# Deep Multi-Task Learning for Spatio-Temporal Incomplete Qualitative Event Forecasting

Tanmoy Chowdhury, Yuyang Gao, and Liang Zhao

*Abstract*—**Forecasting spatiotemporal social events has significant benefits for society to provide the proper amounts and types of resources to manage catastrophes and any accompanying societal risks. Nevertheless, forecasting event subtypes are far more complex than merely extending binary prediction to cover multiple subtypes because of spatial heterogeneity, experiencing a partial set of event subtypes, subtle discrepancy among different event subtypes, nature of the event subtype, spatial correlation of event subtypes. We present Deep multi-task learning for spatio-temporal incomplete qualitative event forecasting (DETECTIVE) framework to effectively forecast the subtypes of future events by addressing all these issues. This formulates spatial locations into tasks to handle spatial heterogeneity in event subtypes and learns a joint deep representation of subtypes across tasks. This has the adaptability to be used for different types of problem formulation required by the nature of the events. Furthermore, based on the "first law of geography", spatially-closed tasks share similar event subtypes or scale patterns so that adjacent tasks can share knowledge effectively. To optimize the non-convex and strongly coupled problem of the proposed model, we also propose algorithms based on the Alternating Direction Method of Multipliers (ADMM). Extensive experiments on real-world datasets demonstrate the model's usefulness and efficiency.**

*Index Terms*—**Multi-task learning, optimization, softmax regression, ordinal regression, deep learning.**

## I. Introduction

Societal events happening at a particular time and location such as disease epidemics and organized crime, natural hazard events, environmental pollution events, and urban-related events have a significant impact on society. The capacity to correctly foresee future spatiotemporal occurrences of this type would thus be immensely advantageous for decision-makers aiming to prevent, manage, or mitigate the accompanying social turmoil and hazards. Spatiotemporal social event forecasting is a rapidly expanding research area that typically forecasts the *occurrence* of future spatial events, specifically whether or not a specific geographical event will occur. In many situations, however, just anticipating the occurrence of an event is insufficient. Understanding a potential event's subtype, category, magnitude, or degree is critical for providing correct and appropriate crisis management resources. For example, Fig. 1(a) shows the percentage of six pollutant subtypes that feature in air pollution events based on the most frequently detected primary pollutants in Shenzhen, China in Summer 2013 [1]. Local Environmental Monitoring Centers try to identify which pollutant source causing the most harm to public health and take appropriate action. For instance, when the pollutant subtype is $PM2.5$ (atmospheric particulate matter with a diameter less than 2.5 micrometers), the government can suggest that people who are sensitive to small particles wear gauze masks to protect themselves. On the other hand, when the subtype is $O_3$ (trioxygen), government agencies need to alert people to avoid going outside when the $O_3$ concentration is highest. Thus, successful forecasting of the pollutant subtypes provides more specific information that enables practitioners to allocate resources that will address public health issues with the specific primary pollutant source most effectively. In another example, as shown in Fig. 1(b), the Centers for Disease Control and Prevention (CDC) rank the severity of ongoing disease outbreaks using five scale points. The successful prediction of the scale of future disease outbreaks enables practitioners to allocate appropriate levels of resources for vaccination and isolation. Accurate forecasts of social events, especially localized ones, are crucial for authorities to plan resource allocation and responses. However, as yet little research has focused specifically on spatial social event scale forecasting.

The majority of prior work in this field, such as [2], [3] has concentrated on the event occurrence rather than the exploration of the various event subtypes. A few preliminary research [4], [5] tried to investigate this open subject using simple multi-class classification approaches. Moreover, standard event forecasting methods often anticipate a binary output (i.e., the event either occurs or it does not) and cannot be used to forecast event scales, which are ordinal variables.

Spatial event subtype forecasting is significantly more challenging than simply adapting a binary classification problem to a multi-class or multi-scale setting. *1) Event-subtype correlation and spatial heterogeneity:* Population, climate, and government policies vary by location, leading to demographic disparities among social media users. For example, the same number of 'flu' mentions on Twitter can indicate different influenza activity levels in California and Nebraska due to population differences. According to the "first law of geography" [6],

Received 6 April 2023; revised 21 June 2024; accepted 10 September 2024. Date of publication 13 September 2024; date of current version 13 November 2024. *(Tanmoy Chowdhury and Yuyang Gao contributed equally to this work.) (Corresponding author: Liang Zhao.)*

Tanmoy Chowdhury is with the Information Technology, Richland County Government, Columbia, SC 29204 USA (e-mail: tchowdh6@gmu.edu).

Yuyang Gao is with the The Home Depot, Atlanta, Georgia 30339 USA (e-mail: yuyang_gao@homedepot.com).

Liang Zhao is with the Department of Computer Science and Informatics, Atlanta, Georgia 30322 USA (e-mail: liang.zhao@emory.edu).

Source Code: https://github.com/tanmoysr/DETECTIVE.

This article has supplementary downloadable material available at https://doi.org/10.1109/TKDE.2024.3460539, provided by the authors.

Digital Object Identifier 10.1109/TKDE.2024.3460539

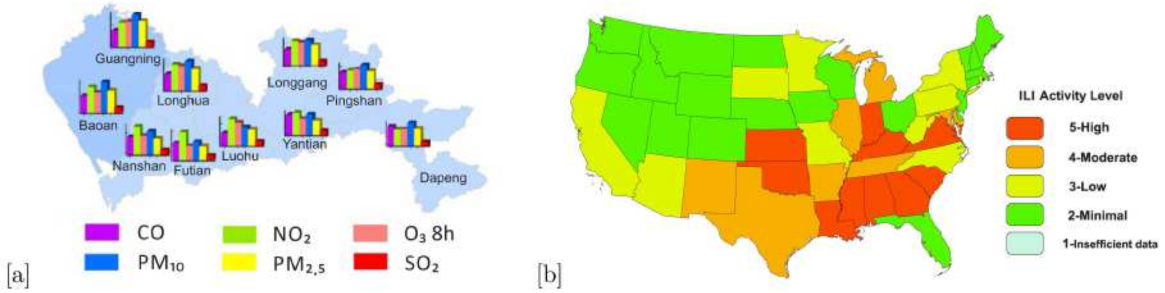


Fig. 1. [a] Relative amounts of six air pollutant subtypes in 10 districts in Shenzhen, China, 2013 [1], [b] ILI data for one week of the 2016-17 influenza season (CDC).

nearby locations tend to have more similar event subtype patterns. For instance, neighboring districts like Guangning and Longhua often share similar pollutant patterns, while distant districts like Longhua and Dapeng show more differences in air pollution patterns in Shenzhen, China. *2) Incomplete labels in spatial event subtypes*: Due to numerous subtypes and limited historical data, new subtypes may not appear in training sets for specific locations. For instance, in the Venezuela civil unrest dataset, 11 out of 14 cities lack some event subtypes. Similarly, in Nebraska, there were no level 3 or level 5 influenza events in 2011, making future forecasts for these scales challenging. This issue hampers the model's ability to predict unseen subtypes or scale levels, posing significant problems for rare but impactful events like pandemics and terrorist attacks. *3) Difficulties in representing event subtype patterns*: Subtle differences between event subtypes are hard to capture with manual features (e.g., bag-of-words), which are sparse, high-dimensional, and suffer from the *curse of dimensionality* [7]. This is exacerbated in multi-location training, underscoring the need for efficient end-to-end representation learning.

In this paper, we propose a novel a Deep multi-task learning for spatio-temporal incomplete qualitative event forecasting (DETECTIVE) framework for spatial event subtype forecasting that addresses all the above challenges. The main contributions of our study are as: *1) Developing a new deep-based framework for societal event subtype forecasting.* We formulate event subtype forecasting for multiple locations as a spatial incomplete multi-task learning problem and propose a novel deep-based framework that learns profound representations of event subtypes across tasks. We enforce shared latent feature representations for different locations while preserving heterogeneity in their event subtype patterns. *2) Proposing a model that enforces spatial event subtype patterns.* Based on the first law of geography, we enforce similar event subtype patterns among spatially-closer tasks via a novel deep regularization term to provide the theoretical equivalence to the ratio of the probabilities of the event subtypes distribution patterns in nearby locations. A shared bottom architecture learns the shared hidden representation of event subtypes across tasks. The representation is then passed into a goal-specific (class/scale) function with weight coefficients and a threshold matrix. We introduce two constraints in the goal-specific function to make the framework compatible with special cases (like multi-class, multi-scale, etc). To be more specific: a) in a multi-class problem, multiple weight coefficient vectors and one threshold matrix will be learned per task; b) in a multi-scale problem, one shared weight coefficient vector and threshold matrix will be learned per task. In cases of incomplete subtype events, we utilize the 'first law of geography'. That's why if a subtype is missing for one location, then it can utilize the ratio of probability of the event subtypes (one event subtype compared to another event subtype) of its adjacent locations to complement the missing subtype. In addition, the newly proposed deep regularization term enjoys better scalability with high-dimensional data and is thus more capable of handling complex real-world problems effectively and efficiently. *3) Developing an efficient algorithm for solving new non-convex and strongly-coupled problems.* To solve the proposed model's objective function, which is non-convex and highly-coupled, we propose algorithms based on the Alternating Direction Method of Multipliers (ADMM) that decomposes the original complex problems into subproblems that can be solved efficiently with analytical solutions and conventional stochastic optimization. *4) Conducting comprehensive experiments to validate the effectiveness and efficiency of the proposed model.* Experiments on six real-world datasets in two domains, civil unrest and air pollution event subtype forecasting, and ten datasets from civil unrest and influenza outbreaks domains for scale-level forecasting demonstrate that the proposed models outperform other comparison methods in different application domains, with sensitivity and qualitative analyses demonstrating the effectiveness of the proposed regularization term.

In summary, this study introduces DETECTIVE, an innovative deep framework for spatiotemporal event subtype forecasting across multiple locations. It addresses incomplete spatial multi-task learning and ensures shared yet heterogeneous event subtype representations. By applying the first law of geography, DETECTIVE enforces similar patterns among geographically close locations through a novel deep regularization term, predicting missing subtypes using probability ratios from nearby locations. It also employs an efficient ADMM-based algorithm to solve the complex, non-convex, coupled objective function by breaking it into manageable subproblems. Extensive experiments on datasets from civil unrest, air pollution, and influenza outbreaks demonstrate the framework's powerful and efficient solution for spatiotemporal event subtype forecasting.

## II. Problem Setups and Preliminaries Setups

### A. Problem Setup

Suppose there are $\mathcal{S}$ spatial locations (e.g., cities, states) in a country of interest and $T$ denotes all the time intervals. The spatio-temporal social indicator data (e.g., social media, news, pollutant factors) for location $s$ and time interval $t$ (e.g., one day) can be formulated as $X_{s,t} \in \mathbb{R}^{1\times D}$, which denotes a $D$-dimension feature vector whose $i$-th element is a feature value (e.g., the term frequency or index value).

The event subtype at location $s$ and time $t$ is defined as a nominal/ordinal response $Y_{s,t} \in \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_K\}$, where $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_K$ are class labels or $Y_{s,t} \in \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_\S\}$, where $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_\S$ are scale levels where $K$ and § is the total number of event classes and scales respectively. Here a "non-event" will also be defined as a default subtype when no event happens. A natural scale level ordering is denoted as $\mathcal{C}_1 \prec \mathcal{C}_2 \prec \ldots \prec \mathcal{C}_\S$, where $\prec$ is the ascending order relation.

Given the input data $X_{s,t}$ for a specific location $s$ and a time interval $t$, the goal is to predict the subtype of a future event, denoted by $Y_{s,\tau}$, for the same location $s$ and a future time interval $\tau$, where $\tau = t + p$ and $p > 0$ is the lead time. In this paper, the default time intervals $t$ is per day and the lead time $p$ is one day ahead unless otherwise specified. Formally, this problem is equivalent to learning a mapping from input data to a future event subtype $X_{s,t} \to Y_{s,\tau}$.

### B. Preliminaries

*1) Multi-Class Classification:* To address this issue, multi-class classification models [8] such as multinomial logistic regression (also known as softmax regression) and neural networks [9] are commonly used to solve the problem due to the nature of predicting multiple outputs with a single model. The objective function of our problem with the softmax regression formulation is as follows:

$$\mathcal{L}(\theta) = -\frac{1}{\mathcal{S}T}\left(\sum_{s}^{\mathcal{S}}\sum_{t}^{T}\sum_{k=1}^{K} 1\{Y_{s,t} = k\} \log \frac{e^{X_{s,t}\theta_k^T}}{\sum_{c=1}^{K} e^{X_{s,t}\theta_c^T}}\right) \tag{1}$$

where $\theta \in \mathbb{R}^{K\times D}$ is the parameter set of the model, $\theta_k \in \mathbb{R}^{1\times D}$ denotes the parameters for class $k$, and $1\{\cdot\}$ is the indicator function. For example, suppose the event subtype for location $s$ at time $t$ is $k$, then $1\{Y_{s,t} = k\} = 1$ while $1\{Y_{s,t} = j\} = 0$ for any $j \neq k$.

The (1) suffers from a critical challenge: all the locations share a single *weight coefficient* vector $\theta$, hence the model cannot handle any spatial heterogeneity in the event subtype for different locations. Consider civil unrest for example, finding 1000 tweets mentioning the keyword "student" in a time period could strongly suggest an education-related event for a city with a population of 10000 but may not be a strong indicator for a city with a population of a million. This discrepancy can lead to considerable heterogeneity of the weight coefficient $\theta$ for different locations.

To address this challenge, we can extend (1) to create a location-specific model, where each location $s$ has its own weight coefficient set, denoted as $\Theta_s \in \mathbb{R}^{K\times D}$. Here, $\Theta_{s,k} \in \mathbb{R}^{1\times D}$ denotes the parameters for location $s$ and for class $k$ and the objective function of the location-based softmax regression formulation is as follows:

$$\mathcal{L}(\Theta) = -\frac{1}{\mathcal{S}T}\left(\sum_{s}^{\mathcal{S}}\sum_{t}^{T}\sum_{k=1}^{K} 1\{Y_{s,t} = k\} \log \frac{e^{X_{s,t}\Theta_{s,k}^T}}{\sum_{c=1}^{K} e^{X_{s,t}\Theta_{s,c}^T}}\right) \tag{2}$$

However, the above formulation is still insufficient as (2) assumes all the locations are independent, even though some spatial correlations will exist among the various locations in terms of the event subtype pattern, as shown in Fig. 1(a). Also, (2) tries to learn an individual parameter set $\Theta_s$ for each location $s$, which can dramatically reduce the training sample size for a given location model. Furthermore, due to a large number of potential subtypes and the limited amount of local historical data, there may be unseen subtypes that have not appeared in a specific location within a time period. For example in Brazil, there were no education or medical-related protests in the city Belo Horizonte during the time period from July, 2013 to February, 2014. The specific model for Belo Horizonte will not be able to predict these two subtypes in the future.

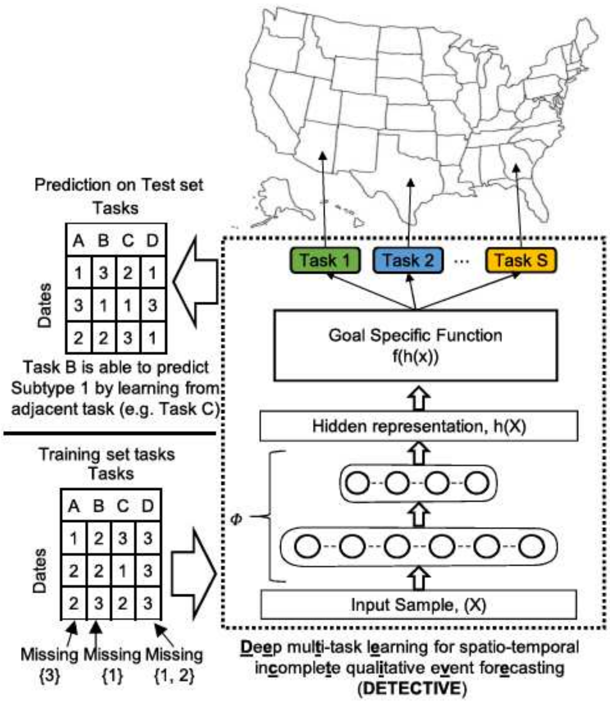
*2) Ordinal Regression:* If the response variable $Y_{s,t}$ is an ordinal variable, the assumption of order between event scales makes it inappropriate to apply conventional methods such as multi-class classification and regression directly. Specifically, conventional regression models like linear regression require continuous values and thus cannot handle the categorical variable $Y_{s,t}$ in our problem. Classification models, although they focus on categorical variables, only address nominal variables and ignore the ordinal information in our problem.

To predict the ordinal variable $Y_{s,t}$, ordinal regression models such as the proportional odds model (POM) [10] are commonly used to effectively leverage and address the ordinal nature of the problem. Compared to multi-class logistic regression, POM adds the constraint, the hyper-planes, that separate different classes are parallel for all classes, which is, the *weight co-efficient* vector $w$ is common across classes. The model also assumes that a latent variable underlies the ordinal response, which will be estimated by *threshold* matrix $b$ in the model in order to separate different class labels.

In the logistic ordinal regression, we model the cumulative probability as the logistic function. Thus, the objective function can be formulated as a negative log-likelihood:

$$\begin{aligned} \arg\min_{W,b} -\sum\nolimits_{s,t=1}^{\mathcal{S},\mathcal{T}} &\log(\sigma(w^T X_{s,t} + b_{s,Y_{s,t}}) \\ &- \sigma(w^T X_{s,t} + b_{s,Y_{s,t}-1})) \\ \text{s.t.} b_{s,1} \leq b_{s,2} &\leq b_{s,3} \leq \cdots \leq b_{s,\S-1} \end{aligned} \tag{3}$$

Where $w \in \mathbb{R}^{(D-1)\times 1}$, and $b \in \mathbb{R}^{s\times\S}$ are the two parameter sets to be estimated in the model, with $b_{s,0}{=}-\infty$ and $b_{s,\S}{=}\infty$ to represent extremal classes, $X_{s,t}$ denotes $t$-th sample of the $s$-th location, $Y_{s,t}$ denotes its corresponding scale. The function $\sigma(x)$ is the logistic sigmoid function denoted as $\sigma(x) = 1/(1 + e^{-x})$. Notice that our problem and proposed models are generic and


Prediction on Test set
Tasks
Dates
Task B is able to predict Subtype 1 by learning from adjacent task (e.g. Task C)
Training set tasks
Tasks
Dates
Missing {3}
Missing {1}
Missing {1, 2}
Task 1
Task 2
...
Task S
Goal Specific Function
f(h(x))
Hidden representation, h(X)
ϕ
Input Sample, (X)
Deep multi-task learning for spatio-temporal incomplete qualitative event forecasting
(DETECTIVE)

| A | B | C | D |
|---|---|---|---|
| 1 | 3 | 2 | 1 |
| 3 | 1 | 1 | 3 |
| 2 | 2 | 3 | 1 |

| A | B | C | D |
|---|---|---|---|
| 1 | 2 | 3 | 3 |
| 2 | 2 | 1 | 3 |
| 2 | 3 | 2 | 3 |

Fig. 2. Generic Flowchart of the proposed DETECTIVE framework. The input samples contain the input data $X_{s,t}$ for a specific location, $s$ (here indicated by A, B, C, D) and a time interval $t$, the goal is to predict the subtype of a future event, $Y_{s,\tau}$ (here indicated by the number in the matrix) for the same location $s$ and a future time interval $\tau$, where $\tau = t + p$ and $p > 0$ is the lead time. A shared bottom architecture, $\phi$ learns the shared hidden representation of event subtypes across tasks. The representation is then passed into a goal-specific (class/scale) function $f(\cdot)$ with weight coefficients, $W$ and a threshold matrix, $b$. This $f(\cdot)$ can be adapted based on the problem's nature. See the Fig. 3 for multi-class problem and the Fig. 4 for multi-scale problem.

can also accommodate other ordinal regression models. In this paper, we focus on POM.

The model proposed in (3) suffers from two challenges: 1) all the locations share a single *weight coefficient* vector $w$ and *threshold* vector $b$, therefore cannot handle any spatial heterogeneity in the event scale for different locations; and 2) (3) assumes all the locations are independent even though some spatial correlations exist among locations regarding the event scale pattern, as shown in Fig. 1(b). In order to jointly handle these challenges, in the next section, we present our DETECTIVE framework.

## III. DETECTIVE

The regression (softmax/ordinal) model can be seen as a special case of a neural network with 0 hidden layers. We propose a generalized Deep multi-task learning for spatio-temporal incomplete qualitative event forecasting (DETECTIVE) framework based on the deep architecture with an arbitrary number of hidden layers. Fig. 2 shows a flowchart of the proposed DETECTIVE framework. The framework adopts the idea of a shared bottom architecture that can learn the shared hidden representations of event subtypes across tasks. Initially, the objective function of the regression model is:

$$\mathcal{L}(W,b,\phi) = -\frac{1}{\mathcal{S}T}$$
$$\left(\sum_{s}^{\mathcal{S}}\sum_{t}^{T}\sum_{c=1}^{C} 1\{Y_{s,t}=c\}\log\frac{f_{W_{s,c},b_{s,c}}(h(X_{s,t}))}{\sum_{c=1}^{C} f_{W_{s,c},b_{s,c}}(h(X_{s,t}))}\right)$$
$$s.t., \hat{\mathcal{C}}_1(W) = \Omega_1, \hat{\mathcal{C}}_2(b) = \Omega_2 \tag{4}$$

where $X_{s,t}$ denotes $t$-th sample of the $s$-th location, $Y_{s,t}$ denotes its corresponding subtype or scale. Here $1\{\cdot\}$ is the indicator function. For example, suppose the event subtype for location $s$ at time $t$ is $c$, then $1\{Y_{s,t}=c\}=1$ while $1\{Y_{s,t}=j\}=0$ for any $j \neq c$.

In generalized DETECTIVE framework, the function $h(\cdot)$ denotes the computation of the shared hidden layers and $\phi$ denotes the parameter set of the network, the activation $h(X_{s,t})$ is thus the hidden representations learned by the shared hidden layers. $h(X_{s,t})$ is then passed as input to the goal (class/scale) specific function $f(\cdot)$ with the weight coefficient ($W \in \mathbb{R}^{S\times C\times(D-1)}$) and the threshold matrix $b$ ($b \in \mathbb{R}^{S\times§}$). Furthermore, the goal-specific (class/scale) function ($f(\cdot)$) with two constraints ($\Omega_1$ and $\Omega_2$) makes the framework compatible with special cases (for example, multi-class, multi-scale problems, etc.). For the special case of multi-scale prediction, the constraint $\Omega_1$ is implemented as $W_{s,c_1} = W_{s,c_2}, \forall c_1, c_2 \in C$. And then constraint $\Omega_1$ is implemented as $b_{s,c_1} \leq b_{s,c_2} \leq b_{s,c_3} \cdots \leq b_{s,c_{§-1}} \leq b_{s,c_§}$.

*Lemma 1:* The ratio of the *probability* of being belong to two types (1 and 2) of two tasks ($i$ and $j$) close in geo-spatial distance should also be similar. Mathematically, this can be expressed as:

$$\frac{P(Y_{i,t}\in\mathcal{C}_1|X_{i,t})}{P(Y_{i,t}\in\mathcal{C}_2|X_{i,t})} \approx \frac{P(Y_{j,t}\in\mathcal{C}_1|X_{j,t})}{P(Y_{j,t}\in\mathcal{C}_2|X_{j,t})} \tag{5}$$

*Proof:* Based on the first law of geography "everything is related to everything else, but near things are more related than distant things"[6], we know nearby locations will tend to be more similar to each other. For a time interval $t$, given two locations $i$ and $j$ that are close in geo-spatial distance, the probability of the event subtype is being belong to $\mathcal{C}_1$ at location $i$ denoted as $P(Y_{i,t}\in\mathcal{C}_1|X_{i,t})$, will be similar to that at location $j$, leads to the following equation:

$$P(Y_{i,t}\in\mathcal{C}_1|X_{i,t}) \approx P(Y_{j,t}\in\mathcal{C}_1|X_{j,t}) \tag{6}$$

Likewise, the ratio of the probability of the event subtype at location $i$ belong to event subtype $\mathcal{C}_1$ compared to event subtype $\mathcal{C}_2$, should also be similar to that at location $j$. This can be expressed as:

$$\frac{P(Y_{i,t}\in\mathcal{C}_1|X_{i,t})}{P(Y_{i,t}\in\mathcal{C}_2|X_{i,t})} \approx \frac{P(Y_{j,t}\in\mathcal{C}_1|X_{j,t})}{P(Y_{j,t}\in\mathcal{C}_2|X_{j,t})}$$

□

Based on Lemma 1, spatial adjacency-based deep regularization terms ($\mathcal{R}(\cdot)$) are proposed to regularize the hidden representation learned by the shared hidden layers to enforce similar event subtype patterns for spatially adjacent tasks. For example, in the left bottom of Fig. 2 Task B is closer to Task C compared to Task A, thus Task B and C can share knowledge of their subtype patterns and influence each other more strongly while Task A, which is further away, will not influence Task C as much. Consequently, with the help of this knowledge sharing, Task B is able to learn unseen event subtypes through Task C, mitigating the problem of incomplete subtype availability due to gaps in the local task training data. We also introduced $\alpha$ and $\beta$ to control the importance of the terms. Finally, the objective function of multi-class, multi-scale problem can be written as:

$$\mathcal{L} = \mathcal{L}(W,b,\phi) + \alpha\mathcal{R}_1(W) + \beta\mathcal{R}_2(W,b)$$
$$s.t., \hat{\mathcal{C}}_1(W) = \Omega_1, \hat{\mathcal{C}}_2(b) = \Omega_2 \tag{7}$$

### A. Special Cases of DETECTIVE

The goal-specific function ($f(\cdot)$) along with the constraints ($\Omega$) gives us the flexibility to implement the DETECTIVE framework for different special cases. In this section, we discussed two special cases which are multi-class and multi-scale problems in Sections III-A1 and III-A2 respectively.

*1) DETECTIVE-K:* For multi-class problem the class specific function $f(\cdot)$ would be:

$$f_{W_{s,c},b_{s,c}}(h(X_{s,t})) = e^{W_{s,c}\times h(X_{s,t})+b_{s,c}} \tag{8}$$

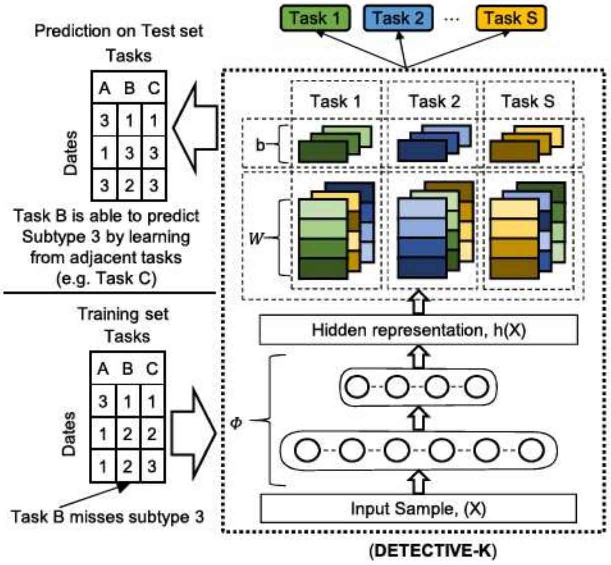
For convenience, we choose to use $\Theta$ ($\Theta \in \mathbb{R}^{S\times C\times D}$) to denote the parameters set consists of $W$ ($W \in \mathbb{R}^{S\times C\times (D-1)}$) and $b$ ($b \in \mathbb{R}^{S\times C}$)

$$\begin{aligned} f_{W_{s,c},b_{s,c}}(h(X_{s,t})) &= e^{W_{s,c}\times h(X_{s,t})+b_{s,c}} \\ &= e^{h(X_{s,t})\Theta_{s,c}} \\ &= f_{\Theta s,c}(h(X_{s,t})) \end{aligned} \tag{9}$$

Now for multi-class problem, $Y_{s,t} \in \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_K\}$, where $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_K$ are class levels. So for the $K$ classes the (4) can be directly applied for multi-class problem without constraints ($\Omega_1$ and $\Omega_2$) by replacing $f(\cdot)$.

$$\begin{aligned} \mathcal{L}(W,b,\phi) &= -\frac{1}{\mathcal{S}T} \\ &\times \left(\sum_{s}^{\mathcal{S}}\sum_{t}^{T}\sum_{k=1}^{K} 1\{Y_{s,t}=k\} \log \frac{e^{W_{s,k}h(X_{s,t})+b_{s,k}}}{\sum_{c=1}^{K} e^{e^{W_{s,c}h(X_{s,t})+b_{s,c}}}}\right) \\ \mathcal{L}(\Theta,\phi) &= -\frac{1}{\mathcal{S}T} \\ &\times \left(\sum_{s}^{\mathcal{S}}\sum_{t}^{T}\sum_{k=1}^{K} 1\{Y_{s,t}=k\} \log \frac{e^{h(X_{s,t})\Theta_{s,k}^T}}{\sum_{c=1}^{K} e^{h(X_{s,t})\Theta_{s,c}^T}}\right) \end{aligned} \tag{10}$$

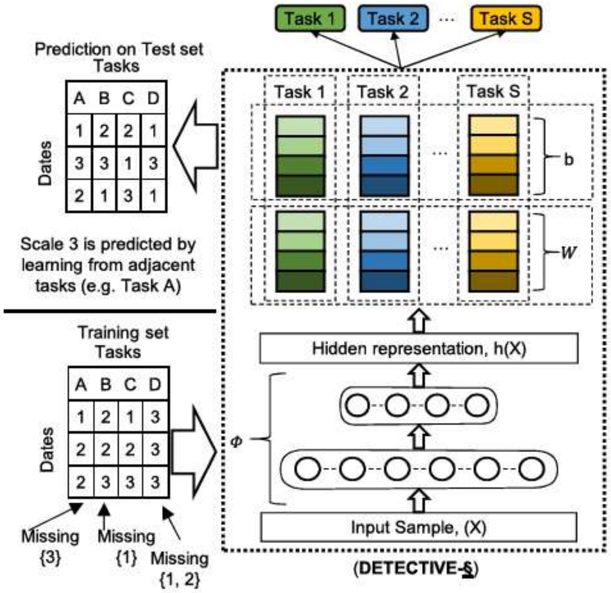
In order to jointly handle the spatial heterogeneity issue in (1) and spatial correlation issue in (2), multi-task learning technique is leveraged which can jointly learn the shared characteristics among tasks while preserve the exclusive patterns for each task [11], [12]. [13] have demonstrated the utility of applying a Multi-Task Learning framework for forecasting spatiotemporal event occurrence. More detailed literature survey is included in the supplemental material, available online. However, when forecasting event subtype, where multi-class classification problem is combined with multi-task learning, each task has only a limited number of samples and thus in practice not every task has a complete set of labels in the training set. For example, in Fig. 3 the bottom left box contains an example of a set of training data labels (event subtypes). Only 3rd task has a complete set of labels, the other two tasks are both missing one class. Consequently, the weight coefficient associated with the missing event subtype $k$ cannot be learned during training and the model is not capable of predicting the missing event subtypes. This issue becomes more severe as the number of



| A | B | C |
|---|---|---|
| 3 | 1 | 1 |
| 1 | 3 | 3 |
| 3 | 2 | 3 |

| A | B | C |
|---|---|---|
| 3 | 1 | 1 |
| 1 | 2 | 2 |
| 1 | 2 | 3 |

Fig. 3. Special Case: DETECTIVE-K framework. Here the goal specific function $f(\cdot)$ showed in the generic model structure (Fig. 2) has been adapted for multi-class problem. Multiple weight coefficient vectors and threshold matrices will be learned per task for class prediction.



| A | B | C | D |
|---|---|---|---|
| 1 | 2 | 2 | 1 |
| 3 | 3 | 1 | 3 |
| 2 | 1 | 3 | 1 |

| A | B | C | D |
|---|---|---|---|
| 1 | 2 | 1 | 3 |
| 2 | 2 | 2 | 3 |
| 2 | 3 | 3 | 3 |

Fig. 4. Special Case: DETECTIVE-§framework. Here the goal specific function $f(\cdot)$ showed in the generic model structure (Fig. 2) has been adapted for multi-scale problem. Multiple weight coefficient vectors and threshold matrices will be learned per task for class prediction. One shared weight coefficient vector and threshold matrix will be learned per task for scale prediction.

class labels increases.

$$\frac{P(Y_{i,t}=\mathcal{C}_a|X_{i,t})}{P(Y_{i,t}=\mathcal{C}_b|X_{i,t})} \approx \frac{P(Y_{j,t}=\mathcal{C}_a|X_{j,t})}{P(Y_{j,t}=\mathcal{C}_b|X_{j,t})} \tag{11}$$

In order to address this problem, we propose allowing correlated tasks to adaptively complement each other's missing classes based on Lemma 1. From the (5), the ratio of the probability of the event class at location $i$ being equal to event class $\mathcal{C}_a$ compared to event class $\mathcal{C}_b$, should also be similar to that at location $j$. By considering $\mathcal{C}_1 = \{\mathcal{C}_a\}$ and $\mathcal{C}_2 = \{\mathcal{C}_b\}$ This can be expressed as (11).

The posterior probability $P(Y_{i,t}=\mathcal{C}_a|X_{i,t})$ can be equivalently represented by any multi-class based models. The similarity pattern based on the ratio of the probability above can thus be equivalently denoted by input $X$ and weight coefficient $\Theta$ based on (2), as shown in Lemma 2.

*Lemma 2:* In the DETECTIVE-K framework, for any deep learning architectures that use the softmax function as their output layer, (11) is theoretically equivalent to the following:

$$h(X_{i,t})(\Theta_{i,a}-\Theta_{i,b})^T \approx h(X_{j,t})(\Theta_{j,a}-\Theta_{j,b})^T \tag{12}$$

where $\Theta_{i,b}$ denotes the task specific output layer weight coefficient vector for task $i$ and class $\mathcal{C}_b$.

*Proof:* We can derive the lemma from the following equations:

$$P(Y_{i,t}=\mathcal{C}_a|X_{i,t}) = \frac{e^{h(X_{i,t})\Theta_{i,a}^T}}{\sum_{k=1}^{K} e^{h(X_{i,t})\Theta_{i,k}^T}} \tag{13}$$

Equation (13) is the definition of the posterior probability of the softmax output layer for a given input $X$ and function $h(\cdot)$. From this, we can derive an equivalent expression in logarithmic form as follows:

$$\log P(Y_{i,t}=\mathcal{C}_a|X_{i,t}) = \log e^{h(X_{i,t})\Theta_{i,a}^T} - \log \sum_{k=1}^{K} e^{h(X_{i,t})\Theta_{i,k}^T}$$

We can now subtract any pair of classes $\mathcal{C}_a$ and $\mathcal{C}_b$ to omit the common denominator in (13), as shown below:

$$\log \frac{P(Y_{i,t}=\mathcal{C}_a|X_{i,t})}{P(Y_{i,t}=\mathcal{C}_b|X_{i,t})} = h(X_{i,t})\Theta_{i,a}^T - h(X_{i,t})\Theta_{i,b}^T \tag{14}$$

Thus, combining (11) and (14), we can safely conclude that given two tasks $i$ and $j$ that are close geo-spatially, the difference between the products of the hidden representation of corresponding input and weight coefficients of any pair of classes $h(X_i)(\Theta_{i,a}-\Theta_{i,b})^T$ and $h(X_j)(\Theta_{j,a}-\Theta_{j,b})^T$ should be similar, as shown in (12). □

Therefore, based on (11), and equivalently on (12) we define the regularization term for $\Theta$ based on spatial adjacency of the tasks.

$$\alpha\mathcal{R}_1(W) = 0,$$

$$\beta\mathcal{R}_2(W,b) = \frac{\beta}{2}\sum\nolimits_{s}^{\mathcal{S}}\sum\nolimits_{i,j}^{C_k^2} \|h(X_{s,t})(\Theta_{s,i}-\Theta_{s,j})^T - \frac{1}{N_s}\sum\nolimits_{c}^{\mathcal{S}} adj(s,c)h(X_{c,t})(\Theta_{c,i}-\Theta_{c,j})^T\|_2^2 \tag{15}$$

Now, replacing $\mathcal{L}(\Theta,\phi)$, $\alpha\mathcal{R}_1(W)$, and $\beta\mathcal{R}_2(W,b)$ of (7) the model for multi-class problem will be:

$$\mathcal{L}(\Theta,\phi) + \frac{\beta}{2}\sum\nolimits_{s}^{\mathcal{S}}\sum\nolimits_{i,j}^{C_k^2} \|h(X_s)(\Theta_{s,i}-\Theta_{s,j})^T - \frac{1}{N_s}\sum\nolimits_{c}^{\mathcal{S}} adj(s,c)h(X_c)(\Theta_{c,i}-\Theta_{c,j})^T\|_2^2 \tag{16}$$

where $\mathcal{L}(\Theta,\phi)$ in (10) is the general multi-task deep learning objective function; $\phi$ is the weight coefficient parameter set for the shared hidden layers; $\Theta$ is the task specific output layer parameter set with $\Theta_{s,i}$ denoting the parameters for task $s$ and for predicting class $\mathcal{C}_i$. The function $adj(s,c)$ defines the adjacency relation between $s$ and $c$, which can be defined based on either spatial correlations such as spatial contiguity or spatial distance. $N_s$ is the normalization term for location $s$ such that $N_s = \sum_{c}^{\mathcal{S}} adj(s,c)$. Here, the adjacent function is defined based on the physical distance and the well-known generalized RBF kernel [14], as: $adj(s,c) = e^{-\gamma d(s,c)^2}$ . The function $d(s,c)$ can be the physical distance between two spatial locations and $\gamma$ is the scaling factor.

The proposed regularization term encourages adjacent tasks to have a similar ratio of the probability between any pair of event subtypes by ensuring the difference between the corresponding weight coefficients and input $h(X_i)(\Theta_{i,\mathcal{C}_a}-\Theta_{i,\mathcal{C}_b})^T$ is similar for adjacent tasks. The regularization hyper-parameter $\beta$ controls the importance of this term, which can be tuned via cross-validation. Lemma 2 and the above model objective indicate that instead of directly applying the regularization on input data $X$, DETECTIVE-K learns the mapping from the input data from different tasks in a deep shared feature space and then applies the spatial regularization to the latent representation.

*2) DETECTIVE-§:* For multi-class, the scale specific function $f(\cdot)$ would be:

$$f_{W_{s,c},b_{s,c}}(h(X_{s,t})) = \sigma(W_{s,c}h(X_{s,t}) + b_{s,c}) - \sigma(W_{s,c-1}h(X_{s,t}) + b_{s,c-1}) \tag{17}$$

For multi-scale problem, $Y_{s,t} \in \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_\S\}$, where $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_\S$ are scale levels. Since for multi-scale case the sum of all classes specific activations are naturally one, this is why we can ignore the denominator of the (4).

$$\mathcal{L}(W,b,\phi) = -\frac{1}{\mathcal{S}T}\left(\sum_{s}^{\mathcal{S}}\sum_{t}^{T}\sum_{\S=1}^{\S} 1\{Y_{s,t}=\S\} \log(\sigma(W_{s,\S}h(X_{s,t}) + b_{s,\S}) - \sigma(W_{s,\S-1}h(X_{s,t}) + b_{s,\S-1}))\right) \tag{18}$$

We have two constraints $\Omega_1$ and $\Omega_2$ in the (7). For the special case of mult-scale prediction, the constraint $\Omega_1$ is implemented as $W_{s,c_1} = W_{s,c_2}, \forall c_1, c_2 \in C$. And then constraint $\Omega_2$ is implemented as $b_{s,0} \leq b_{s,1} \leq b_{s,2} \cdots \leq b_{s,\S-1} \leq b_{s,\S}$. Due to $\Omega_1$, we can define feature weight coefficient matrix $W \in \mathbb{R}^{\mathcal{S}\times(D-1)}$ instead of $W \in \mathbb{R}^{\mathcal{S}\times C\times(D-1)}$ where each column of $W$, denoted as $W_{s,\cdot}$, is the feature weight coefficient vector for task $s$, while each row of $b$, denoted as $b_{s,\cdot}$, is the threshold vector for task $s$.

So, for multi-scale problem, the (7) will take the form as:

$$\underset{W,b}{\arg\min} - \sum\nolimits_{s,t}^{\mathcal{S},\mathcal{T}} \log\Bigg(\sigma(W_s h(X_{s,t}) + b_{s,\S}) - \sigma(W_s h(X_{s,t}) + b_{s,\S-1})\Bigg) + \alpha\mathcal{R}_1(W) + \beta\mathcal{R}_2(W,b)$$
$$\text{s.t. } b_{s,1} \le b_{s,2} \le b_{s,3} \le \cdots \le b_{s,\S-1},\ s \in \{1,2,\ldots,\S\} \quad (19)$$

Where $W_s \in \mathbb{R}^{1\times(D-1)}$, and $b_s \in \mathbb{R}^{(\S-1)\times 1}$ are the two parameter sets to be estimated in the model, with $b_{s,0}{=}-\infty$ and $b_{s,\S}{=}\infty$ to represent extremal classes; $X_{s,t}$ denotes $t$-th sample of the $s$-th location; $Y_{s,t}$ denotes its corresponding scale; $\sigma(x)$ is the logistic sigmoid function denoted as $\sigma(x) = 1/(1+e^{-x})$, and $h(\cdot)$ denotes the computation of the shared hidden layers coupled with parameter set $\phi$.

To handle the spatial heterogeneity of event scale criteria for different locations, we need to build an exclusive model for each individual locations, all of which have their own *thresholds*. Although these *thresholds* are different, different locations share similar feature *weight coefficients* patterns because people generally share a common language and speak in a similar way, so the keywords for a topic of interest will be similar across different locations, for example, "influenza" and "cough",would both refer to the topic 'flu'. Learning multiple related tasks simultaneously effectively increases the sample size for each task, since when we learn a model for a specific task, we use information from all other tasks.

Therefore, we propose to leverage multitask learning in ordinal regression to enforce different tasks that share a similar *weight coefficients* pattern but reserve their own *thresholds*. The similar pattern of $W$ across different tasks is achieved by enforcing a similar sparsity pattern among tasks. We can add $\ell_{2,1}$ norm regularization over the $W$ matrix, which sums the $\ell_{2,1}$ norms for each feature, and each $\ell_{2,1}$ norm is enforced for all the tasks for each feature. Thus, the $i$-th feature, which corresponds to the $i$-th element in each model, is likely to be selected or not by all models simultaneously. Mathematically, we propose the model as follows:

$$\arg\min\nolimits_{W,b,\phi} \mathcal{L}(W,b,\phi) + \alpha||W||_{2,1}$$
$$\text{s.t. } b_{s,1} \le b_{s,2} \le b_{s,3} \le \cdots \le b_{s,\S-1},\ s \in \{1,2,\ldots,\mathcal{S}\}$$

Where we define $\mathcal{L}(W,b,\phi)$ for simplicity and for later use as:

$$-\sum\nolimits_{s,t=1}^{\mathcal{S},\mathcal{T}} \log\left(\sigma\left(W_s h(X_{s,t}) + b_{s,Y_{s,t}}\right) - \sigma\left(W_s h(X_{s,t}) + b_{s,Y_{s,t}-1}\right)\right)$$

Here we can consider $\mathcal{R}_1(W) = ||W||_{2,1}$ is the group sparsity term for matrix $W$ which encourages all tasks to select a common set of features; it can be computed as the sum of $\ell_2$-norm for each row in $W$. The regularization hyper-parameter $\alpha$ controls the sparsity.

The *odds* of being equal or under $\mathcal{C}_a$ is defined as the fraction of the probability of being equal or under $\mathcal{C}_a$ over the probability of being above $\mathcal{C}_a$ and mathematically:

$$odds(Y_{i,t} \preceq \mathcal{C}_a|X_{i,t})) = \frac{P(Y_{i,t} \preceq \mathcal{C}_a|X_{i,t})}{P(Y_{i,t} \succ \mathcal{C}_a|X_{i,t})} \quad (20)$$

Now from Lemma 1, if we consider $\mathcal{C}_1 = \{\mathcal{C}_i|\mathcal{C}_i \preceq \mathcal{C}_a\}$ and $\mathcal{C}_2 = \{\mathcal{C}_i|\mathcal{C}_i \succ \mathcal{C}_a\}$ then

$$\frac{P(Y_{i,t} \preceq \mathcal{C}_a|X_{i,t})}{P(Y_{i,t} \succ \mathcal{C}_a|X_{i,t})} \approx \frac{P(Y_{j,t} \preceq \mathcal{C}_a|X_{j,t})}{P(Y_{j,t} \succ \mathcal{C}_a|X_{j,t})}$$
$$odds(Y_{i,t} \preceq \mathcal{C}_a|X_{i,t})) = odds(Y_{i,t} \preceq \mathcal{C}_a|X_{j,t})) \quad (21)$$

So, the ratio of the *odds* of being equal or under two adjacent scales $a$ and $b$ of two tasks ($i$ and $j$) close in geo-spatial distance should also be similar. Mathematically:

$$\frac{odds(Y_{i,t} \preceq \mathcal{C}_a|X_{i,t})}{odds(Y_{i,t} \preceq \mathcal{C}_b|X_{i,t}))} \approx \frac{odds(Y_{j,t} \preceq \mathcal{C}_a|X_{j,t})}{odds(Y_{j,t} \preceq \mathcal{C}_b|X_{j,t}))} \quad (22)$$

The similarity pattern in (22) can thus be equivalently denoted by thresholds (shown in Lemma 3).

*Lemma 3:* Equation (22) is theoretically equivalent to:

$$b_{i,\mathcal{C}_b} - b_{i,\mathcal{C}_a} \approx b_{j,\mathcal{C}_b} - b_{j,\mathcal{C}_a} \quad (23)$$

where $i$ and $j$ are two tasks that are close in geo-spatial distance and $\mathcal{C}_a$ and $\mathcal{C}_b$ are two adjacent event scales.

*Proof:* We can derive the theorem from the following equations:

$$ln\left(\frac{P(Y_{i,t} \preceq \mathcal{C}_a|X_{i,t})}{P(Y_{i,t} \succ \mathcal{C}_a|X_{i,t})}\right) = W_{\cdot,i}^T h(X_{i,t}) + b_{i,\mathcal{C}_a} \quad (24)$$

Equation (24) is the definition of POM. From this, we can derive an equivalent expression with $\mathcal{C}_b$ and subtract one from the other to omit the input vector $X_{i,t}$ on the right, as shown in the following equation:

$$ln\left(\frac{P(Y_{i,t} \preceq \mathcal{C}_b|X_{i,t})}{P(Y_{i,t} \succ \mathcal{C}_b|X_{i,t})}\right) - ln\left(\frac{P(Y_{i,t} \preceq \mathcal{C}_a|X_{i,t})}{P(Y_{i,t} \succ \mathcal{C}_a|X_{i,t})}\right)$$
$$= b_{i,\mathcal{C}_b} - b_{i,\mathcal{C}_a}$$

Combining above equation with (20), where the term *odds* is defined, we obtain the ratio of odds with $\theta$ as:

$$\frac{odds(Y_{i,t} \preceq \mathcal{C}_b|X_{i,t})}{odds(Y_{i,t} \preceq \mathcal{C}_a|X_{i,t}))} = e^{b_{i,\mathcal{C}_b} - b_{i,\mathcal{C}_a}} \quad (25)$$

Thus, by combining (22) and (25), gives the conclusion that given two tasks $i$ and $j$ that are close in geo-spatial distance, the difference between threshold $b_{i,\mathcal{C}_b} - b_{i,\mathcal{C}_a}$ and $b_{j,\mathcal{C}_b} - b_{j,\mathcal{C}_a}$ should be similar, as in (23). □

Therefore, we define the other regularization term $\beta\mathcal{R}_2(W,b)$ to encourage the difference between threshold parameter $b_{i,\mathcal{C}_b} - b_{i,\mathcal{C}_a}$ to be similar among adjacent tasks. Mathematically, the DETECTIVE-§model is as follows:

$$\arg\min\nolimits_{W,b,\phi} \mathcal{L}(W,b,\phi) + \alpha\|W\|_{2,1} +$$
$$\frac{\beta}{2}\sum_{i=1}^{\mathcal{S}}\sum_{j=2}^{\S-1}\left\|(b_{i,j} - b_{i,j-1}) - \frac{1}{N_i}\sum_{z\in adj(i)}(b_{z,j} - b_{z,j-1})\right\|_2^2$$
$$\text{s.t. } b_{i,1} \le b_{i,2} \le b_{i,3} \le \cdots \le b_{i,\S-1},\ i \in \{1,2,\ldots,\mathcal{S}\} \quad (26)$$

Where the function $adj(i)$ returns the set of tasks that is adjacent to task $i$ and $N_i$ is the total number of its neighbors. This term will encourage adjacent tasks to have a similar ratio for the *odds* between two consecutive scales by encouraging the difference between threshold parameter $b_{i+1} - b_i$ to be similar among adjacent tasks. The regularization hyper-parameter $\beta$ controls the importance of this term.

## IV. Algorithms

### A. Algorithm of DETECTIVE-K

The problem in (16) is nonconvex and parameters are tightly coupled together within the new regularization term. Moreover, the function $h(\cdot)$ involves the shared neural network layers, with highly complex objective functions coupled with parameter set $\phi$. Instead of directly solving the whole problem with regularization, existing works typically first decompose it into subproblems which are much simpler or even with analytical solutions and hence ensures the efficiency. For example, several ADMM [15] based methods has been proposed: [16] applied ADMM on deep convolutional neural networks with sparse regularization and observed improvement on the optimization efficiency and overall performance; [17] proposed ADMM-NET for solving the general Compressive Sensing MRI problem. However, those algorithms are normally problem dependent and thus can not be directly used here. A new method is needed to solve our new problem which is highly challenging. Thus, we propose a new algorithm based on ADMM that first decomposes the original problem into several simpler subproblems that can then be solved iteratively. Our algorithm ensures global optimal solutions with analytical solutions for all subproblems except the subproblem that includes the original deep model loss, which will be solved with Stochastic Gradient Descent (SGD) to get local optima. More details of the algorithm are presented as follows.

Based on the ADMM formulation, the original objective function of DETECTIVE-K can now be re-written as follows:

$$\begin{aligned}
&\mathcal{L}(\Theta,\phi) + \frac{\beta}{2}\sum\nolimits_{s}^{\mathcal{S}}\sum\nolimits_{i,j}^{C_k^2}\Bigg\|\bar{Z}_s(\bar{V}_{s,i} - \bar{V}_{s,j})^T \\
&\qquad - \frac{1}{N_s}\sum\nolimits_{c}^{\mathcal{S}} adj(s,c) Z_c(\bar{U}_{c,i} - \bar{U}_{c,j})^T\Bigg\|_2^2 \\
&\text{s.t. } \Theta = \bar{V}, \Theta = \bar{U}, \bar{Z} = h(X) \qquad (27)
\end{aligned}$$

Thus, by decoupling the output layer parameter set $\Theta$ that appears both in deep model loss and regularization term, the original problem is transformed into a simpler one with auxiliary variables $\bar{V}$, $\bar{U}$ and $\bar{Z}$. The augmented Lagrangian that uses additional quadratic penalty terms with penalty parameter $\rho$ is further computed as follows:

$$\begin{aligned}
&L(\phi,\Theta,\bar{V},\bar{U},\bar{Z}) \\
&= \mathcal{L}_D(\phi,\Theta) + (y^{(1)}(\bar{Z} - h(X))^T) + \frac{\rho}{2}||\bar{Z} - h(X)||_2^2 \\
&+ \frac{\beta}{2}\sum_{s}^{\mathcal{S}}\sum_{i,j}^{C_k^2}\Bigg\|\bar{Z}_s(\bar{V}_{s,i} - \bar{V}_{s,j})^T - \frac{1}{N_s}\sum_{c}^{\mathcal{S}} adj(s,c)\bar{Z}_c(\bar{U}_{c,i} - \bar{U}_{c,j})^T\Bigg\|_2^2 \\
&+ tr(y^{(2)}(\Theta - \bar{V})^T) + \frac{\rho}{2}||\Theta - \bar{V}||_2^2 \\
&+ tr(y^{(3)}(\Theta - \bar{U})^T) + \frac{\rho}{2}||\theta - \bar{U}||_2^2
\end{aligned}$$

where the $tr(\cdot)$ operator denotes the trace of the matrix.

The pseudo-code of the proposed algorithm is summarized in Algorithm 1. The parameter set $\{\phi, \Theta, \bar{V}, \bar{U}, \bar{Z}, y^{(1)}, y^{(2)}, y^{(3)}\}$ is alternately solved by the proposed algorithm until convergence is achieved. Lines 3-15 show the alternating optimization for each of the variables. $M \in \mathbb{R}^{k \times C_k^2}$ is an auxiliary matrix to help make the computation in matrix format. The detailed optimization for all the variables is described in supplemental materials, available online.

### B. Algorithm of DETECTIVE-§

The pseudo-code of the proposed algorithm is summarized in Algorithm 2 to solve the (26). The parameter set $\{W, b, \phi, U, V, y^{(1)}, y^{(2)}, y^{(3)}\}$ is alternately solved by the proposed algorithm until convergence is achieved. Lines 3-7 show the alternating optimization of each of the variables. The detailed optimization for all the variables is described in more detail below.

Base on ADMM formulation, the original objective function of model can be re-written as follows:

$$\begin{aligned}
L(W,b,\phi,U,V) = \mathcal{L}(W,b,\phi) &+ \alpha||U||_{2,1} \\
&+ \frac{\beta}{2}\sum_{i=1}^{\mathcal{S}}\sum_{j=2}^{\S-1}\Bigg\|(V_{i,j} - V_{i,j-1}) \\
&- \frac{1}{N_i}\sum_{z\in adj(i)}(V_{z,j} - V_{z,j-1})\Bigg\|_2^2 \\
&\text{s.t. } W = U, b = V, \\
V_{i,1} \le V_{i,2} \le V_{i,3} \le \cdots \le V_{i,\S-1} &\text{ for } i \in \{1,2,\ldots,\mathcal{S}\} \qquad (28)
\end{aligned}$$

Thus, the augmented Lagrangian is:

$$\begin{aligned}
&\underset{W,b,\phi,U,V}{\arg\min}\ \mathcal{L}(W,b,\phi) + \alpha||U||_{2,1} \\
&+ trace(y^{(1)}(W-U)^T) + \rho/2||W-U||_2^2 \\
&+ trace(y^{(2)}(b-V)^T) + \rho/2||b-V||_2^2 \\
&+ \frac{\beta}{2}\sum_{i=1}^{\mathcal{S}}\sum_{j=2}^{\S-1}\Bigg\|(V_{i,j} - V_{i,j-1}) - \frac{1}{N_i}\sum_{z\in adj(i)}(V_{z,j} - V_{z,j-1})\Bigg\|_2^2 \\
&+ \sum\nolimits_{i=2}^{\S-1} y_{\cdot,i}^{(3)}(V_{\cdot,i-1} - V_{\cdot,i})^T + \rho/2\sum\nolimits_{i=2}^{k-1}||\max(V_{\cdot,i-1} \\
&- V_{\cdot,i}, 0)||_2^2
\end{aligned}$$

Notice that the $max$ operator here acts as a vector max which will set the element of the vector to 0 when it is less than 0.

**Algorithm 1:** The Proposed Algorithm for DETECTIVE-K.

```
Require: $X, Y, \rho, \beta, \lambda$
Ensure: solution $\phi, \Theta$
initialize $\phi^0, \Theta^0, \bar{V}^0, \bar{U}^0, \bar{Z}^0, y^{(1)0}, y^{(2)0}, y^{(3)0}, i=0$
repeat
    % Solve subproblem of variable $\phi, \Theta$ by fixing the other variables
    $\phi^i, \Theta^i \Leftarrow \arg\min_{\phi,\Theta} \mathcal{L}_D(\phi, \Theta) + (y^{(1)}(\bar{Z} - h(X))^T) + \frac{\rho}{2}||\bar{Z} - h(X)||_2^2 + (y^{(2)}(\Theta - \bar{V})^T) + \frac{\rho}{2}||\Theta - \bar{V}||_2^2 + (y^{(3)}(\Theta - \bar{U})^T) + \frac{\rho}{2}||\Theta - \bar{U}||_2^2$
    for $s \Leftarrow 1$ to $K$ do
    % Get the analytical solution by setting $\nabla_{\bar{V}_s} L(\phi, \Theta, \bar{V}, \bar{U}, \bar{Z}) = 0$
    $V_s^i \Leftarrow (\beta(\bar{Z}_s^T \bar{Z}_s) \otimes (MM^T) + \rho I)^{-1} vec(y_s^{(2)} + \rho\Theta_s + \beta M(\frac{1}{N_s}\sum_c^{\mathcal{S}} adj(s,c)\bar{Z}_c \bar{U}_c^T M)^T \bar{Z}_s)$
    end for
    for $c \Leftarrow 1$ to $K$ do
    % Get the analytical solution by setting $\nabla_{\bar{U}_c} L(\phi, \Theta, \bar{V}, \bar{U}, \bar{Z}) = 0$
    $\bar{U}_c^i \Leftarrow (\beta\sum_s^{\mathcal{S}} \frac{adj(s,c)^2}{N_s^2}(\bar{Z}_c^T \bar{Z}_c) \otimes (MM^T) + \rho I)^{-1} vec(y_c^{(3)} + \rho\Theta_c - \beta\sum_s^{\mathcal{S}} M(\frac{1}{N_s}\sum_{i \neq c}^{\mathcal{S}} adj(s,i)\bar{Z}_i \bar{U}_i^T M - \bar{Z}_s \bar{V}_s^T M)^T \bar{Z}_c)$
    end for
    for $s \Leftarrow 1$ to $K$ do
    % Get the analytical solution by setting $\nabla_{\bar{Z}_s} L(\phi, \Theta, \bar{V}, \bar{U}, \bar{Z}) = 0$
    $\bar{Z}_s^i \Leftarrow (-y_s^{(1)} + \rho h(X_s) + \beta(\frac{1}{N_s}\sum_c^{\mathcal{S}} adj(s,c)\bar{Z}_c \bar{U}_c^T M)M^T \bar{V}_s)(\beta\bar{V}_s^T MM^T \bar{V}_s + \rho I)^{-1}$
    end for
    $y^{(1)i} \Leftarrow y^{(1)} + \rho(\bar{Z} - f(X))$    % Update dual variable $y^{(1)}$
    $y^{(2)i} \Leftarrow y^{(2)} + \rho(\Theta - \bar{V})$    % Update dual variable $y^{(2)}$
    $y^{(3)i} \Leftarrow y^{(3)} + \rho(\Theta - \bar{U})$    % Update dual variable $y^{(3)}$
    $i \Leftarrow i + 1$
until convergence
```

*1) Update $W, b, \phi$:* The sub-problem of updating $W$, $b$ and $\phi$ is as follows:

$$\arg\min_{W,b,\phi} \mathcal{L}(W, b, \phi) + trace(y^{(1)}(W - U)^T) + \rho/2||W - U||_2^2 + trace(y^{(2)}(b - V)^T) + \rho/2||b - V||_2^2 \quad (29)$$

Since $\mathcal{L}(W, b, \phi)$ is a non-convex function with respect to $W$, $b$ and $\phi$, we will use a traditional gradient descent algorithm, carefully choosing the step size $\lambda_W$, $\lambda_b$ and $\lambda_\phi$ for $W$, $b$ and $\phi$ to jointly update them to local optima.

**Algorithm 2:** The Proposed Algorithm for DETECTIVE-§.

```
Require: $X, Y, \rho, \alpha, \beta, \lambda_W, \lambda_b, \lambda_\phi$
Ensure: solution $W, b, \phi$
initialize $W^0, b^0, \phi^0, U^0, V^0, y^{(1)0}, y^{(2)0}, y^{(3)0}, i = 0$
repeat
    $W^i, b^i, \phi^i \Leftarrow$ Equation (29)
    $U^i \Leftarrow$ Equation (30)
    $V^i \Leftarrow$ calculation following Theorem IV-B3
    $y^{(1)i}, y^{(2)i}, y^{(3)i} \Leftarrow$ Equation (31)
    $i \Leftarrow i + 1$
until convergence
```

*2) Update $U$:* The sub-problem of updating $U$ is as follows:

$$\arg\min_U \alpha||U||_{2,1} + trace(y^{(1)}(W - U)^T) + \rho/2||W - U||_2^2 \quad (30)$$

This can be solved by proximal gradient descent using the proximal operator on the $\ell_{2,1}$ norm [18].

*3) Update $V$:* The sub-problem of updating $V$ is as follows:

$$\begin{aligned}\arg\min_V \frac{\beta}{2}\sum_{i=1}^{\mathcal{S}}\sum_{j=2}^{§-1}\Bigg\| (V_{i,j} - V_{i,j-1}) \\ - \frac{1}{N_i}\sum_{z \in adj(i)} (V_{z,j} - V_{z,j-1}) \Bigg\|_2^2 \\ + \rho/2||b - V||_2^2 + trace(y^{(2)}(b - V)^T) \\ + \textstyle\sum_{i=2}^{§-1} y_{\cdot,i}^{(3)}(V_{\cdot,i-1} - V_{\cdot,i})^T \\ + \rho/2\textstyle\sum_{i=2}^{§-1} ||\max(V_{\cdot,i-1} - V_{\cdot,i}, 0)||_2^2\end{aligned}$$

The $adj()$ function introduces some difficulties for updating $V$, since every pair of consecutive class level thresholds for the same task show in the same term. In addition, the same class level threshold among all tasks will also lead to recursive relationships. This makes elemental-wise updating of $V$ impossible in practice.

In order to address this problem, we can treat the $adj()$ function as the matrix representation $R \in \mathbb{R}^{\mathcal{S} \times \mathcal{S}}$, and reformulate the problem as matrix multiplication:

$$\begin{aligned}\arg\min_V \frac{\beta}{2}\sum_{i=1}^{\mathcal{S}}\sum_{j=2}^{§-1} \left|\left| (V_{\cdot,j} - V_{\cdot,j-1})^T R^T \right|\right|_2^2 \\ + trace\left(y^{(2)}(b - V)^T\right) \\ + \frac{\rho}{2}||b - V||_2^2 + \sum_{i=2}^{§-1} y_{\cdot,i}^{(3)}(V_{\cdot,i-1} - V_{\cdot,i})^T \\ + \frac{\rho}{2}\sum_{i=2}^{§-1} ||\max(V_{\cdot,i-1} - V_{\cdot,i}, 0)||_2^2\end{aligned}$$

Where $R_{i,i} = 1$ and $R_{i,adj(i)} = -\frac{1}{N_i}$, for $i = 1 \ldots \mathcal{S}$. $N_i$ is the total number of neighbors of task $i$.

*Corollary 1:* The optimal solution for matrix V can be obtained by computing its column vectors in order as follows:

$$V_{\cdot,1} = y^{(2)}_{\cdot,1/}\rho + b_{\cdot,1}$$

$$V_{\cdot,i} =$$

$$\begin{cases} (\beta V_{\cdot,i\text{-}1} L + y^{(2)}_{\cdot,i} + \rho(b_{\cdot,i} + V_{\cdot,i\text{-}1}) + y^{(3)}_{\cdot,i\text{-}1}) \\ \qquad \times(\beta L + 2\rho I)^{-1} & V_{\cdot,i} < V_{\cdot,i\text{-}1} \\ (\beta V_{\cdot,i-1} L + y^{(2)}_{\cdot,i} + \rho b_{\cdot,i} + y^{(3)}_{\cdot,i-1}) \\ \qquad \times(\beta L + \rho I)^{-1} & V_{\cdot,i} \geq V_{\cdot,i\text{-}1} \end{cases}$$

Where $L = R^T R$ and $i = 2 \ldots \S - 1$.

*Proof:* Recall that the problem of update V in matrix multiplication format is as follows:

$$\begin{aligned} \arg\min{}_V \frac{\beta}{2} \sum_{i=1}^{\mathcal{S}} \sum_{j=2}^{k-1} \left|\left|(V_{i,j} - V_{i,j-1})\, R^T\right|\right|_2^2 \\ + trace\left(y^{(2)}(\Theta - V)^T\right) + \frac{\rho}{2}||\Theta - V||_2^2 \\ + \sum_{i=2}^{k-1} y^{(3)}_{\cdot,i}(V_{\cdot,i-1} - V_{\cdot,i})^T \\ + \frac{\rho}{2}\sum_{i=2}^{k-1} ||\max(V_{\cdot,i-1} - V_{\cdot,i}, 0)||_2^2 \end{aligned}$$

Where $R_{i,i} = 1$ and $R_{i,adj(i)} = -\frac{1}{N_i}$, for $i = 1 \ldots \mathcal{S}$. $N_i$ is the total number of neighbors of task $i$.

For the sub-problem of solving $V_{\cdot,1}$, the analytical solution is fairly straight forward, since it is not involved in the max operator:

$$V_{\cdot,1} = y^{(2)}_{\cdot,1}/\rho + \Theta_{\cdot,1}$$

For each $V_{\cdot,i}(i > 1)$ there is a $max$ operator, thus the derivative with respect to $V_{\cdot,i}(i > 1)$ lies on two situations as follows:

$$\begin{cases} \beta(V_{\cdot,i} - V_{\cdot,i-1})L - y^{(2)}_{\cdot,i} + \rho(V_{\cdot,i} - \Theta_{\cdot,i}) \\ \qquad -y^{(3)}_{\cdot,i-1} + \rho(V_{\cdot,i} - V_{\cdot,i-1}) & V_{\cdot,i} < V_{\cdot,i-1} \\ \beta(V_{\cdot,i} - V_{\cdot,i-1})L - y^{(2)}_{\cdot,i} + \rho(V_{\cdot,i} - \Theta_{\cdot,i}) \\ \qquad -y^{(3)}_{\cdot,i-1} & V_{\cdot,i} \geq V_{\cdot,i-1} \end{cases}$$

Where $L = R^T R$ and $i = 2 \ldots k - 1$.

The above equations demonstrate that the analytical solution of $V_{\cdot,i}$ relies on $V_{\cdot,i-1}$. However, as we can obtain analytical solution for $V_{\cdot,1}$, we can get the analytical solution of $V_{\cdot,i}$ consecutively in ascending order. The analytical solution can therefore be computed as shown in Lemma 3. □

*4) Update y:* Finally, update $y^{(1)}, y^{(2)}, y^{(3)}$ as follows:

$$y^{(1)} = y^{(1)} + \rho(W - U);$$

$$y^{(2)} = y^{(2)} + \rho(b - V);$$

$$y^{(3)}_i = \max(y^{(3)}_i + \rho(V_{i-1} - V_i), 0), \text{ for } i = 2 \ldots \S - 1 \quad (31)$$

## V. Experiments

### A. Experiments for DETECTIVE-K

*1) Dataset and Experiment Setup:* The description of the datasets, settings (parameters and hyper-parameters), and sensitivity analysis for comparison models are included in the supplemental material, available online. In a nutshell, five datasets (Brazil, Colombia, Mexico, Paraguay, and Venezuela) from civil unrest forecasting and one dataset from air pollution event forecasting are used for the experimental evaluations. All the experiments were conducted on a 64-bit machine with Intel(R) core(TM) quad-core processor (i7CPU 2.5 GHz) and 16 GB memory. The hyper-parameters and network structure are chosen via a grid search based on model performance on the validation set. For all neural network based models, fully connected layers with sigmoid activation function are used. To evaluate the model performance, macro-average precision, recall and F1-Score are used here to provide an overall measure of model performance across all event subtype classes. In addition, we also introduce the Receiver Operating Characteristic (ROC) curve to further evaluate the overall prediction power. The performance of the proposed model is compared with baselines as well as existing state of the art methods, namely: *SVC1V1* (Support Vector Classifier with OneVsOne) and *SVC1VA* (Support Vector Classifier with OneVsAll) [19], *SR* (Softmax Regression) [4], [20], *SBM* (Shared-Bottom Model) [21], [22], *T-GCN* (Temporal Graph Convolutional Network) [23], and *MegaCRN* (Meta-Graph Convolutional Recurrent Network) [24].

*2) Performance:* Tables I and II show the performance for all the methods on all the datasets over all the event subtypes based on macro-average precision, recall and F1-score. For neural network based models the numbers attached along with the model name are the number of hidden layers, notice that DETECTIVE-K-SR is DETECTIVE-K framework used with Softmax Regression (i.e., without hidden-layers).

Table I shows that DETECTIVE-K framework used along with deep architectures performs consistently well across all the different countries, with DETECTIVE-K-2 achieving the highest scores in most cases. Specifically, DETECTIVE-K-2 attains an impressive F1 score in Colombia and Paraguay, indicating robust predictive capabilities. Conversely, traditional methods like SVC1VA, SVC1V1, and MLP display lower performance metrics. Additionally, advanced models like T-GCN and MegaCRN show moderate effectiveness but fail to outperform the DETECTIVE-K consistently. The DETECTIVE-K-2 method, in particular, demonstrates substantial improvement over others, particularly evident in Brazil and Paraguay, highlighting its efficacy in civil unrest detection tasks across diverse regions.

Table II also demonstrate the effectiveness of the proposed methods in the domain of air pollution event forecasting with different prediction lead times. DETECTIVE-K used along with deep architectures outperforms the baseline models consistently by 5%–10% in terms of the F1-score and achieves the top performance for both precision and recall. The results presented in this table also highlight the increasing difficulty of predictions with longer lead times, as forecasting long-term future events introduces considerably more uncertainty.

TABLE I
PERFORMANCE COMPARISON FOR THE CIVIL UNREST DATASETS (MACRO PRECISION, RECALL, AND F1)

| Method | Brazil | Colombia | Mexico | Paraguay | Venezuela |
|---|---|---|---|---|---|
| SVC1VA | 0.2318,0.2479,0.2368 | 0.2374,0.2673,0.2447 | 0.1798,0.1991,0.1738 | 0.2009,0.2396,0.2055 | 0.2136,0.2348,0.2069 |
| SVC1V1 | 0.2444,0.2582,0.2465 | 0.2062,0.2096,0.1995 | 0.1651,0.1600,0.1511 | 0.2058,0.2715,0.2152 | 0.2118,0.2481,0.2058 |
| SR | 0.2131,0.2525,0.2247 | 0.2496,0.2840,0.2545 | 0.1781,0.1888,0.1676 | 0.2212,0.2644,0.2287 | 0.2239,0.2507,0.2191 |
| DETECTIVE-K-SR | 0.2586,0.2699,0.2560 | 0.2568,0.2799,0.2645 | 0.2106,0.1897,0.1849 | 0.2378,0.2935,0.2402 | 0.2538,0.2661,0.2326 |
| MLP-1 | 0.2423,0.2358,0.2359 | 0.2369,0.2354,0.2357 | 0.1800,0.1957,0.1715 | 0.2160,0.3145,0.2234 | 0.2174,0.2200,0.2155 |
| MLP-2 | 0.2512,0.2575,0.2530 | 0.2594,0.2736,0.2634 | 0.1757,0.1534,0.1608 | 0.2300,0.2694,0.2307 | 0.2180,0.2311,0.2152 |
| MLP-3 | 0.2699,0.2590,0.2643 | 0.2400,0.2628,0.2436 | 0.1842,0.1539,0.1675 | 0.2133,0.2049,0.2084 | 0.2174,0.2200,0.2155 |
| SBM-1 | 0.2821,0.2634,0.2696 | 0.2956,0.2701,0.2762 | 0.2237,0.2051,0.2121 | 0.2447,0.3655,0.2543 | 0.2212,0.2115,0.2122 |
| SBM-2 | 0.2560,0.2737,0.2597 | 0.2919,0.2637,0.2732 | 0.2104,0.1951,0.2009 | 0.2363,0.2971,0.2416 | 0.2455,0.2505,0.2286 |
| SBM-3 | 0.2821,0.2637,0.2714 | 0.2759, 0.3176,0.2863 | 0.2060,0.1793,0.1908 | 0.2392,0.2459,0.2369 | 0.2545,0.1910,0.2162 |
| T-GCN | 0.2511,0.3798,0.2532 | 0.2451,0.2475,0.2463 | 0.1882,0.2154,0.1835 | 0.2540,0.2761,0.2610 | 0.2474,0.2718,0.2482 |
| MegaCRN | 0.2538,0.2635,0.2560 | 0.2159,0.2500,0.2317 | 0.1507,0.2000,0.1719 | 0.2380,0.3038,0.2628 | 0.1585,0.2000,0.1759 |
| DETECTIVE-K-1 | 0.2848,0.2804,0.2788 | 0.3067,0.2761,0.2845 | 0.2187,0.2070,0.2123 | 0.2467,0.3749,0.2562 | 0.2684,0.2422,0.2477 |
| DETECTIVE-K-2 | 0.3558, 0.2887,0.2779 | 0.2648,0.3130,0.2670 | 0.2252,0.2110,0.2176 | 0.2543,0.3373,0.2638 | 0.2704,0.2421,0.2471 |
| DETECTIVE-K-3 | 0.2828,0.2641,0.2712 | 0.2689,0.3152,0.2710 | 0.2081,0.2338,0.2000 | 0.2473,0.4482,0.2532 | 0.2178,0.2571,0.2174 |

Bold underlining means the best performance; only underlining without bold means thesecond-best performance.

TABLE II
CHINA AIR POLLUTION EVENT FORECASTING DATASET WITH VARIOUS PREDICTION LEAD TIMES (MACRO PRECISION, RECALL, AND F1)

| Method | 1-day | 3-days | 5-days | 7-days |
|---|---|---|---|---|
| SVC1VA | 0.4966, 0.5255, 0.5009 | 0.4362, 0.4768, 0.4309 | 0.3940, 0.3946, 0.3872 | 0.4334, 0.4553, 0.4240 |
| SVC1V1 | 0.5700, 0.5716, 0.5652 | 0.4532, 0.4849, 0.4565 | 0.4361, 0.4545, 0.4380 | 0.4351, 0.4412, 0.4302 |
| SR | 0.4254, 0.4338, 0.4287 | 0.4082, 0.4229, 0.4102 | 0.3949, 0.4208, 0.3974 | 0.4126, 0.4277, 0.4104 |
| DETECTIVE-K-SR | 0.5290,0.6436, 0.5572 | 0.4256, 0.6395, 0.4293 | 0.4281, 0.6350, 0.4236 | 0.4541, 0.6863, 0.4412 |
| MLP-1 | 0.5640, 0.5625, 0.5594 | 0.4679, 0.4809, 0.4614 | 0.4596, 0.4761, 0.4451 | 0.4646, 0.4684, 0.4592 |
| MLP-2 | 0.6108, 0.5567, 0.5693 | 0.4687, 0.4805, 0.4638 | 0.4378, 0.4359, 0.4308 | 0.4605, 0.4472, 0.4504 |
| MLP-3 | 0.5739, 0.5873, 0.5719 | 0.4989, 0.4916, 0.4902 | 0.4848, 0.4683, 0.4718 | 0.4597, 0.4537, 0.4364 |
| SBM-1 | 0.5710, 0.6162, 0.5812 | 0.5718, 0.5230, 0.5162 | 0.5692, 0.5075, 0.5134 | 0.5763, 0.5343, 0.4896 |
| SBM-2 | 0.5383, 0.5981, 0.5509 | 0.4630, 0.6396, 0.4880 | 0.4802, 0.6211, 0.4997 | 0.5070, 0.6457, 0.5256 |
| SBM-3 | 0.5284, 0.6085, 0.5526 | 0.5154, 0.5426, 0.5035 | 0.5089, 0.6331, 0.5236 | 0.5271, 0.5631, 0.5184 |
| T-GCN | 0.3303, 0.4485, 0.3423 | 0.0704, 0.2470, 0.1096 | 0.0696, 0.2389, 0.1078 | 0.0690, 0.2490, 0.1080 |
| MegaCRN | 0.2436, 0.4288, 0.2524 | 0.2560, 0.4270, 0.2555 | 0.2628, 0.4446, 0.2747 | 0.2432, 0.4311, 0.2461 |
| DETECTIVE-K-1 | 0.5558, 0.5668, 0.5560 | 0.4761, 0.5704, 0.5046 | 0.4878, 0.6562, 0.5085 | 0.4738, 0.6539, 0.4698 |
| DETECTIVE-K-2 | 0.5605, 0.6556, 0.5863 | 0.4932, 0.6186, 0.5290 | 0.4935, 0.5289, 0.4991 | 0.5627, 0.6390, 0.5868 |
| DETECTIVE-K-3 | 0.5979, 0.6364, 0.6002 | 0.5633, 0.5776, 0.5431 | 0.5256, 0.5851, 0.5300 | 0.5138, 0.6310, 0.5425 |

Bold underlining means the best performance; only underlining without bold means thesecond-best performance.

However, the proposed model behaves stably and suffers from less decline in terms of its overall performance compared with the other methods. For instance, the F1-score only decreases by about 10% for the DETECTIVE-K-3 model, while other baselines decrease by about 15%–30%. This may suggest that the proposed spatial regularization term in DETECTIVE-K improves the robustness of the deep model substantially, enabling it to capture more long-term dependencies of the data and the corresponding event subtypes.

The experimental results in both Tables I and II show that overall shallow models such as SR, SVM based models, and DETECTIVE-K-SR perform worse than deep models with hidden layers such as MLP, SBM and DETECTIVE-K-3. This is largely because shallow models cannot discriminate the subtle differences between event subtype patterns very well. Among the shallow models, DETECTIVE-K-SR still outperforms all other baselines most of the time, which further demonstrates the effectiveness of the proposed spatial regularization even on shallow models on various application domains. Moreover, DETECTIVE can forecast with a time GAP (Table II) while the lead time gap affects the sequence-based model such as TGCN.

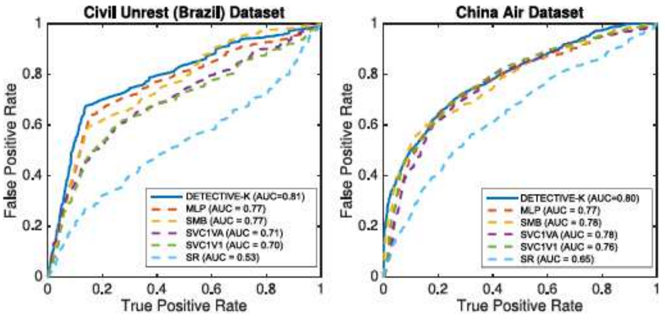


Fig. 5. Macro-average ROC comparison.

In the Receiver Operating Characteristic (ROC) curve analysis (Fig. 5) Brazil dataset is used to represent the Civil Unrest dataset, other datasets follow the similar trends. The China Air dataset has a lead time of 7 days. For neural network based models, only those giving the best AUC scores are shown here. The curves for the Civil Unrest dataset on the left clearly show that the DETECTIVE-K model achieves the best ROC

TABLE III
EVENT FORECASTING PERFORMANCE COMPARISON ON CIVIL UNREST DATASETS (MZE, MAE)

| Method | Argentina | Brazil | Chile | Colombia | Mexico | Paraguay | Uruguay | Venezuela |
|---|---|---|---|---|---|---|---|---|
| SVC1VA | 0.0368, 0.0708 | 0.0440, 0.0857 | 0.0657, 0.1129 | 0.0552, 0.0916 | 0.1284, 0,2284 | 0.0353, 0.0674 | 0.0223, 0.0390 | 0.0615, 0.1127 |
| SVC1V1 | 0.0339, 0.0670 | 0.0441, 0.0860 | 0.0610, 0.1098 | 0.0506, 0.0884 | 0.1187, 0.2184 | 0.0339, 0.0610 | 0.0227, 0.0403 | 0.0690, 0.1201 |
| SVMOP | 0.0392, 0.0709 | 0.0482, 0.0854 | 0.0740, 0.1189 | 0.0542, 0.0889 | 0.1187, 0.2184 | 0.0337, 0.0608 | 0.0239, 0.0398 | 0.0690, 0.1201 |
| POM | 0.0287, 0.0572 | 0.0626, 0.1230 | 0.0524, 0.0989 | 0.0376, 0.0724 | 0.0982, 0.1906 | 0.0367, 0.0717 | 0.0340, 0.0667 | 0.0374, 0.0722 |
| T-GCN | 0.0765, 0.0881 | 0.0736, 0.0905 | 0.0798, 0.1066 | 0.0293, 0.0459 | 0.2006, 0.2632 | 0.0326,0.0566 | 0.0444,0.0547 | 0.0519, 0.0693 |
| MegaCRN | 0.0115, 0.0228 | 0.0520, 0.0660 | 0.0426, 0.0676 | 0.0221, 0.0368 | 1.0000,0.1617 | 0.0314,0.0613 | 0.0076, 0.0141 | 0.0499, 0.0666 |
| MITOR-I | 0.0161, 0.0306 | 0.0344, 0.0665 | 0.0436, 0.0812 | 0.0280, 0.0534 | 0.0967, 0.1875 | 0.0284, 0.0551 | 0.0132, 0.0250 | 0.0289, 0.0551 |
| MITOR-II | 0.0158, 0.0305 | 0.0339, 0.0657 | 0.0436, 0.0812 | 0.0274, 0.0521 | 0.0875, 0.1690 | 0.0286, 0.0555 | 0.0122, 0.0231 | 0.0286, 0.0545 |
| DETECTIVE-§ | 0.0147, 0.0291 | 0.0348, 0.0674 | 0.0353, 0.0647 | 0.0197, 0.0366 | 0.0736, 0.1414 | 0.0259, 0.0499 | 0.0092, 0.0170 | 0.0226, 0.0425 |

Bold underlining means the best performance; only underlining without bold means thesecond-best performance.

curve, with an AUC score of 0.81. This is also the case for the China air pollution dataset, where the DETECTIVE-K model again achieves the highest AUC score of 0.80. This further demonstrates the effectiveness and overall prediction power of the proposed DETECTIVE-K.

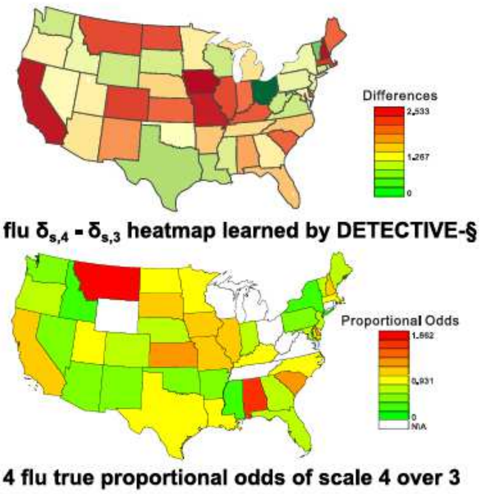
### B. Experiments for DETECTIVE-§

*1) Dataset and Experiment Setup:* The detailed description of the datasets, parameter settings and sensitivity analysis, introduction and hyper-parameter settings for comparison models are included in the supplemental material, available online. As a summary, in this study, 8 datasets from civil unrest forecasting and 2 datasets from influenza outbreak forecasting are used for the experimental evaluations. All the experiments were conducted on a 64-bit machine with Intel(R) core(TM) quad-core processor (i7CPU 2.5 GHz) and 16 GB memory. The hyper-parameters in the proposed model have been chosen based on the performance for the validation set. The validation set consists of a randomly chosen 20% of the training data. We used three fully connected layers (256, 128, 64) with relu activation function, and SGD for optimization. To evaluate the prediction performance for ordinal variables, Mean Zero-one Error (MZE) and Mean Absolute Error (MAE) are commonly used. MZE is the error rate of the classifier: $MZE = \frac{1}{N}\sum_{i=1}^{N}[\![y_i^* \neq y_i]\!] = 1 - Acc$, where $y_i$ is the true label, $y_i^*$ is the predicted label and $Acc$ is the accuracy of the classifier. MZE values range from 0 to 1; they are related to global performance, but do not consider the order. MAE is the average deviation in absolute value of the predicted rank $y_i^*$ from the true one $y_i$ [25]: $MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i^* - y_i|$. MAE values range from 0 to $k-1$ (maximum deviation in number of scales). The performance of the proposed models is compared with the baseline as well as the state-of-the-art methods, namely: *SVC1V1* (Support Vector Classifier with OneVsOne), *SVC1VA* (Support Vector Classifier with OneVsAll) [19], *SVMOP* (Support Vector Machines with OrderedPartitions) [26], *POM* (Proportional Odds Model) [27], *T-GCN* (Temporal Graph Convolutional Network) [23], *MegaCRN* (Meta-Graph Convolutional Recurrent Network) [24], and MITOR [28].

TABLE IV
EXPERIMENTAL RESULTS FOR U.S. FLU DATASETS (MZE, MAE)

| Model | 2011-2012 | 2013-2014 |
|---|---|---|
| SVC1VA | 0.2246, 0.3167 | 0.2861, 0.4367 |
| SVC1V1 | 0.2220, 0.3096 | 0.2869, 0.4368 |
| POM | 0.2250, 0.3117 | 0.3036, 0.4822 |
| SVMOP | 0.2269, 0.3118 | 0.2921, 0.4310 |
| T-GCN | 0.2541, 0.3231 | 0.3128, 0.4107 |
| MegaCRN | 0.1737, 0.1750 | 0.1815, 0.2172 |
| MITOR-I | 0.1148, 0.1900 | 0.1796, 0.3473 |
| MITOR-II | 0.1145, 0.1895 | 0.1794, 0.3466 |
| DETECTIVE-§ | 0.1098, 0.1827 | 0.1657, 0.2985 |

Bold underlining means the best performance; only underlining without bold means thesecond-best performance.



(a) 2013-14 flu $\delta_{s,4}$ - $\delta_{s,3}$ heatmap learned by DETECTIVE-§

(b) 2013-14 flu true proportional odds of scale 4 over 3

Fig. 6. The heat map for the US flu dataset for $b_{.,4} - b_{.,3}$ and ground truth proportional odds of class 4 over class 3.

*2) Performance:* Tables III and IV show the performance for all the methods on all the datasets based on both MZE and MAE. These indicate that the methods that utilize DETECTIVE-§frameworks perform better than most baseline methods overall. Table III shows that DETECTIVE-§consistently performs well across different countries, being the best in Argentina, Chile, Colombia, Mexico, Uruguay, and Venezuela and outperforming the baseline models by 4%–29% both in MZE and MAE. In the case of Brazil, the result degraded. One reason for that can be the language difference between Brazil and other countries. Table IV also demonstrate the effectiveness of the proposed methods. DETECTIVE-§outperformed the baseline models consistently by 4%–13% both in MZE. While MegaCRN has a low MAE, its high MZE suggests it often misses the exact scale of the event.

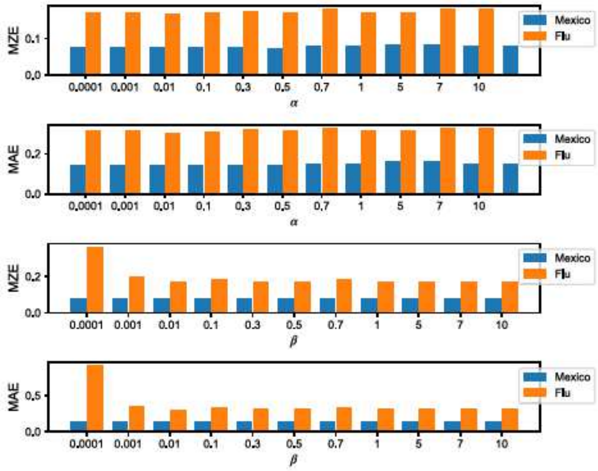


Fig. 7. Sensitivity analysis for hyper-parameters.

*3) The Effect of Scale Pattern Regularization:* Fig. 6 compares the scale patterns in terms of $b$ learned by DETECTIVE-§. Each of Fig. 6(a) shows the difference between 3rd and 4th thresholds $bi, 4 - bi, 3$ for each $i$th task (state) in the U.S. Fig. 6(b) shows the ground truth proportional odds of class 4 over class 3 for each of the states for two years, 2013 and 2014. Fig. 6(a) the patterns among nearby states is spatially smoothed, which is more similar to the patterns in ground truth shown in Fig. 6(b). This is because DETECTIVE-§can utilize threshold regularization to encourage the nearby states to share their knowledge with each other under the "first law of geography", which will largely alleviate each state's incompleteness of label set. For example, the pattern of the relatively small state "Colorado" suffered from data incompleteness and deviated from the neighbor states, but DETECTIVE-§corrected this, as compared with the ground truth in Fig. 6(b).

*4) Parameter Sensitivity Study of DETECTIVE-§:* There are two hyper-parameters in the proposed DETECTIVE-§model, as shown in (28), where $\alpha$ controls group sparsity $\ell_{2,1}$ norm and $\beta$ controls the proposed regularization term on $\theta$. Fig. 7 show the MZE and MAE of the model versus $\alpha$ and $\beta$ respectively. Only the results for Mexico within civil unrest datasets and 2013-14 influenza outbreak dataset are shown due to space limitations. The top 2 bar charts in Fig. 7 show the MZE and MAE of the model versus $\alpha$. By varying $\alpha$ across the range from 0.0001 to 10, the performance of the influenza outbreak dataset is stable, with the fluctuation ranges less than 0.01. For the civil unrest dataset, the fluctuation range is 0.015 in MZE and 0.03in MAE. The best performance is obtained when $\alpha = 0.5$. We can also see a clear trend where both MZE and MAE increase when $\alpha$ is too large or too small. The bottom 2 bar charts illustrate the MZE and MAE of the model versus $\beta$, which is varied across the same range as $\alpha$. The fluctuation ranges around 0.01 for both MZE and MAE. In general, the performance is good when $\beta$ is small, but deteriorates once $\beta$ becomes too large. A large $\beta$ will force the model to pay too much attention to being similar to its adjacent tasks and may thus lead to the loss of its own characteristic and a consequent decrease in overall performance.

## VI. Conclusion

Effective future event subtype forecasting can be utilized to qualitatively guide precautionary resource allocation and enable practitioners to take more precise preemptive measures. This work offers a unique spatial incomplete multi-task deep learning (DETECTIVE) architecture that addresses geographical heterogeneity, task label incompleteness, event subtype pattern correlations, and model adaptability in order to accomplish this goal. To deal with this non-convex and strongly coupled model objective, two effective algorithms were proposed. Comprehensive experiments on real-world datasets show that the suggested model performs better in a variety of application areas than alternative baseline methods.

## References

[1] X. Xia et al., "Pattern of spatial distribution and temporal variation of atmospheric pollutants during 2013 in Shenzhen, China," *ISPRS Int. J. Geo-Inf.*, vol. 6, no. 1, 2016, Art. no. 2.

[2] L. Zhao, F. Chen, C.-T. Lu, and N. Ramakrishnan, "Spatiotemporal event forecasting in social media," in *Proc. SIAM Int. Conf. Data Mining*, SIAM, 2015, pp. 963–971.

[3] J. Wang, Y. Gao, A. Züfle, J. Yang, and L. Zhao, "Incomplete label uncertainty estimation for petition victory prediction with dynamic features," in *Proc. IEEE Int. Conf. Data Mining*, 2018, pp. 537–546.

[4] Z. Chen et al., "Forecast oriented classification of spatio-temporal extreme events," in *Proc. Int. Joint Conf. Artif. Intell.*, 2013, pp. 2952–2954.

[5] Y. Ning, S. Muthiah, H. Rangwala, and N. Ramakrishnan, "Modeling precursors for event forecasting via nested multi-instance learning," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2016, pp. 1095–1104.

[6] N. Cressie, *Statistics for Spatial Data*. Hoboken, NJ, USA: Wiley, 2015.

[7] R. Bellman, *Dynamic Programming*. North Chelmsford, MA, USA: Courier Corporation, 2013.

[8] L. Wu, I. E.-H. Yen, F. Xu, P. Ravikuma, and M. Witbrock, "D2KE: From distance to kernel and embedding," 2018, *arXiv: 1802.04956*.

[9] K. Xu, L. Wu, Z. Wang, and V. Sheinin, "Graph2Seq: Graph to sequence learning with attention-based neural networks," 2018, *arXiv: 1804.00823*.

[10] P. McCullagh, "Regression models for ordinal data," *J. Roy. Stat. Soc. Ser. Methodol.*, vol. 42, pp. 109–142, 1980.

[11] Z. Yuan, X. Zhou, and T. Yang, "Hetero-ConvLSTM: A deep learning approach to traffic accident prediction on heterogeneous spatio-temporal data," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2018, pp. 984–992.

[12] S. Thrun and J. O'Sullivan, "Clustering learning tasks and the selective cross-task transfer of knowledge," in *Learning to Learn*. Berlin, Germany: Springer, 1998, pp. 235–257.

[13] L. Zhao, Q. Sun, J. Ye, F. Chen, C.-T. Lu, and N. Ramakrishnan, "Multi-task learning for spatio-temporal event forecasting," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2015, pp. 1503–1512.

[14] B. Haasdonk and C. Bahlmann, "Learning with distance substitution kernels," in *Proc. Joint Pattern Recognit. Symp.*, Springer, 2004, pp. 220–227.

[15] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.

[16] F. Kiaee, C. Gagné, and M. Abbasi, "Alternating direction method of multipliers for sparse convolutional neural networks," 2016, *arXiv:1611.01590*.

[17] J. Sun et al., "Deep ADMM-net for compressive sensing MRI," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2016, pp. 10–18.

[18] F. Bach et al., "Optimization with sparsity-inducing penalties," *Found. Trends Mach. Learn.*, vol. 4, no. 1, pp. 1–106, 2012.

[19] C.-W. Hsu and C.-J. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 415–425, Mar. 2002.

[20] N. M. Nasrabadi, "Pattern recognition and machine learning," *J. Electron. Imag.*, vol. 16, no. 4, 2007, Art. no. 049901.

[21] R. Caruana, "Multitask learning," in *Learning to Learn*. Berlin, Germany: Springer, 1998, pp. 95–133.

[22] R. Caruna, "Multitask learning: A knowledge-based source of inductive bias," in *Proc. 10th Int. Conf. Mach. Learn.*, 1993, pp. 41–48.

[23] L. Zhao et al., "T-GCN: A temporal graph convolutional network for traffic prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 9, pp. 3848–3858, Sep. 2020.

[24] R. Jiang et al., "Spatio-temporal meta-graph learning for traffic forecasting," in *Proc. AAAI Conf. Artif. Intell.*, 2023, pp. 8078–8086.

[25] S. Baccianella, A. Esuli, and F. Sebastiani, "Evaluation measures for ordinal regression," in *Proc. 9th Int. Conf. Intell. Syst. Des. Appl.*, 2009, pp. 283–287.

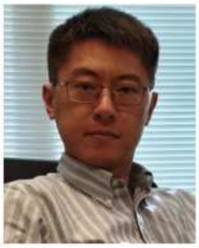
[26] W. Waegeman and L. Boullart, "An ensemble of weighted support vector machines for ordinal regression," *Int. J. Comput. Syst. Sci. Eng.*, vol. 3, no. 1, pp. 47–51, 2009.

[27] P. McCullagh and J. A. Nelder, *Generalized Linear Models, no. 37 in Monograph on Statistics and Applied Probability*. London, U.K.: Chapman & Hall, 1989.

[28] Y. Gao and L. Zhao, "Incomplete label multi-task ordinal regression for spatial event scale forecasting," in *Proc. 33rd AAAI Conf. Artif. Intell.*, 2018, Art. no. 366.

**Tanmoy Chowdhury** received the PhD degree in IT from George Mason University, in 2023. He is a Sr. programmer analyst with Richland County Government. His research interests include data mining, computer-aided design, natural language processing, computational modeling, and reasoning.

**Yuyang Gao** received the PhD degree in computer science from Emory University, in 2022. He is a data scientist working with the Home Depot Inc. His research focuses on data mining and ML techniques that can handle complex structured data, such as spatiotemporal and graph-structured data. Also interested in opening the 'black-box' of the deep learning models via designing the bio-inspired model architectures as well as via enhancing their interpretability and explainability through new techniques.



**Liang Zhao** received the PhD degree from Computer Science Department, Virginia Tech, in 2016. He is an associate professor with the Department of Computer Science, Emory University. His research interests include data mining, artificial intelligence, and machine learning, with special interests in spatiotemporal and network data mining, deep learning on graphs, nonconvex optimization, and interpretable machine learning.