Feature Driven Learning Framework for Cybersecurity Event Detection

Taoran Ji[1,2], Xuchao Zhang[1,2], Nathan Self[1,2], Kaiqun Fu[1,2], Chang-Tien Lu[1,2], and Naren Ramakrishnan[1,2]
[1]Discovery Analytics Center, Virginia Tech, Arlington, VA 22203, USA
[2]Department of Computer Science, Virginia Tech, Arlington, VA 22203, USA

***Abstract*—Cybersecurity event detection is a crucial problem for mitigating effects on various aspects of society. Social media has become a notable source of indicators for detection of diverse events. Though previous social media based strategies for cybersecurity event detection focus on mining certain event-related words, the dynamic and evolving nature of online discourse limits the performance of these approaches. Further, because these are typically unsupervised or weakly supervised learning strategies, they do not perform well in an environment of biased samples, noisy context, and informal language which is routine for online, user-generated content. This paper takes a supervised learning approach by proposing a novel multi-task learning based model. Our model can handle diverse structures in feature space by learning models for different types of potential high-profile targets simultaneously. For parameter optimization, we develop an efficient algorithm based on the alternating direction method of multipliers. Through extensive experiments on a real world Twitter dataset, we demonstrate that our approach consistently outperforms existing methods at encoding and identifying cybersecurity incidents.**

I. Introduction

Cybersecurity incidents have garnered increased attention from the public due to their potentially tremendous impacts on society. For example, several high profile security events happened in recent years including Equifax data breach which exposed the vital information of 143 million people and Yahoo data breach which impacted 3 billion user accounts. In addition to these, there are countless attempted and successful account hijackings aimed at personal and corporate social media accounts. Traditionally, the problem of cyberattack detection is framed as anomaly detection at the network level. For example, Davis et al. [1] and Kwon et al. [2] leverage network tarffic data to detect certain types of attacks. However, this kind of cyberattack detection approach is difficult to generalize across different types of cyberattack, and the requisite data is usually expensive to obtain. Based on the intuition that information about organizational compromise can originate outside the organization, we look to open source indicators (e.g., Twitter messages), which is argued carrying rich security-related discussions for ongoing cyberattacks [3], as a source of information for cyberattack detection.

The motivation for this research is thus to mitigate the impacts of cyberattacks by detecting and identifying their occurrences at an early stage using data from social media. Open source indicators have been used as real-time "sensors" for detecting and forecasting social incidents such as civil unrest [4], [5], disease outbreaks [6], and elections [7]. Most current work is focused on unsupervised or weakly supervised learning methods which address the problem by mining keywords specific to a type of cyberattack. For instance, Ritter et al. [3] propose a weakly supervised method to capture cyberattack events by training with annotated samples from Twitter and fixed contextual feature sets. Khandpur et al. [8] leverage the dynamically evolving nature of cyberattacks with an unsupervised method to identify tweets related to cybersecurity incidents via queries dynamically expanded from a set of fixed seed queries.

These approaches suffer from the following challenges. (1) **The sparsity of cyberattack features.** Among all the linguistic clues hidden in a tweet's content, only a small portion of these "particles" play the crucial role of hinting at ongoing cybersecurity incidents. (2) **The ability to capture weak signals.** Events like the hijacking of individual social media accounts or data leakage from small organizations typically cause only a small range of discussion on Twitter. Such weak signals are often missed by unsupervised and weakly supervised learning methods. (3) **Generalization of models for different kinds of security events.** Previous studies which analyze network data are hard to generalize because different types of cyberattack imply different mechanisms of detection. On the other hand, existing methods which use social media data are typically designed to identify a specific type of incident by identifying linguistic clues related to a particular type of malicious activities such as DDoS attacks, data breaches, or account hijackings. As a result, this class of methods poorly suited for generalizing to all types of cyberattack. (4) **Insufficiently exploiting model-wise relatedness in learning models.** Most of the signal from Twitter data is generated by victims. As a result, we argue that the critical information from tweets mostly relates to the consequences of cyberattack. This information can involve different vocabulary across domains. For instance, a cyberattack on a retail business or e-commerce website is likely to involve credit/debit card information, different from an attack

ASONAM '19, August 27–30, 2019, Vancouver, BC, Canada

ACM ISBN 978-1-4503-6868-1/19/08…$15.00
http://doi.org/10.1145/3341161.3342871

on medical service providers which might expose patients' health information. Alternatively, the same vocabulary may be shared across distinct types of cyberattack. For instance, "*hacked*" could refer to an attempt to steal information from a government organization or it could refer to a DDoS attack on an internet content provider.

To address all these technical challenges, this paper presents a novel supervised learning model: the **multi-type feature-block regularized multi-task learning model (MFBR)**. MFBR treats detection of each type of incident as a task and can handle multiple types of relatedness among tasks in the problem space. Thus, it better characterizes the feature space for detection of cyberattacks. The main contributions of this research are as follows:

- **Formulating a multi-task learning framework for cyberattack event detection.** Different from existing work, we formulate the problem of detecting different kinds of security events as a multi-task supervised learning problem. In the proposed method, models for different types of security events are jointly optimized and are enhanced by satisfying constraints which describe different types of relatedness across tasks.
- **Modeling multi-type task relatedness in feature space.** Based on the thorough analysis of messages related to cybersecurity incidents on social media, we exploit three types of task relatedness in order to guide models to share knowledge across tasks while avoiding negative knowledge transfer. These three types of relatedness are governed by regularizers and constraints and cover task-wise shared feature learning, task characteristic feature learning, and task-wise variably overlapping feature learning.
- **Developing an efficient algorithm to solve the proposed model.** The optimization of the proposed multi-task model is a non-smooth, multi-convex, inequality-constrained problem which is challenging to solve. By introducing auxiliary variables, we design an efficient, converging algorithm which decouples the original problem into several simpler optimization sub-problems using ADMM.
- **Conducting extensive experiments to validate the effectiveness and efficiency of the proposed method.** The proposed model was evaluated on a dataset collected from Twitter from August 2014 to October 2016. For comparison, we implemented a broad range of state-of-the-art methods including DTQE, LR, LASSO, MTL-LASSO, RMFTL, and MTL-DM. The results demonstrate that the proposed model consistently outperforms the best of the existing methods along multiple metrics.

II. Related Work

Cyberattack Detection via Network Data Analysis. A large body of work focuses on constructing appropriate graph representations to analyze network traffic data. The approach frames the detection of different types of malicious behavior (intrusion detection, malicious server detection, fraud detection, etc.) as a problem of anomaly detection across the graph model [1], [2], [9], [10]. More recently, researchers sought to explore the possibility of proactively characterizing and forecasting malicious activities by leveraging network and associated traffic flow information [11]. These methods usually mine signals from vulnerable misconfigurations of a network (e.g. misconfigured DNS, BGP networks) or from reputation blacklists (e.g. malicious activities observed by spam traps). For instance, Liu et al. [12], [13] built and trained a random forest classifier to forecast whether a particular organization is, or will be, under attack. Soska et al. [14] collected historical records of malicious websites on blacklists and trained a classifier to predict whether a currently benign website will become malicious in the future.

Cyberattack Characterization via Social Media Analysis. In recent years, researchers have started making use of rich security-related information and discussions in online media such as blogs and Twitter [15], [16]. For instance, Tsai et al. [17] built a probabilistic model to analyze security information in tech weblogs to uncover ongoing cyberattacks. Liao et al. [18] focused on automatically mining and collecting critical indicators of compromise (e.g. malware signatures, botnet IPs) exchanged by security professionals through online media. Exploring and leveraging vulnerability-related information disseminated on Twitter, Sabottke et al. [19] constructed a classifier model to predict if a specific CVE vulnerability will be exploited in practice. Not until very recently did researchers start to use Twitter as a data source, because its broader user population promises richer information collection for ongoing cyberattacks from victims' perspective. Ritter et al. [3] cast the problem of cyberattack detection as a learning problem and design a weakly supervised learning model to identify and analyze security-related tweets. Khandpur et al. [8] design an unsupervised mining method which aims to capture the evolving nature of security-related discussions on Twitter and from there to detect ongoing cyberattack events.

Spatiotemporal Event Detection on Twitter. Our work is also related, in general, to event detection [20], [21] using Twitter. This covers various topics including natural disasters [22], criminal incidents [23], disease outbreaks [24], population migrations [25], trending news [26], [27], and activity planning [28]. One common method for event extraction is to use unsupervised learning models that work via keyword matching, clustering, and topic modeling [29]–[31]. Example applications are incidents detection of civil unrest events [4] and imminent threats to airports [32]. Also, researchers have used supervised learning models over social media data for stock market predictions [33], crime predictions [34] and civil unrest detection [4].

III. Problem Setup

Let $\mathcal{D} = \{\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_\mathcal{T}\}$ denote a collection of time-ordered tweets organized along $\mathcal{T}$ time slots. Let $\mathcal{V}$ denote a set of different kinds of organizations of interest: businesses,

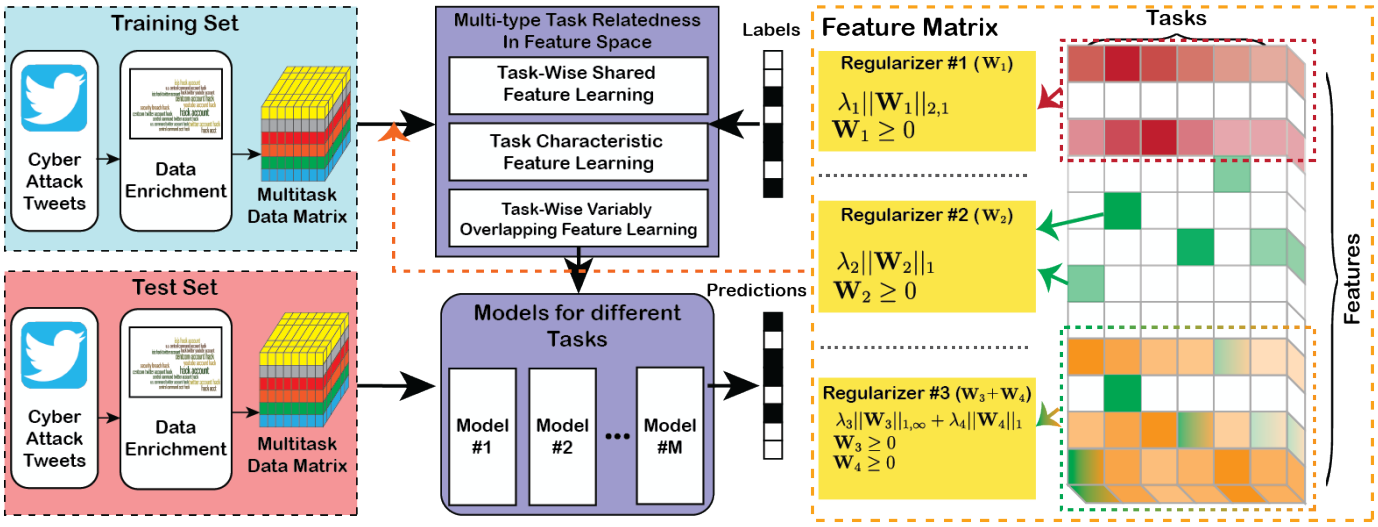


Fig. 1: A schematic view of the multi-type feature-block regularized multi-task learning model for cybersecurity incident detection (MFBR). In particular, regularizer #1 encourages models to select critical features (rows with red cells) shared across tasks while ignoring noisy features (rows with white cells). Regularizer #2 guides models to learn the weights for features (green cells) which can characterize incidents for a certain task. Regularizer #3 illustrates the learning of features that overlap variably across tasks by considering task-wise shared features and element-wise features together.

educational institutions, government agencies, hospitals, individual social media accounts, etc. Each tweet subcollection $\mathcal{D}_t, t = 1, \ldots, \mathcal{T}$ is then further divided into $|\mathcal{V}|$ subsets $\mathcal{D}_t^j, j = 1, \ldots, |\mathcal{V}|$, where each subset $\mathcal{D}_t^j$ refers to all the tweets in $\mathcal{D}_t$ related to the j-th organization type in $\mathcal{V}$.

Let $\mathcal{F}$ denote a set of keywords that are relevant to cyberattack topics, provided by domain experts [8]. For each subset of tweets $\mathcal{D}^j$, we define a matrix $\mathbf{X}^j \in \mathbb{R}^{|\mathcal{T}| \times |\mathcal{F}|}$. In each matrix, the (t, k)-th entry refers to the frequency of the k-th term of $\mathcal{F}$ in Twitter subcollection $\mathcal{D}_t^j$. By applying this to each Twitter subcollection $\mathcal{D}^j$, we obtain a data tensor $\mathbf{X} = \{\mathbf{X}^j\}, j = 1, 2, \ldots, |\mathcal{V}|$. For each data matrix $\mathbf{X}^j$, the corresponding response vector is denoted by $\mathbf{Y}^j$, where each element in $\mathbf{Y}^j$ is a binary variable. Then the corresponding response matrix for $\mathbf{X}$ is denoted by a $|\mathcal{F}| \times |\mathcal{V}|$ matrix $\mathbf{Y}$.

Our problem is as follows: given a type of potential cyberattack target $j \in \mathcal{V}$, a time slot $t \in \mathcal{T}$, and the corresponding data vector $\mathbf{X}_t^j$, is there an ongoing cybersecurity event? Mathematically, the problem can be formulated as learning a function which maps $\mathbf{X}_t^j$ to $\mathbf{Y}_t^j$ for each type of organization $j \in \mathcal{V}$:

$$\mathrm{F}_j(\mathbf{X}_t^j) \to \mathbf{Y}_t^j. \tag{1}$$

The problem is challenging in three aspects: (1) Features $|\mathcal{F}|$ and data samples are within the same order of magnitude which implies that this is a high-dimensional setting and, therefore, likely to exhibit sparsity. Indeed, the sparsity is likely more severe due to Twitter's restrictive character count. (2) Cyberattacks on some types of organizations (e.g. government agencies, schools) are not widely discussed on Twitter. These weak signals are hard to capture. (3) The relatedness across different types of victim organizations varies in feature space and is too crucial to be neglected.

IV. Model

In this section, we propose a new model MFBR to address the challenges mentioned above. MFBR treats each type of organization as a task, and can make use of the different types of relatedness across distinct tasks simultaneously.

A. Multi-type task relatedness learning in feature space

Multi-task learning is especially useful when it is empowered by relatedness across tasks — information which would otherwise be lost by single task learning models. Furthermore, to address real-world problems, it is essential to handle all types of task relatedness in the feature space. This prompts us to propose the following model:

$$\begin{aligned} \underset{\mathbf{W}_d, d \in \Phi}{\operatorname{argmin}} \; & \frac{1}{|\mathcal{V}|} \sum_{j=1}^{|\mathcal{V}|} \sum_{i=1}^{m_j} \log \left(1 + \exp\{-[\mathbf{Y}^j]_i [\sum_{d \in \Phi} \mathbf{X}_d^j \mathbf{W}_d^j]_i\} \right) \\ & + \sum_{d \in \Phi} \lambda_d \mathcal{R}_d(\mathbf{W}_d) \\ \text{s.t.} \; & \mathbf{W}_{d \in \Phi} \geq 0, \end{aligned} \tag{2}$$

where m_j refers to number of samples for the j-th task, index operator $[\cdot]_i$ refers to the i-th element of the specified vector, $\Phi = \{\mathcal{F}_1, \ldots, \mathcal{F}_{|\Phi|}\}$ is a partition of feature space which groups features into non-empty and non-overlapping subsets, $\mathbf{W}_d$ and $\mathbf{X}_d$ are the corresponding block matrices in $\mathbf{W}$ and $\mathbf{X}$ for features $\mathcal{F}_d$, respectively, and $\mathcal{R}_d(\mathbf{W}_d)$ is a regularizer which models the task relatedness for features $\mathcal{F}_d$ across tasks. As for classification problems, the first term of the model is the logistic loss. We counteract the noise in our Twitter data

source with the inequality constraint by which we suppose that all features either positively related to or completely irrelevant to cybersecurity events.

B. Cybersecurity Event Detection Model

For this cybersecurity event detection problem, it is preferable to split the entire feature space $\mathcal{F}$ into three non-overlapping groups due to the different types of task relatedness shared across tasks:

- **Task-Wise Shared Features** The presence of features like "*take over*" and "*take down*" in a tweet does not necessitate an absolute occurrence of a cyberattack; rather, it acts as a possible clue to event recognition in the Twitter environment. For instance, the tweet, "*If Lizard Squad hacking group have **taken down** #PSN and #Xbox live, then I hope they get tracked and locked up.*" uses "*take down*" to report a hacking incident.
- **Task Characteristic Features** In our formulation, each task represents a type of organization (e.g., educational institution), and there are features unique to individual tasks which should not be shared across all tasks. For instance, "*patients*", "*phi*", and "*medical records*" are features related only to cyberattacks on healthcare and medical providers.
- **Task-Wise Variably Overlapping Features** There is a final set of features which are partially shared over tasks. For instance, the keyword "*phish*" is directly related to security events and is usually found in tweets talking about cyberattacks on educational institutions and medical service providers.

To address the above three types of task relatedness in feature space together, we reformulate the optimization problem given in Equation 2 as follows:

$$\begin{aligned}&\underset{\mathbf{W}_1,\mathbf{W}_2,\mathbf{W}_3,\mathbf{W}_4}{\operatorname{argmin}} \frac{1}{|\mathcal{V}|}\sum_{j=1}^{|\mathcal{V}|}\sum_{i=1}^{m_j}\log\left(1+\exp\{-[\mathbf{Y}^j]_i[\sum_d \mathbf{X}_d^j\mathbf{W}_d^j]_i\}\right)\\&+\lambda_1\|\mathbf{W}_1\|_{2,1}+\lambda_2\|\mathbf{W}_2\|_1+\lambda_3\|\mathbf{W}_3\|_{1,\infty}+\lambda_4\|\mathbf{W}_4\|_1\\&\text{s.t. } \mathbf{W}=[\mathbf{W}_1,\mathbf{W}_2,\mathbf{W}_3+\mathbf{W}_4]^{\mathrm{T}},\mathbf{W}_d\geq 0,d=1,\ldots,4,\end{aligned} \tag{3}$$

where $\mathbf{W}_1$ refers to the weight of task-wise shared features to be identified by $\ell_{2,1}$ norms, $\mathbf{W}_2$ refers to weight of task characteristic features enforced by ℓ_1 norms and $\mathbf{W}_3, \mathbf{W}_4$ together designate the weight of task-wise variably overlapping features jointly governed by ℓ_1 and $\ell_{1,\infty}$ norms. The above three regularizeried structures are illustrated by regularizer #1, regularizer #2 and regularizer #3 in Figure 1, respectively.

Several classic methods emerge as special cases of our model proposed in Equation 3. If we let $\Phi = \{\mathcal{F}\}$ and enforce the $\ell_{2,1}$ norm on all feature groups without inequality constraints, our proposed model is reduced to the constrained multi-task feature selection model [20]. If we let $\Phi = \{\mathcal{F}\}$ and apply the ℓ_1 norm on all feature groups in Φ, our MFBR is then reduced to LASSO [35]. When $\lambda_1 = \lambda_2 = 0$ and inequality constraints are removed, the proposed model is reduced to a dirty model [36].

V. Parameter Optimization

The objective function given in Equation 3 is a multi-convex, non-smooth, inequality-constrained problem and is hard to solve directly. One efficient way to solve this is to transform the original problem into the following, equivalent problem by introducing two sets of auxiliary variables: $\Theta_2 = \{\widetilde{\mathbf{W}}_d\}_{d=1}^4$ and $\Theta_3 = \{\widehat{\mathbf{W}}_d\}_{d=1}^4$.

$$\begin{aligned}&\underset{\Theta_1,\Theta_2,\Theta_3}{\operatorname{argmin}} \frac{1}{|\mathcal{V}|}\sum_{j=1}^{|\mathcal{V}|}\sum_{i=1}^{m_j}\log\left(1+\exp\{-[\mathbf{Y}^j]_i[\sum_d \mathbf{X}_d^j\mathbf{W}_d^j]_i\}\right)\\&+\lambda_1\|\widetilde{\mathbf{W}}_1\|_{2,1}+\lambda_2\|\widetilde{\mathbf{W}}_2\|_1+\lambda_3\|\widetilde{\mathbf{W}}_3\|_{1,\infty}+\lambda_4\|\widetilde{\mathbf{W}}_4\|_1\\&\text{s.t. } \mathbf{W}_d=\widetilde{\mathbf{W}}_d,\mathbf{W}_d=\widehat{\mathbf{W}}_d,\widehat{\mathbf{W}}_d\geq 0,d=1,\ldots,4,\end{aligned} \tag{4}$$

where $\Theta_1 = \{\mathbf{W}_d\}_{d=1}^4$ are the original variables.

We use the alternating direction method of multipliers (ADMM) [37] to decouple this problem into easier to handle sub-problems. After further reformulating it into *augmented Lagrangian* with penalty parameter ρ, we have:

$$\begin{aligned}&\underset{\Theta_1,\Theta_2,\Theta_3}{\operatorname{argmin}} \frac{1}{|\mathcal{V}|}\sum_{j=1}^{|\mathcal{V}|}\sum_{i=1}^{m_j}\log\left(1+\exp\{-[\mathbf{Y}^j]_i[\sum_d \mathbf{X}_d^j\mathbf{W}_d^j]_i\}\right)\\&+\lambda_1\|\widetilde{\mathbf{W}}_1\|_{2,1}+\lambda_2\|\widetilde{\mathbf{W}}_2\|_1+\lambda_3\|\widetilde{\mathbf{W}}_3\|_{1,\infty}+\lambda_4\|\widetilde{\mathbf{W}}_4\|_1\\&+\frac{\rho}{2}\sum_{d=1}^4\left(\|\mathbf{W}_d-\widetilde{\mathbf{W}}_d\|_2^2+\|\mathbf{W}_d-\widetilde{\mathbf{W}}_d\|_2^2\right)\\&+\sum_{d=1}^4\left(\langle\widetilde{\mathbf{U}}_d,\mathbf{W}_d-\widetilde{\mathbf{W}}_d\rangle+\langle\widehat{\mathbf{U}}_d,\mathbf{W}_d-\widehat{\mathbf{W}}_d\rangle\right)\end{aligned} \tag{5}$$

where $\widetilde{\mathbf{U}} = \{\widetilde{\mathbf{U}}_k\}_{k=1}^4$ and $\widehat{\mathbf{U}} = \{\widehat{\mathbf{U}}_k\}_{k=1}^4$ are Lagrangian multipliers. Next, all parameters Θ_1, Θ_2, Θ_3, $\widetilde{\mathbf{U}}$, and $\widehat{\mathbf{U}}$ are optimized alternatively, as described in the following sections, until convergence.

A. 1. Update Θ_1

Primal variables Θ_1 are updated by solving the following problem:

$$\begin{aligned}&\underset{\Theta_1}{\operatorname{argmin}} \frac{1}{|\mathcal{V}|}\sum_{j=1}^{|\mathcal{V}|}\sum_{i=1}^{m_j}\log\left(1+\exp\{-[\mathbf{Y}^j]_i[\sum_d \mathbf{X}_d^j\mathbf{W}_d^j]_i\}\right)\\&+\frac{\rho}{2}\sum_{d=1}^4\left(\|\mathbf{W}_d-\widetilde{\mathbf{W}}_d\|_2^2+\|\mathbf{W}_d-\widetilde{\mathbf{W}}_d\|_2^2\right)\\&+\sum_{d=1}^4\left(\langle\widetilde{\mathbf{U}}_d,\mathbf{W}_d-\widetilde{\mathbf{W}}_d\rangle+\langle\widehat{\mathbf{U}}_d,\mathbf{W}_d-\widehat{\mathbf{W}}_d\rangle\right)\end{aligned} \tag{6}$$

The problem as stated above is smooth and multi-convex (i.e. the objective function is convex on each variable in Θ_1 when the other three variables are fixed). This kind of problem can be solved by block coordinate descent (BCD) [38] which iteratively updates one variable while fixing the other

variables. Note that, when all but one variable is fixed, our objective function from Equation 6, denoted by $\mathcal{H}$, is smooth and convex, and thus can be optimized using gradient descent:

$$\frac{\partial \mathcal{H}}{\mathbf{W}_d} = \frac{1}{|\mathcal{V}|}(\mathbf{X}_d)^{\mathrm{T}}\mathbf{G} + \widetilde{\mathbf{U}}_d + \widehat{\mathbf{U}}_d + \rho(2\mathbf{W}_d - \widetilde{\mathbf{W}}_d - \widehat{\mathbf{W}}_d) \tag{7}$$

where

$$\mathbf{G} = -\mathbf{Y} \circ (\mathbf{I} - \mathbf{I} \oslash (\mathbf{I} + \exp\{\mathbf{Z}\})),$$
$$\mathbf{Z} = -\mathbf{Y} \circ (\mathbf{X}_1\mathbf{W}_1 + \mathbf{X}_2\mathbf{W}_2 + \mathbf{X}_3(\mathbf{W}_3 + \mathbf{W}_4)),$$

and $\circ$ is the element-wise product (Hadamard product), $\oslash$ is element-wise division (Hadamard division), and $\mathbf{I}$ is the identity vector.

B. 2. Update Θ_2 and Θ_3

Dual variables in Θ_2 are updated by solving optimization sub-problems:

$$\underset{\widetilde{\mathbf{W}}_d}{\operatorname{argmin}} \mathcal{R}(\widetilde{\mathbf{W}}_d) + \langle \widetilde{\mathbf{U}}_d, \mathbf{W}_d - \widetilde{\mathbf{W}}_d \rangle + \frac{\rho}{2}\|\mathbf{W}_d - \widetilde{\mathbf{W}}_d\|_2^2. \tag{8}$$

To solve the above problem, we reformulated it to an equivalent proximal operator:

$$\widetilde{\mathbf{W}}_d^+ \leftarrow \mathrm{prox}_{\rho^{-1}f_d}(\widetilde{\mathbf{U}}_d + \mathbf{W}_d), \tag{9}$$

where

$$\begin{aligned} f_1(\widetilde{\mathbf{W}}_1) &= \lambda_1\|\widetilde{\mathbf{W}}_1\|_{2,1}, \quad f_2(\widetilde{\mathbf{W}}_2) = \lambda_2\|\widetilde{\mathbf{W}}_2\|_1, \\ f_3(\widetilde{\mathbf{W}}_3) &= \lambda_3\|\widetilde{\mathbf{W}}_3\|_{1,\infty}, \quad f_4(\widetilde{\mathbf{W}}_4) = \lambda_4\|\widetilde{\mathbf{W}}_4\|_1. \end{aligned} \tag{10}$$

For each dual variable in Θ_3, the following optimization problem needs to be solved:

$$\widehat{\mathbf{W}}_d^+ \leftarrow \underset{\widehat{\mathbf{W}}_d \geq 0}{\operatorname{argmin}} \langle \widehat{\mathbf{U}}_d, \mathbf{W}_d - \widehat{\mathbf{W}}_d \rangle + \frac{\rho}{2}\|\mathbf{W}_d - \widehat{\mathbf{W}}_d\|_2^2 \tag{11}$$

Then, for each dual variable in Θ_3, we have

$$\widehat{\mathbf{W}}_d^+ \leftarrow \max(\mathbf{W}_d + \frac{\widehat{\mathbf{U}}_d}{\rho}, 0). \tag{12}$$

C. 3. Update $\widetilde{\mathbf{U}}$ and $\widehat{\mathbf{U}}$

Lagrangian multipliers are updated as follows:

$$\widetilde{\mathbf{U}}_d^+ \leftarrow \widetilde{\mathbf{U}}_d + \rho(\mathbf{W}_d^+ - \widetilde{\mathbf{W}}_d^+), \widehat{\mathbf{U}}_d^+ \leftarrow \widehat{\mathbf{U}}_d + \rho(\mathbf{W}_d^+ - \widehat{\mathbf{W}}_d^+). \tag{13}$$

Finally, primal and dual residuals are computed with

$$\begin{aligned} r &= \sum_{d=1}^{4} \left(\|\mathbf{W}_d - \widetilde{\mathbf{W}}_d\|_2 + \|\mathbf{W}_d - \widehat{\mathbf{W}}_d\|_2 \right), \\ s &= \rho \sum_{d=1}^{4} \|\widetilde{\mathbf{W}}_d^+ - \widetilde{\mathbf{W}}_d + \widehat{\mathbf{W}}_d^+ - \widehat{\mathbf{W}}_d\|_2. \end{aligned} \tag{14}$$

Pseudocode for the algorithm is given in Algorithm 1. In particular, line 4 updates the primal variables Θ_1. Lines 5 and 6 update the dual variables Θ_2 and Θ_3. Lagrangian multipliers are updated in line 7. Line 8 calculates both primal and dual residuals. Lines 9 to 11 check the stop criterion.

Algorithm 1: Parameter Optimization for MFBR

```
1  Input: X, Y, Output: W;
2  Initialize ρ = 1, Θ1, Θ2, Θ3, ε^r, ε^s, MAX_ITER;
3  for k = 1 : MAX_ITER do
4      Update primal variables in Θ1 using 7;
5      Update dual variables in Θ2 using 9;
6      Update dual variables in Θ3 using 12;
7      Update Lagrangian multipliers using 13;
8      Compute r and s using 14;
9      if r ≤ ε^r and s ≤ ε^s then
10         break;
11     end
12 end
```

VI. Evaluation

A. Ground Truth and Dataset

Gold Standard Report (GSR) Collection. For evaluating the performance of our proposed method and for comparing it to existing approaches, we compiled a ground truth database which we call the gold standard report (GSR). We compiled the GSR from two different sources:

- **Privacy Rights Clearinghouse (PRC)** [1] is an independently maintained collection of reports about cybersecurity incidents organized by victim organization type (businesses, educational institutions, medical service providers, government agencies, etc.). We extracted 1,064 cybersecurity incidents from January 2014 to December 2016. After removing incidents that did not occur in the United States of America (due to the concern of tweet language) and that do not fall within the time range of our Twitter dataset, we are left with 893 unique cybersecurity events.
- **Hackmageddon** [2] is another reputable collection of public reports about cybersecurity incidents. We extracted 1,307 cybersecurity incidents from January 2014 to November 2016. After filtering, again, on country and time range, we have 1,064 unique cybersecurity events from this collection.

In both databases, each event report comprises of an event type, date, victim organization(s), and a short description. Additionally, PRC publishes a type for each victim organization. To further organize and combine these two databases, we manually label the victim type for each event based on the definitions shown in Table I. These organization type definitions are based on the organization code and PRC's organization type. In particular, we make following revisions to PRC's schema: (1) We redefine BSR to be organizations which use customers' financial data (e.g. credit/debit card number, bank account number, etc.). PRC uses this label less specifically for retail or online businesses. (2) IND is a

[1] https://www.privacyrights.org/

[2] https://www.hackmageddon.com/

TABLE I: Definition of organization type.

ORG	Definition
BSF	Financial and (non-health) insurance businesses.
BSO	Businesses that do not need customers' sensitive financial data.
BSR	Business that use customers' payment method information such as credit card number, etc.
EDU	Educational institutions.
GOV	Government and military.
IND	Individual social media (e.g. Twitter). accounts.
MED	Medical providers and medical insurance services.

newly defined target type which refers to individual social media accounts on platforms such as Twitter, Facebook, and Vine. After merging the two databases and removing duplicate records, our GSR contains 1,510 unique incidents from August 2014 to October 2016.

Twitter Dataset and Preprocessing. We evaluate the proposed method with a large stream of tweets from GNIP's decahose (a 10% sample of all tweets) collected from August 2014 through October 2016. The entire dataset was subjected to a preprocessing pipeline in which all retweets and non-English tweets were removed; enrichment, including tokenization and lemmatization, was performed using spaCy[3]; and, English stop words, words with diacritical marks, and user mentions were eliminated from the tweets. As a result, we obtain 4,975,992,550 enriched tweets from these 27 months as the final dataset. Then, to evaluate our model, we separate the dataset into two parts: (1) data from August 2014 to December 2015, which serves as the training set for supervised learning methods, and (2) data from January 2016 to October 2016, which serves as the test set for comparison of our method against other methods (including a state-of-the-art unsupervised learning method). Each of the two subcollections are partitioned into a sequence of one-day-interval bins and labeled by named entities mentioned within that bin and the organization type corresponding to those entities.

B. Experiment Setup and Comparison Methods

All the models are validated and quantified based on four metrics. *Precision* designates the ratio of correctly detected events over all detected events. *Recall* denotes the percentage of all cybersecurity incidents that are actually recognized by the model. *F-measure* is the harmonic mean of precision and recall which is defined as $2 \cdot \text{precision} \cdot \text{recall}/(\text{precision} + \text{recall})$. *AUC* is the area under the receiver operating characteristic (ROC) curve and designates the model's classification ability as its discrimination threshold varies. A 10-fold cross validation was utilized on training set to examine each parameter from 0.1 to 1 with step size 0.1. The values with best performance on training set were selected. The following methods are included in the performance comparison:

[3]https://spacy.io/

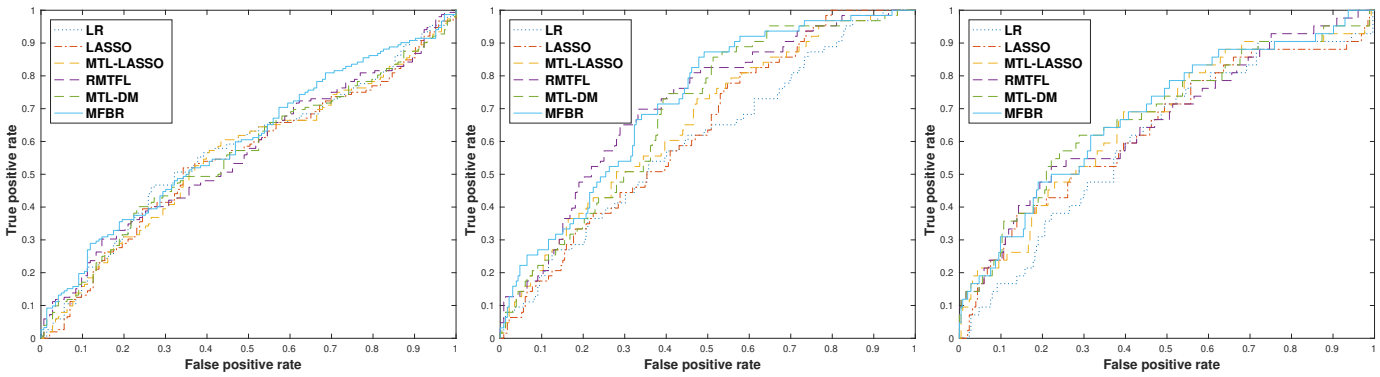
- **Dynamic Typed Query Expansion (DTQE) [8]**. DTQE is a state-of-the-art unsupervised learning method which works by identifying and extracting event-related tweets using syntactic structures expanded from a small set of seed event triggers. DTQE is performed on per-day tweet subcollections from the test set with event triggers provided in [8].
- **Logistic Regression (LR) [39]**. For each task, LR uses a logit function to predict the probability of the occurrence of a cybersecurity event based on per-day tweet subcollections. A corresponding data matrix is generated using feature counts and no tunable parameters are required.
- **LASSO with Logistic Loss (LASSO) [4]**. For each task, LASSO is trained on a logistic loss, regularized with ℓ_1-norm to control the feature sparsity. There is one tunable parameter and it is searched from 0.1 to 1 with a 0.1 step size. The data matrix is generated using feature counts.
- **Multi-Task LASSO with Logistic Loss (MTL-LASSO) [40]**. A multi-task learning based LASSO method which shares one penalty parameter for controlling sparsity across all tasks. The data matrix is generated using feature counts and the only tunable parameter is searched from 0.1 to 1 with a 0.1 step size.
- **Regularized Multi-Task Feature Learning Model with Logistic Loss (RMTFL) [20]**. We replaced the least squares loss with logistic loss to fit our proposed classification problem. The data matrix is generated using feature counts and the only tunable parameter is searched from 0.1 to 1 with a 0.1 step size.
- **A Dirty Model for Multi-Task Learning with Logistic Loss (MTL-DM) [36]**. This uses logistic loss instead of least squares loss. The data matrix is generated using feature counts and the two tunable parameters are searched from 0.1 to 1 with a 0.1 step size.

C. Measuring Performance

Detection Performance on Precision, Recall and F-measure. Table II presents the comparison of our proposed MFBR method with the competing techniques. Due to space limitations, only 4 out of 7 tasks are reported; the performance on the other 3 tasks, BSF, GOV and EDU are similar to the 4 tasks shown. We quantify each method's performance with precision, recall, and F-measure. First of all, though DTQE gets a relative high precision, its recall and F-measure are much lower compared to the supervised learning methods, which justifies our motivation of designing a supervised learning method for cybersecurity event detection. Also, Table II shows that, in general, the performance of regularized methods like LASSO, MTL-LASSO, RMTFL, MTL-DM, MFBR consistently surpasses the non-regularized method, LR, by 3% to 17% on F-measure. This demonstrates that a sparsity structure exists in cyberattack feature space and that the regularizers used in all of these models contribute to filtering out unrelated features and ensuring the model's generalizability.

TABLE II: Cybersecurity Events Detection Performance Comparison (Precision, Recall, F-measure, AUC)

Method	BSO	BSR	MED	IND
DTQE	**0.63**, 0.10, 0.17, NA	**0.75**, 0.09, 0.16, NA	0.33, 0.01, 0.01, NAJ	**0.47**, 0.19, 0.27, NA
LR	0.51, 0.80, 0.62, 0.57	0.29, 0.54, 0.38, 0.60	0.58, 0.37, 0.46, 0.56	0.22, 0.38, 0.28, 0.61
LASSO	0.52, 0.97, 0.67, 0.55	0.28, 0.81, 0.41, 0.63	0.58, 0.48, 0.52, 0.56	0.31, 0.38, 0.34, 0.64
MTL-LASSO	0.51, 0.99, **0.68**, 0.56	0.29, 0.73, 0.42, 0.66	0.61, 0.54, 0.57, 0.57	0.26, 0.48, 0.33, 0.66
RMTFL	0.52, 0.99, **0.68**, 0.56	0.38, 0.65, **0.48**, 0.71	0.52, **0.81**, **0.63**, 0.54	0.28, 0.52, 0.37, 0.66
MTL-DM	0.52, **1.00**, **0.68**, 0.56	0.33, 0.75, 0.46, 0.69	**0.64**, 0.59, 0.61, **0.60**	0.29, 0.55, 0.38, 0.68
MFBR	0.52, 0.99, **0.68**, **0.60**	0.33, **0.87**, 0.47, **0.72**	0.55, 0.68, 0.61, 0.58	0.31, **0.67**, **0.42**, **0.70**



(a) BSO (b) BSR (c) IND

Fig. 2: Receiver operating characteristic (ROC) curves for the performances on different tasks.

In addition, Table II shows that, in general, the multi-task based methods have better performance than single task models and that, for specific tasks, the performance of the multi-task models varies. This demonstrates the benefits of leveraging task relatedness and also suggests that, because there are many types of relatedness in feature space, arbitrarily modeling the problem with one kind of structure may increase or decrease performance on specific tasks. This is because multi-task models make use of knowledge across tasks to improve the performance on the single task. If a wrong structure is assumed, the model is misled into learning a negative knowledge transfer across tasks. For example, MTL-LASSO gains 5% improvement in F-measure against LASSO for MED identification while its performance on the IND task is diminished. On the other hand, MTL-DM, which encourages the model to learn a balanced mixed sparsity structure, is outperformed by RMTFL on the BSR task.

Furthermore, our proposed MFBR model can, in general, achieve the best performance among multi-task learning methods on all tasks. Specifically, it avoids the negative knowledge transfer of RMTFL in IND, and MTL-LASSO in BSR, and increases performance in IND by 4% to 9% against other multi-task based methods. This again demonstrates our motivation for modeling multi-type task relatedness in feature space to avoid negative knowledge transfer in a multi-task setting. Also, during the experiments, we observed that the training time of MFBR is slighly slower (40 - 50 seconds) than other multi-task learning methods. This is expected because MFBR considers multiple types task relatedness in the model which increase the algorithm complexity and result in a multi-convex, non-smooth and inequality-constrained problem.

Detection performance on ROC curves. The area under the ROC curve (AUC) for four tasks is reported in Table II. Figure 2 illustrates the performance of models on ROC curves for three tasks. The ROC curve for the omitted other tasks follows a similar pattern and is not reported here due to space limitations. The curves are drawn by plotting the true positive rate (TPR) against the false positive rate (FPR) at varying cutoff points for positive and negative predictions. First of all, we observe that, in general, the non-regularized method, LR, has the worst performance for every organization type. Figure 2 also shows that the multi-task model's performance varies on the task. For instance, RMTFL performs better on BSR than on IND. However, our model consistently has the best performance among multi-task models and gains a significant boost against the single task model LASSO.

VII. Conclusion

We have demonstrated the effectiveness of our multi-task based supervised learning model for cybersecurity detection using social media data. Our work considers structures in feature space that are exclusive to this application, which are leveraged by means of a block-sparsity regularizer for features shared across tasks, an element-wise regularizer for features characteristic of a particular task, and a regularizer for variably

overlapping features. We also propose an efficient ADMM-based algorithm to decouple the original, complex problem into several easier-to-handle sub-problems which are solved by block coordinate descent and proximal operators. Our empirical results demonstrate that our proposed model can effectively benefit from modeling multi-type sparsity structures and can achieve the best performance among both our multi-task based comparison methods and a single task model on all tasks.

REFERENCES

[1] M. Davis, W. Liu, P. Miller, and G. Redpath, "Detecting anomalies in graphs with numeric labels," in *Proc. CIKM'11*, 2011.

[2] B. J. Kwon, J. Mondal, J. Jang, L. Bilge, and T. Dumitras, "The dropper effect: Insights into malware distribution with downloader graph analytics," in *Proc. CCS'15*, 2015.

[3] A. Ritter, E. Wright, W. Casey, and T. Mitchell, "Weakly supervised extraction of computer security events from twitter," in *Proc. WWW'15*, 2015.

[4] N. Ramakrishnan, P. Butler, S. Muthiah, N. Self, R. Khandpur, P. Saraf, W. Wang, J. Cadena, A. Vullikanti, G. Korkmaz *et al.*, "'beating the news' with embers: forecasting civil unrest using open source indicators," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2014, pp. 1799–1808.

[5] 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*, vol. 9, no. 10, p. e110206, 2014.

[6] 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 a h1n1 pandemic," *PloS one*, vol. 6, no. 5, p. e19467, 2011.

[7] A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welpe, "Predicting elections with twitter: What 140 characters reveal about political sentiment." *Icwsm*, vol. 10, no. 1, pp. 178–185, 2010.

[8] R. P. Khandpur, T. Ji, S. Jan, G. Wang, C.-T. Lu, and N. Ramakrishnan, "Crowdsourcing cybersecurity: Cyber attack detection using social media," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, ser. CIKM '17. New York, NY, USA: ACM, 2017, pp. 1049–1057. [Online]. Available: http://doi.acm.org/10.1145/3132847.3132866

[9] Q. Ding, N. Katenka, P. Barford, E. Kolaczyk, and M. Crovella, "Intrusion as (anti)social communication: Characterization and detection," in *Proc. KDD'12*, 2012.

[10] C. C. Noble and D. J. Cook, "Graph-based anomaly detection," in *Proc. KDD'03*, 2003.

[11] F. Li, Z. Durumeric, J. Czyz, M. Karami, M. Bailey, D. McCoy, S. Savage, and V. Paxson, "You've got vulnerability: Exploring effective vulnerability notifications," in *Proc. USENIX Sec'16*, 2016.

[12] Y. Liu, A. Sarabi, J. Zhang, P. Naghizadeh, M. Karir, M. Bailey, and M. Liu, "Cloudy with a chance of breach: Forecasting cyber security incidents," in *Proc. USENIX Sec'15*, 2015.

[13] Y. Liu, J. Zhang, A. Sarabi, M. Liu, M. Karir, and M. Bailey, "Predicting cyber security incidents using feature-based characterization of network-level malicious activities," in *Proc. IWSPA'15*, 2015.

[14] K. Soska and N. Christin, "Automatically detecting vulnerable websites before they turn malicious," in *Proc. USENIX Sec'14*, 2014.

[15] A. Modi, Z. Sun, A. Panwar, T. Khairnar, Z. Zhao, A. Doupé, G. J. Ahn, and P. Black, "Towards automated threat intelligence fusion," in *Proc. IEEE CIC'16*, 2016.

[16] D. J. Weller-Fahy, "Towards finding malicious cyber discussions in social media," in *Proc. AICS'17*, 2017.

[17] F. S. Tsai and K. L. Chan, "Detecting cyber security threats in weblogs using probabilistic models," in *Proc. PAISI'07*, 2007.

[18] X. Liao, K. Yuan, X. Wang, Z. Li, L. Xing, and R. Beyah, "Acing the ioc game: Toward automatic discovery and analysis of open-source cyber threat intelligence," in *Proc. CCS'16*, 2016.

[19] C. Sabottke, O. Suciu, and T. Dumitras, "Vulnerability disclosure in the age of social media: Exploiting twitter for predicting real-world exploits," in *Proc. USENIX Sec'15*, 2015.

[20] L. Zhao, Q. Sun, J. Ye, F. Chen, C.-T. Lu, and N. Ramakrishnan, "Multi-task learning for spatio-temporal event forecasting," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2015, pp. 1503–1512.

[21] L. Zhao, F. Chen, C.-T. Lu, and N. Ramakrishnan, "Multi-resolution spatial event forecasting in social media," in *Data Mining (ICDM), 2016 IEEE 16th International Conference on*. IEEE, 2016, pp. 689–698.

[22] T. Sakaki, M. Okazaki, and Y. Matsuo, "Earthquake shakes twitter users: real-time event detection by social sensors," in *Proc. WWW'10*, 2010.

[23] X. Wang, M. S. Gerber, and D. E. Brown, "Automatic crime prediction using events extracted from twitter posts," in *Proc. SBP'12*, 2012.

[24] 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 *Data Mining (ICDM), 2015 IEEE International Conference on*. IEEE, 2015, pp. 639–648.

[25] J. Piskorski, H. Tanev, and A. Balahur, "Exploiting twitter for border security-related intelligence gathering," in *Proc. EISIC'13*, 2013.

[26] F. Atefeh and W. Khreich, "A survey of techniques for event detection in twitter," *Comput. Intell.*, vol. 31, no. 1, pp. 132–164, 2015.

[27] A. Ritter, Mausam, O. Etzioni, and S. Clark, "Open domain event extraction from twitter," in *Proc. KDD'12*, 2012.

[28] H. Becker, D. Iter, M. Naaman, and L. Gravano, "Identifying content for planned events across social media sites," in *Proc. WSDM'12*, 2012.

[29] H. Tanev, M. Ehrmann, J. Piskorski, and V. Zavarella, "Enhancing event descriptions through twitter mining," in *Proc. ICWSM'14*, 2012.

[30] H. Becker, M. Naaman, and L. Gravano, "Beyond trending topics: Real-world event identification on twitter," in *Proc. ICWSM'14*, 2012.

[31] X. Zhou and L. Chen, "Event detection over twitter social media streams," *The VLDB Journal*, vol. 23, no. 3, pp. 381–400, 2014.

[32] R. P. Khandpur, T. Ji, Y. Ning, L. Zhao, C.-T. Lu, E. R. Smith, C. Adams, and N. Ramakrishnan, "Determining relative airport threats from news and social media," in *Proc. AAAI'16*, 2016.

[33] J. Bollen, H. Mao, and X. Zeng, "Twitter mood predicts the stock market," *Journal of computational science*, vol. 2, no. 1, pp. 1–8, 2011.

[34] X. Wang, M. S. Gerber, and D. E. Brown, "Automatic crime prediction using events extracted from twitter posts," in *International conference on social computing, behavioral-cultural modeling, and prediction*. Springer, 2012, pp. 231–238.

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

[36] A. Jalali, S. Sanghavi, C. Ruan, and P. K. Ravikumar, "A dirty model for multi-task learning," in *Advances in Neural Information Processing Systems*, 2010, pp. 964–972.

[37] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein *et al.*, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[38] Y. Xu and W. Yin, "A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion," *SIAM Journal on imaging sciences*, vol. 6, no. 3, pp. 1758–1789, 2013.

[39] R. Compton, C. Lee, J. Xu, L. Artieda-Moncada, T.-C. Lu, L. De Silva, and M. Macy, "Using publicly visible social media to build detailed forecasts of civil unrest," *Security informatics*, vol. 3, no. 1, p. 4, 2014.

[40] J. Zhou, J. Chen, and J. Ye, "Malsar: Multi-task learning via structural regularization," *Arizona State University*, vol. 21, 2011.