	#Events
Brazil	civil unrest	07/01/2012-05/31/2013	57	O Globo; O Estado de São Paulo; Jornal do Brasil	451
Paraguay	civil unrest	07/01/2012-05/31/2013	8	ABC Color; Ultima Hora; La Nacíon	563
Mexico	civil unrest	07/01/2012-05/31/2013	51	La Jornada; Reforma; Milenio	1217
Venezuela	civil unrest	07/01/2012-05/31/2013	45	El Universal; El Nacional; Ultimas Notícias	678
the United States	influenza	01/01/2011-04/30/2014	9,586	Centers for Disease Control and Prevention	127

6.1 Experiment Setup

In the experimental evaluation, two datasets for different regions, Latin America and the United States, were utilized for the research on civil unrest and influenza outbreaks, respectively.

6.1.1 Datasets

For the datasets on Latin America, the raw data was obtained by randomly sampling 10% (by volume) of the Twitter data from July 2012 to May 2013 in 4 countries, namely Brazil, Paraguay, Mexico, and Venezuela, as shown in Table 1. Twitter data collection is partitioned into a sequence of date-interval subcollections. The Twitter data for the period from July 1, 2012 to December 31, 2012 is used for training, while the data for the second half of the period, from January 1, 2013 to May 31, 2013, is used for the performance evaluation. The locations of the tweets are geocoded by the geocoder in [21]. The event forecasting results are validated against a labeled events set, known as the gold standard report (GSR), exclusively provided by MITRE [19]. GSR is a collection of civil unrest news reports from the most influential newspaper outlets in Latin America [32], as shown in Table 1. An example of a labeled GSR event is given by the tuple: (CITY="Hermosillo", STATE = "Sonora", COUNTRY = "Mexico", DATE = "2013-01-20").

For the datasets in the United States, the raw data was crawled from January 2011 to April 2014 in all 50 states, as shown in Table 1. As in the first dataset, Twitter data collection is partitioned into a sequence of date-interval subcollections. The Twitter data for the period from January 1, 2011 to December 31, 2012 is used for training while the second half of the period, from January 1, 2013 to April 30, 2014, is used for the performance evaluation. The locations of the tweets are geocoded by the Carmen geocoder [13], which resolves the location of each tweet into a tuple containing information at the country, state, county, and city level. About 70% of the tweets in our dataset are assigned a location by Carmen. The forecasting results for the flu outbreaks are validated against the corresponding influenza statistics reported by the Centers for Disease Control and Prevention (CDC) [11]. CDC publishes the weekly influenza-like illness (ILI) activity level within each state in the United States based on the proportion of outpatient visits to healthcare providers for ILI. There are 4 ILI activity levels: minimal, low, moderate, and high, where the level "high" corresponds to a salient flu outbreak and is considered for forecasting. An example of a CDC flu outbreak event is: (STATE = "Virginia", COUNTRY = "United States", WEEK = "01-06-2013 to 01-12-2013").

6.1.2 Settings

In this experiment, two types of features are utilized. As described above, the first type consists of static features, which examine the relevance of tweets to fixed keywords. Specifically, these are defined as the daily counts of the keywords in the tweets. For the civil unrest domain, the keyword set includes 614 civil unrest related words (such as "protest" and "riot"), 192 phrases (such as "election fraud"), and country-specific actors (e.g., political parties and public figures). For each keyword, its translations in Spanish, Portuguese, and English are all included. For the influenza outbreaks, the keyword set includes 545 disease-related words extracted based on the keywords list used in [35]. The second type consists of dynamic features, which examine the volume of tweets containing dynamic keywords. Specifically, dynamic features are a set of counts, where each count is the number of daily tweets containing any of the top k ($k \in [1, s]$) dynamic keywords. The dynamic keywords are extracted and ranked based on dynamic query expansion (DQE) [32], which utilizes both semantic and social relationships to expand real-time keywords from the original seed query, as described in Section 3. For the civil unrest domain, the seed query terms include: "protest", "march", "movement", "patriotic", "manifest", and their translations in Spanish and Portuguese. For the influenza outbreaks domain, the seed query terms include: "flu", "influenza","h1n1","h5n1", and "h7n9". In this experiment, s was set to 20. Thus we have 20 dynamic features. The experiments were conducted on a 64-bit machine with 80 processors (Intel Xeon CPU E7-4850@2.00GHz) and 528.0GB memory. Our parallel algorithm is based on openMP with the C++ compiler GCC 5.1.0[2]. 20 threads were used for each parallel loops in the Algorithms 1 and 2.

In the experiment, given the day-by-day tweet data, the event forecasting task is to utilize one day tweet data to predict whether or not there will be an event in the next day for a specific city (for the civil unrest domain), or a specific state (for the influenza outbreaks domain). To perform this task, we create a training set and a test set for each city (or state), where each data sample is the daily tweet observation with the above-mentioned features. On the training set, we set the label for each data sample as "1" if there is an event on the next day; and "0" otherwise. The predicted events are structured as tuples of (date, city/state). A predicted event is matched to a GSR event if both the date and

[1] In addition to the top 3 domestic news outlets in each country, the following news outlets were included: The New York Times; The Guardian; The Wall Street Journal; The Washington Post; The International Herald Tribune; The Times of London; Infolatam.

[2] Downloadable: http://tdm-gcc.tdragon.net/download. Dec 2016.

TABLE 2: Event forecasting performance in AUC

Dataset	DQEF	LASSO-K	DQEF+LASSO	LASSO	rMTFL-D	rMTFL-K	rMTFL	CMTFL-I	CMTFL-II	CMTFL-III
Venezuela	0.5358	0.5586	0.633	0.6073	0.6486	0.6497	0.7889	0.7363	0.758	**0.8011**
Mexico	0.5397	0.4989	0.5627	0.5749	0.6817	0.6151	0.6831	0.6719	0.6934	**0.7019**
Brazil	0.4954	0.451	0.5108	0.4774	0.6466	0.4295	0.605	0.6049	**0.6518**	0.651
Paraguay	0.5592	0.5657	0.7177	0.6237	**0.8307**	0.6605	0.8013	0.8039	0.8232	0.8151
Flu	0.4706	0.6824	0.4783	0.6497	0.7207	0.74	0.7501	0.7643	0.7687	**0.8036**
Overall	0.4939	0.6236	0.519	0.6243	0.7114	0.6934	0.7369	0.7406	0.7519	**0.7786**

city/state attributes are matched; otherwise, it is considered a false forecast. To validate the prediction performance, the Area Under the Curve (AUC) of Receiver operating characteristic (ROC) curve were adopted. ROC curve illustrates the performance of a binary classifier as its discrimination threshold is varied. The curve is created by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings. The AUC measures the area below this curve, which is a well-recognized metric to reflect the comprehensive performance of a classifier.

6.1.3 *Comparison Methods*

The following methods are included for the performance comparison:

1) **LASSO** [27]. For each location, two LASSO models are trained utilizing different sets of features: i) both static and dynamic features, and ii) only static features (denoted as **LASSO-K**). The regularization parameters of these models for different cities are set based on a 10-fold cross validation.
2) DQE-based event forecasting (**DQEF**). This model only considers the dynamic features, as explained in Section 3. The number of top dynamic keywords, k, and the tweet count threshold γ are set for each country by a 10-fold cross-validation on the training set.
3) **DQEF+LASSO**. For each location, the DQEF method is first used to perform the forecasting. If there is no predicted event, i.e., $Y_{l,t} = 0$, the LASSO model using only static features will be employed for forecasting.
4) Regularized Multi-task Feature Learning Model (**rMTFL**). For each country, an rMTFL model is built where each task consists of the event forecasting for a location. This model utilizes three sets of features: i) both static and dynamic features, ii) only static features (denoted as **rMTFL-K**); and iii) only dynamic features (denoted as **rMTFL-D**). The regularization parameters ρ_1 and ρ_0 are set based on a 10-fold cross-validation.
5) Constrained multi-task feature learning model I (**CMTFL-I**). For each country, a model is built where each task consists of the event forecasting for a location. All the tasks share the same features, i.e., both static and dynamic features. The feature number constraint r and the regularization parameter ρ_1 are set based on a 10-fold cross-validation.
6) Constrained multi-task feature learning model II (**CMTFL-II**). Once again, for each country, a model is built where each task is the event forecasting for a location. All the tasks share the same features, i.e., static and dynamic features. We use a 10-fold cross-validation to set the regularization parameter ρ_1, the numbers of static features u, and dynamic features v for each country. The sensitivities of these three parameters are discussed in Section 6.3.
7) Constrained multi-task feature learning model III (**CMTFL-III**). For each country, a model is built where each task consists of the event forecasting for a city/state. All the tasks utilize both static and dynamic features and we use the a 10-fold cross-validation to set the regularization parameters ρ_0, ρ_1 and dynamic features v for each country.

6.2 Performance

The proposed and comparison methods are evaluated on both the civil unrest and influenza outbreak datasets. Both quantitative and qualitative evaluations are conducted, described in more detail in the following.

6.2.1 *Quantitative evaluation*

Table 2 summarizes the comparison among the various methods for event forecasting in five different datasets. Among them, four datasets of four different countries Venezuela, Mexico, Brazil, and Paraguay are in civil unrest domain; the other dataset is for flu outbreaks in the United States. The results show that the methods that utilize both static and dynamic features perform better than those utilizing either one alone. For example, the rMTFL model outperforms rMTFL-D and rMTFL-K by 3% and 6%, respectively. These results confirm the effectiveness of combining both types of features for event forecasting. Among all the methods, the four proposed models rMTFL, CMTFL-I, CMTFL-II, and CMTFL-III achieve the score over 0.73, outperforming the baselines. The data presented in Table 2 show that the multi-task models outperformed the traditional LASSO models by 20% on average. This reveals the advantage enjoyed by the multi-task models, which can select features by learning from similar forecasting tasks for all the cities (or states). The generalization and stability of the forecasting performance can be improved by learning models for different cities together, especially for those cities that lack sufficient training samples. And CMTFL-III obtains the best overall performance in these five datasets. For the countries Venezuela, Mexico, and United States, CMTFL-III achieves the best performance, and for the other two datasets, Brazil and Paraguay, it still achieves the second and third best performance among all the 10 methods. This is likely because: (1) CMTFL-III is able to ensure the inclusion of dynamic features, which is demonstrably more effective than only using static features alone in the modeling; and (2) Unlike CMTFL-I and CMTFL-II, CMTFL-III does not require the determination of the number of selected static features or the number of selected total features, which are parameters sensitive to the performance, as shown in Figure 5. And it is time-consuming to tune them by cross-validation when the total number of all the features is non-small.

TABLE 3: Top 10 static features (translated into English) and the selection of dynamic features for the civil unrest domain. TRUE means there is at least one dynamic feature being selected; FALSE means no dynamic feature selected. CMTFL-II and CMTFL-III can ensure the selection of effective dynamic feature(s). CMTFL-III obtains the features with higher quality.

		Mexico					Brazil		
Methods	Features	Mexico City	Cuernavaca	Guadalajara	Morelia	Oaxaca	Brasília	Rio de Janeiro	São Paulo
LASSO	Static	block fight work help hearsay president initiation occupy request power	complaint gunfire tranquility forward power avoid	request confront water danger results order help national	request meet water danger results order help national initiation town	help power avoid	send power food forward money street	problem water official work fight government national employ	throw bond unit defeat send forward control confront expensive finish
	Dynamic	TRUE	FALSE	TRUE	FALSE	TRUE	FALSE	TRUE	TRUE
rMTFL	Static	fight movement election president congress initiative progress hard help government	fight hate hungry street sent calling hungry work eliminate forcibly	remember street work hate president unit poor permit killing remove	employ remember unit water university change class statement force problem	university allow work develop hatred problem progress released congress killing	participant increased expensive prepare include protest strength march gringo screams	expensive strength gringo cries progress participant protest student censorship include	prisoners expensive increase cries force include censorship progress prepare student
	Dynamic	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE	FALSE
CMTFL-II	Static	protest fight president government movement death poor national expected wait	police protest struggle patriot movement hunger student block work memories	university expected movement manifest occupy hate change class block official	movement occupy encounter hunger national change request fear money country	block money encounter memories change police occupy steal fight president	shooting order movement throw government submit march national block attack	attack block occupy arrest control kill followers throw ask march	march resolve attack warrant payment poor claim block hatred problem
	Dynamic	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
CMTFL-III	Static	protest crisis rage impose embargo conflict call-for angry fight hate	force protesters development embargo military punishment effort march violence criminal	treaty matches killer angry assault conflict march unemployment defeat Workers	punishment Rejected eviction fire ban crisis force army embargo attorney	burning revenge control embargo problem town march military to break labor	justice atrocity protest Racism solve community unity organized censorship punishment	abortion punishment racism extreme protest poor minister kill hate Burning	force atrocity Justice punishment protest attack torture censorship power military
	Dynamic	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE

6.2.2 *Qualitative evaluation*

Table 3 shows the specific features selected by different models, including LASSO, rMTFL, CMTFL-II, and CMTFL-III for several cities in two countries, namely Mexico (where Spanish is spoken) and Brazil (where Portuguese is spoken). As Table 3 shows, CMTFL-II and CMTFL-III effectively select static features (i.e., keywords) that are very relevant to civil unrest, and the selection is stable and consistent across different cities. The geographical heterogeneity is reflected in the difference of the top features selected by the models of different locations. Moreover, the selection of dynamic feature(s), as shown in the bottom row, enhances the capacity to consider the burstiness of tweets containing dynamic keywords. The rMTFL model also performs effectively when selecting civil unrest-related keywords as the top static features. However, this model cannot guarantee the selection of dynamic features because it fails to select dynamic features in any of the listed cities for Brazil. The static features the LASSO model selects are not consistent across different cities and, more importantly, are not as relevant and sufficient as those identified by the above-three multi-task learning models in several cities, especially in smaller cities, such as Oaxaca and Cuernavaca. Additionally, the selection of dynamic features is not guaranteed, as is the case in Morelia and Brasília.

Table 4 shows the specific features selected by the different models, including LASSO, rMTFL, CMTFL-II, and CMTFL-III for several states of the United States for outbreaks of influenza. According to Table 4, CMTFL-III achieves most effective selection of static features (i.e., keywords) that are relevant to the description of catching flu, such as “flu”, “sick”, “cold”, and “chills”. CMTFL-II obtains effective selection of related keywords, but involves relatively more general keywords like “stay” and “around” while misses some important ones like “flu” and “illness”. Table 4 also shows that performance of CMTFL-II and CMTFL-III are stable and consistent across different states, regardless of whether it is a large state such as New York state or one with a small tweet volume like Alaska. Moreover, the selection of dynamic feature(s) is ensured, as shown in the bottom row, thus enhancing the capacity to consider the burstiness of tweets containing flu-related dynamic keywords. The rMTFL model also selects some influenza-related keywords as its top static features. However, the quality of the top keywords is not as high as that for CMTFL-III. The selected static features for the LASSO model are not consistent across different states and, more importantly, not as relevant and sufficient as the above-two multi-task learning models in several states, especially those with a small tweet volume, such as Alaska, where only one static keyword “immune” is selected. Additionally, the selection of dynamic features is not ensured, for example in Nebraska, Washington, and New York.

6.3 Parameter Sensitivity Study

There are totally five tunable parameters in all the four proposed models, rMTFL, CMTFL-I, CMTFL-II, and CMTFL-III model, namely 1) the regularization parameter ρ_0 for rMTFL

TABLE 4: Top 10 static features and the selection of dynamic features for the influenza outbreaks domain. TRUE means there is at least one dynamic feature being selected; FALSE means no dynamic feature selected. CMTFL-II can ensure the selection of effective dynamic feature(s). CMTFL-III obtains the features with higher quality.

		the United States							
Methods	Features	Wyoming	Nebraska	Washington	New York	California	Alaska	Florida	New Mexico
LASSO	Static	four excuse works job diet cancelled boss ankle complicate NIH	birds drop thinks dealing warm body pissed practice masks class	jadi tired 101 birds 2nd cancer classes hands miss recover	drop chicken vomiting late bottle quickly miserable ate brought hrs	fast sleep decided ill started quite normal less years gak	immune	kalo four past 12s pigs pissed heard tea infected wasn	officially tea juga drop strains die nausea swear fight gettin
	Dynamic	TRUE	FALSE	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE
rMTFL	Static	catching jab vaccination excuse daughter quickly outbreak poor died four	warm goin drop practice thinks class pissed excuse dealing body	ankle poor pray gym disease jadi finally quarantine thera severe	drop chicken begginning hospitalize month infections kind throat bro barely	fast appetite tired quite lemon energy vomit sleep normal killing	immune fever strep bug bird week flu virus vaccination tomorrow	12s pigs ebola past wasn helps tea practice heard kalo	comining slime thanks vomiting tea less positive catch starting weak
	Dynamic	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
CMTFL-II	Static	sucks week bed home work days sick year doctor around	strep stay around house bed feeling work days week tomorrow	house away days tonight bug stay doctor bed school week	house around doctor school away sick symptoms bed home tonight	house school fever days sucks tonight bug stay bed tomorrow	sick cold bed school around home swine away throat bug	year soon tonight bug symptoms coming since tomorrow around work	days stay coming tomorrow away strep bug house soon sick
	Dynamic	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
CMTFL-III	Static	flu sick cold days bed feeling stomach week work soon	stomach cold sick feeling week days bed soon work flu	cold sick bed week days flu sucks stomach soon feeling	bed stomach cold days soon family sucks week feeling sick	bed days feeling cold week sick soon work sucks family	chills illness trip official wanted bring decided cancelled avoid taking	stomach flu soon sick days work awful body least pretty	sick stomach bed cold days flu week feeling work soon
	Dynamic	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE

and CMTFL-III; 2) the Tikhonov regularization parameter ρ_1 for all the four models; 3) the number of selected static features u for CMTFL-II; 4) the number of selected dynamic features v for CMTFL-II and CMTFL-III; and 5) the number of all the features for CMTFL-I. In this experiment, the AUC scores for the Venezuela dataset are illustrated; those for the other datasets exhibit a very similar pattern.

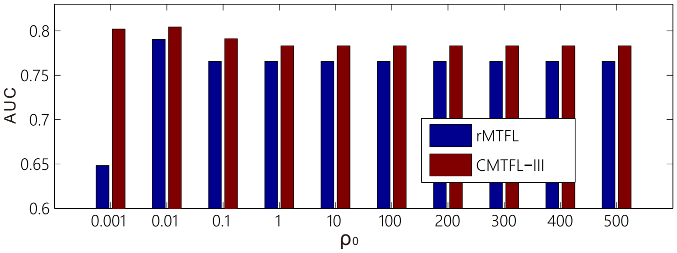


Fig. 3: Sensitivity analysis for the regularization parameter ρ_0.

Figure 3 illustrates the performance of the proposed model versus, ρ_0, the regularization parameter. By varying ρ_0 over a large range from 0.001 to 500, the performance in terms of AUC remains stable for CMTFL-III. The AUC score for rMTFL increases by 0.13 when ρ_0 increases from 0.001 to 0.01 and becomes stable after that.

Figure 4 shows that by varying ρ_1 over a large range from 0.001 to 500, the AUC scores for all the four models remain stable with the fluctuation ranges less than 0.02.

Figure 5 shows the sensitivity results of varying the number of selected features for different models. Figure 5(a) demonstrates that by changing the number of dynamic features from 1 to 20, the AUC scores change within 0.04 for both CMTFL-II and CMTFL-III. In Figure 5(b), the number of static features is shown to be sensitive to the performance of the model CMTFL-II. The dramatic fluctuation of AUC happens every 10-20 increase of the number of selected static features. This demonstrates the difficulty in tuning the parameter in the model CMTFL-II and the advantage of CMTFL-III because its parameter ρ_0 is not that sensitive as shown in Figure 3. Finally, the number of the selected features for the model CMTFL-I is also sensitive with the fluctuation as large as 0.2.

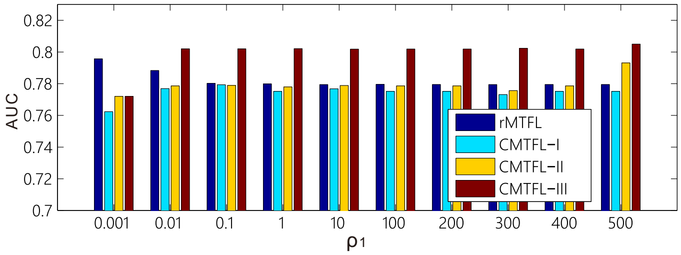
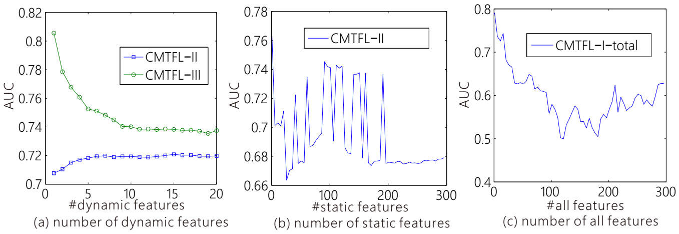


Fig. 4: Sensitivity analysis for the regularization parameter ρ_1.

6.4 Scalability Analysis

To examine the scalability of the proposed methods, we can measure the training runtimes of all the methods when varying the number of tasks and features. Here, we present the results of our experiments for the influenza outbreak dataset; the performance on the civil unrest dataset exhibits a similar pattern.



(a) number of dynamic features
(b) number of static features
(c) number of all features

Fig. 5: Sensitivity analysis for the number of selected features.

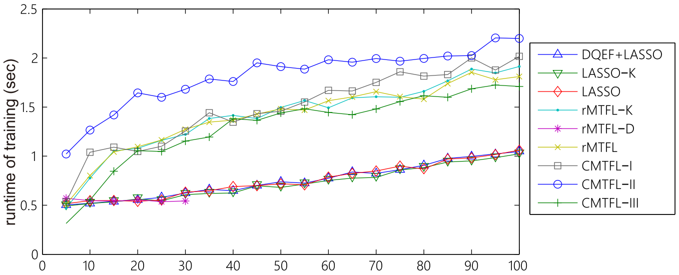


Fig. 6: Scalability on number of features.

Figure 6 compares the running times for all the methods when the number of features they utilize changes from 5 to 100. As can be seen from the graph, the runtimes of all the methods basically increase linearly with the number of features. Among them, the methods DQEF+LASSO, LASSO-K, and LASSO require shorter runtimes compared to other methods because they are much simpler. The parallel computing strategy for the proposed models effectively reduces the computation time. CMTFL-III achieves a relatively low computation time among the proposed models due to the parallel strategy for computing in both different features and tasks as shown in Steps 4 and 10 of Algorithm 2.

To examine the scalability for an increasing number of tasks, Figure 7 illustrates the running times of all the methods when the number of tasks jumps from 4 to 40. Similar to the situation shown in Figure 6, the runtimes of all the methods increase linearly with the number of tasks, demonstrating good scalability of the proposed methods with the number of tasks. Note that, the simplest methods, namely DQEF+LASSO, LASSO-K, and LASSO, achieve little shorter runtimes on average. The proposed methods such as CMTFL-III and CMTFL-II are also very efficient (i.e., less than 10s when considering 40 tasks) for practical applications such as these thanks to the use of parallel optimization algorithms.

6.5 Case Studies

We observed numerous interesting events predicted by three of the proposed approaches, CMTFL-I, CMTFL-II, and CMTFL-III, in our experiments. For the civil unrest domain, Figures 8 and 9 depict two waves of civil unrest events that occurred on March 17th, 2013 in Brazil, and April 17th, 2013 in Paraguay, respectively. For the influenza outbreak domain, Figure 10 illustrates the influenza outbreaks occurring between Feb 10th, 2013 and Feb 16th, 2013 in the United States.

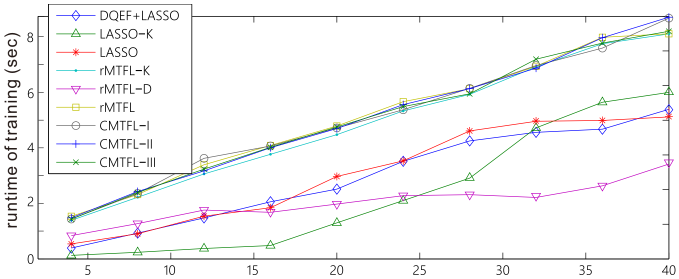


Fig. 7: Scalability on number of tasks.

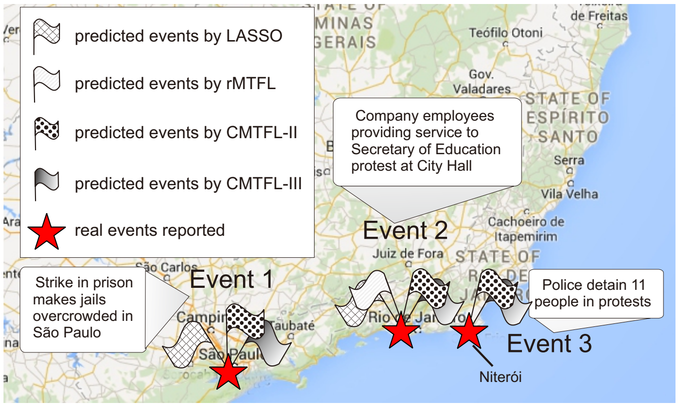

predicted events by LASSO
predicted events by rMTFL
predicted events by CMTFL-II
predicted events by CMTFL-III
real events reported
Company employees providing service to Secretary of Education protest at City Hall
Strike in prison makes jails overcrowded in São Paulo
Police detain 11 people in protests
Event 1
Event 2
Event 3
Niterói
MINAS GERAIS
STATE OF ESPÍRITO SANTO
Teófilo Otoni
Gov. Valadares
Serra
Vila Velha
Cachoeiro de Itapemirim
Juiz de Fora
São Carlos
Campinas
Taubaté
São Paulo
Rio de Janeiro
de Freitas

Fig. 8: A map of civil unrest events and forecasting hotspots on March 17th, 2013 in Brazil.

6.5.1 Case Studies on civil unrest forecasting

For the case studies in the civil unrest domain, Figure 8 shows three events in Brazil, among which Event 1 and Event 2 happened in large cities, namely Sao Paulo and Rio de Janeiro, respectively, while Event 3 was in a smaller city, Niterói. Note that the city Niterói does not have any training sample. The proposed CMTFL-II and CMTFL-III models successfully predicts all three of these events, even the one that occurred in Niterói. This is because CMTFL-II and CMTFL-III jointly learn the models for all the tasks (i.e., cities), so even where the model of the city has no training sample, it can still be estimated using data from other cities. The LASSO model predicts two of the events but fails to forecast Event 3. This is because the LASSO model is trained for each city individually, and so events that occur in a city with no training sample cannot be predicted. The rMTFL model only predicts one event, that in Rio de Janerio. Its failure to discover the events in the two other cities might be due to its exclusion of the dynamic features after training, as shown in Table 3. This reduces its capacity to uncover the burstiness of dynamic keywords. This confirms the need for a separate selection of the static and dynamic features, as in our proposed CMTFL-II model.

Figure 9 shows four events in Paraguay, among which Event 2, Event 3, and Event 4 were successfully predicted by CMTFL-II. And Event 1, Event 3, and Event 4 were successfully predicted by CMTFL-III. rMTFL predicted Event 2 and Event 3, while LASSO failed to predict any of the events. As shown in Table 1, Paraguay is a country where the number of reported events is large but the volume of tweets is relatively small, so the ratio of #tweets (or #events) is less than one third of that seen in other countries. The sparsity of tweet data makes forecasting more difficult for Paraguay for methods that do not incorporate using multi-task learning, as shown in Table 2.

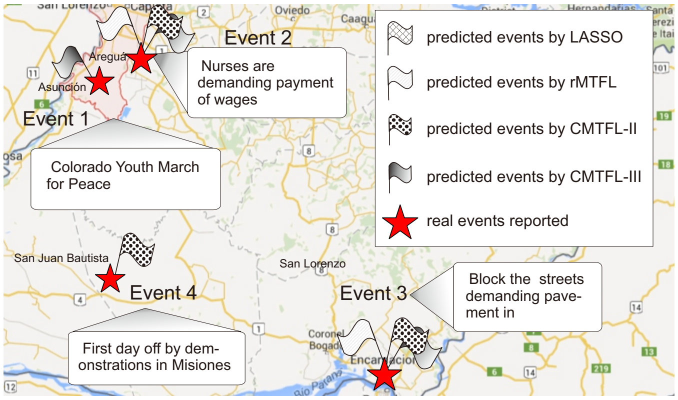

Event 2
Nurses are demanding payment of wages
Asunción
Areguá
Event 1
Colorado Youth March for Peace
San Juan Bautista
Event 4
First day off by demonstrations in Misiones
San Lorenzo
Event 3
Block the streets demanding payement in
Encarnación
predicted events by LASSO
predicted events by rMTFL
predicted events by CMTFL-II
predicted events by CMTFL-III
real events reported

Fig. 9: A map of civil unrest events and forecasting hotspots on April 17, 2013 in Paraguay.

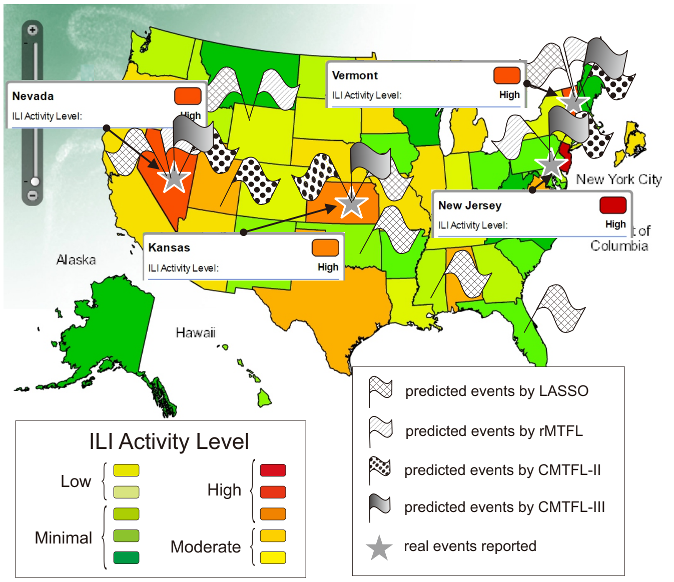


Fig. 10: A map of influenza outbreaks and forecasting hotspots for the period Feb 10 - Feb 16, 2013 in the United States.

6.5.2 Case Study on forecasting influenza outbreaks

For the case study on the influenza outbreak domain, Figure 10 shows that there were basically four states with high influenza activity in the United States that week, among which Nevada and Kansas are two states with relatively small average volume of tweet postings. The proposed rMTFL , CMTFL-II, and CMTFL-III models successfully predicted all of the events for both the smaller and larger states. This is because they jointly learned the models for all the tasks (i.e., cities). Even the model of the state (in this case, Alaska) with the fewest training samples can still be estimated by using data from other states. Among them the CMTFL-III performed the best because it did not generate any false positives while rMTFL and CMTFL-III have one false alarm in a state. This again demonstrated the effectiveness of CMTFL-III in optimizing the static feature selection and ensuring the inclusion of dynamic features. The LASSO model successfully predicted two of the influenza outbreak events but failed to forecast the events in Nevada and Kansas. This is because the LASSO model is trained for each state individually, and thus the performance for events in states with small training sets cannot be guaranteed. However, although both the rMTFL and CMTFL-II models successfully identified all the events, they also generated several false alarms. For example, the rMTFL model generated 3 false alarms in the states of Mississippi, Oklahoma, and Florida. CMTFL-II generated only one false alarm, for the state of Colorado, which does actually coincide with nontrivial flu activity. The better performance of CMTFL-II compared to rMTFL might be due to the consideration of dynamic features. Overall, this case study confirms the need for a separate selection of the static and dynamic features, as in our CMTFL-II model.

7 Conclusion

This paper presents a novel multi-task learning framework to address the problem of spatial event forecasting in Social Media. Existing methods are not able to concurrently address critical challenges, such as the dynamic patterns of features, and geographic heterogeneity. Our work considers the estimation of predictive models in different locations as a multi-task learning problem, thus making it possible to use shared information between locations and effectively increasing the sample size for each location. We further model the static and dynamic features using different constraints to balance both the homogeneity and diversity between these two types of features. We propose a set of efficient algorithms based on the IGHT that are able to predict spatial events in real time. Our empirical results demonstrate that we can effectively detect civil unrest and influenza outbreak events, outperforming existing methods by a substantial margin on both precision and recall. Multiple case studies are provided to demonstrate the usefulness of the proposed method in practical applications. In future work, we plan to extend our multi-task learning framework by exploring more complex relationships between locations and integrating human domain knowledge as priors.

Acknowledgment

Supported by the Intelligence Advanced Research Projects Activity (IARPA) via DoI/NBC contract number D12PC000337, the US Government is authorized to reproduce and distribute reprints of this work for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoI/NBC, or the US Government.

References

[1] J. Abernethy, F. Bach, T. Evgeniou, and J.-P. Vert. A new approach to collaborative filtering: Operator estimation with spectral regularization. *The Journal of Machine Learning Research*, 10:803–826, 2009.

[2] H. Achrekar, A. Gandhe, R. Lazarus, S.-H. Yu, and B. Liu. Predicting flu trends using Twitter data. In *IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 702–707, 2011.

[3] C. C. Aggarwal and K. Subbian. Event detection in social streams. In *SDM*, pages 624–635, 2012.

[4] R. Ando and T. Zhang. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6:1817–1853, 2005.

[5] A. Argyriou, T. Evgeniou, and M. Pontil. Multi-task feature learning. *Advances in Neural Information Processing Systems*, 19:41, 2007.

[6] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[7] T. Blumensath and M. E. Davies. Iterative thresholding for sparse approximations. *Journal of Fourier Analysis and Applications*, 14(5-6):629–654, 2008.

[8] T. Blumensath and M. E. Davies. Iterative hard thresholding for compressed sensing. *Applied and Computational Harmonic Analysis*, 27(3):265–274, 2009.

[9] J. Bollen, H. Mao, and X. Zeng. Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1–8, 2011.

[10] R. Caruana. Multitask learning. *Machine Learning*, 28(1):41–75, 1997.

[11] CDC. Fluview interactive. Accessed May 31, 2015. http://www.cdc.gov/flu/weekly/fluviewinteractive.htm.

[12] J. Chen, J. Liu, and J. Ye. Learning incoherent sparse and low-rank patterns from multiple tasks. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 5(4):22, 2012.

[13] M. Dredze, M. J. Paul, S. Bergsma, and H. Tran. Carmen: A Twitter geolocation system with applications to public health. In *AAAI Workshop on Expanding the Boundaries of Health Informatics Using AI (HIAI)*, pages 20–24. Citeseer, 2013.

[14] T. Evgeniou and M. Pontil. Regularized multi-task learning. In *KDD 2004*, pages 109–117. ACM, 2004.

[15] P. Gong, C. Zhang, Z. Lu, J. Z. Huang, and J. Ye. A general iterative shrinkage and thresholding algorithm for non-convex regularized optimization problems. In *ICML*, volume 28, page 37. NIH Public Access, 2013.

[16] Z. Guan, S. Yang, H. Sun, M. Srivatsa, and X. Yan. Fine-grained knowledge sharing in collaborative environments. *IEEE Transactions on Knowledge and Data Engineering*, 27(8):2163–2174, 2015.

[17] S. Ji and J. Ye. An accelerated gradient method for trace norm minimization. In *ICML 2009*, pages 457–464. ACM, 2009.

[18] T. Lappas, M. R. Vieira, D. Gunopulos, and V. J. Tsotras. On the spatiotemporal burstiness of terms. *VLDB*, 5(9):836–847, 2012.

[19] MITRE. http://www.mitre.org/.

[20] B. O'Connor, R. Balasubramanyan, B. R. Routledge, and N. A. Smith. From tweets to polls: Linking text sentiment to public opinion time series. *ICWSM*, 11:122–129, 2010.

[21] N. Ramakrishnan, P. Butler, S. Muthiah, et al. 'beating the news' with embers: forecasting civil unrest using open source indicators. In *KDD 2014*, pages 1799–1808. ACM, 2014.

[22] J. Ritterman, M. Osborne, and E. Klein. Using prediction markets and Twitter to predict a swine flu pandemic. In *1st international workshop on mining social media*, volume 9, 2009.

[23] T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake shakes Twitter users: real-time event detection by social sensors. In *WWW*, pages 851–860, 2010.

[24] A. Signorini, A. M. Segre, and P. M. Polgreen. The use of Twitter to track levels of disease activity and public concern in the us during the influenza an H1N1 pandemic. *PLoS One*, 6(5):e19467, 2011.

[25] S. Tan, Y. Li, H. Sun, Z. Guan, X. Yan, J. Bu, C. Chen, and X. He. Interpreting the public sentiment variations on twitter. *ieee transactions on knowledge and data engineering*, 26(5):1158–1170, 2014.

[26] S. Thrun and J. O'Sullivan. Clustering learning tasks and the selective cross-task transfer of knowledge. *Learning to Learn*, pages 181–209, 1998.

[27] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[28] X. Wang, M. S. Gerber, and D. E. Brown. Automatic crime prediction using events extracted from Twitter posts. In *Social Computing, Behavioral-Cultural Modeling and Prediction*, pages 231–238. Springer, 2012.

[29] J. Weng and B.-S. Lee. Event detection in Twitter. *ICWSM*, 11:401–408, 2011.

[30] S. Xiang, T. Yang, and J. Ye. Simultaneous feature and feature group selection through hard thresholding. In *KDD 2014*, pages 532–541. ACM, 2014.

[31] J. Zhang, Z. Fang, W. Chen, and J. Tang. Diffusion of "following" links in microblogging networks. *IEEE Transactions on Knowledge and Data Engineering*, 27(8):2093–2106, 2015.

[32] L. Zhao, F. Chen, J. Dai, T. Hua, C.-T. Lu, and N. Ramakrishnan. Unsupervised spatial event detection in targeted domains with applications to civil unrest modeling. *PLoS One*, 9(10):e110206, 2014.

[33] L. Zhao, F. Chen, C.-T. Lu, and N. Ramakrishnan. Dynamic theme tracking in twitter. In *Big Data (Big Data), 2015 IEEE International Conference on*, pages 561–570. IEEE, 2015.

[34] L. Zhao, F. Chen, C.-T. Lu, and N. Ramakrishnan. Spatiotemporal event forecasting in social media. In *SDM 15*, pages 963–971. SIAM, 2015.

[35] L. Zhao, J. Chen, F. Chen, W. Wang, C.-T. Lu, and N. Ramakrishnan. Simnest: Social media nested epidemic simulation via online semi-supervised deep learning. In *ICDM 2015*, pages 639–648. IEEE, 2015.

[36] J. Zhou, J. Chen, and J. Ye. *MALSAR: Multi-tAsk Learning via StructurAl Regularization*. Arizona State University, 2011.

Liang Zhao is an Assistant Professor at Information Science and Technology Department of George Mason University. He received the Ph.D. degree from Virginia Tech, USA. His research interests include natural language processing, text mining, machine learning, and robotics. In recent years, he is interested in applications to social media, civil unrests, and public health informatics.

Qian Sun is a fourth-year Ph.D candidate majoring in computer science, and her supervisor is Dr. Jieping Ye. She obtained her Bachelor's degree in Electrical Engineering in Nanjing University of Aeronautics and Astronautics (NUAA) in 2008. Her Research mainly focus on developing novel transfer learning algorithms, as well as the applications on image applications.

Jieping Ye is an Associate Professor of Computer Science and Engineering at Arizona State University. He received his Ph.D. in Computer Science from University of Minnesota, Twin Cities in 2005. His research interests include machine learning, data mining, and biomedical informatics. He won the outstanding student paper award at ICML in 2004, the SCI Young Investigator of the Year Award at ASU in 2007.

Feng Chen received the B.S. degree from Hunan University, Changsha, China, in 2001; the M.S. degree from Beihang University, Beijing, China, in 2004; and the Ph.D. degree from Virginia Tech USA, in 2012, all in computer science. He is an Assistant Professor with the University at Albany, SUNY. His research focuses on the detection of emerging events and other relevant patterns in the mobile context.

Chang-Tien Lu received the M.S. degree in computer science from the Georgia Institute of Technology in 1996 and the Ph.D. degree in computer science from the University of Minnesota in 2001. He is an Associate Professor with the Department of Computer Science, Virginia Tech. His research interests include spatial databases, data mining, geographic information systems, and intelligent transportation systems.

Naren Ramakrishnan received the PhD degree in computer sciences from Purdue University, West Lafayette, IN, in 1997. He is currently the Thomas L. Phillips Professor of Engineering in the Department of Computer Science at Virginia Tech, Blacksburg, VA. His research has been supported by NSF, DHS, NIH, NEH, DARPA, IARPA, ONR, General Motors, HP Labs, NEC Labs, and Advance Auto Parts.