Downloaded 06/06/23 to 130.215.207.14 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

HintNet: Hierarchical Knowledge Transfer Networks for Traffic Accident Forecasting on Heterogeneous Spatio-Temporal Data

Bang An* Amin Vahedian† Xun Zhou* W. Nick Street* Yanhua Li‡

Abstract

Traffic accident forecasting is a significant problem for transportation management and public safety. However, this problem is challenging due to the spatial heterogeneity of the environment and the sparsity of accidents in space and time. The occurrence of traffic accidents is affected by complex dependencies among spatial and temporal features. Recent traffic accident prediction methods have attempted to use deep learning models to improve accuracy. However, most of these methods either focus on small-scale and homogeneous areas such as populous cities or simply use sliding-window-based ensemble methods, which are inadequate to handle heterogeneity in large regions. To address these limitations, this paper proposes a novel Hierarchical Knowledge Transfer Network (**HintNet**) model to better capture irregular heterogeneity patterns. HintNet performs a multi-level spatial partitioning to separate sub-regions with different risks and learns a deep network model for each level using spatio-temporal and graph convolutions. Through knowledge transfer across levels, HintNet archives both higher accuracy and higher training efficiency. Extensive experiments on a real-world accident dataset from the state of Iowa demonstrate that HintNet outperforms the state-of-the-art methods on spatially heterogeneous and large-scale areas.

1 Introduction

Traffic accident is a major safety concern in modern society. In the United States, it is estimated that nearly 40 thousand people died in traffic accidents in 2020, according to the National Highway Traffic Safety Administration (NHTSA) [20]. This number shows an increase of around 7% compared to 2019, despite the effects of the coronavirus pandemic on mobility. Moreover, the US Census Bureau found that the average commuting time of Americans has increased by 10% between 2006 and 2019 [9]. With nearly 85% of Americans driving for their commute [10] and the increased average commute time, the number of fatalities can continue to grow in the coming years. Therefore, the ability to forecast accidents is of significant importance to the members of society, as it can help trigger public safety preparedness as well as alert drivers to the potential risk of accidents at certain locations at certain times in the future. The availability of large-scale data on accidents, climate, road dynamics, and other spatial and temporal characteristics has enabled the researchers to propose machine learning approaches to traffic accidents.

Accidents are a relatively rare phenomenon. Of all the location-time instances, very few of them contain a traffic accident. As a result, it is challenging for a machine learning method to learn the complex patterns leading up to an accident, with the presence of an overwhelming majority of times and locations that have zero accidents. Moreover, such patterns are not likely to be homogeneous over space and time. The patterns that predict traffic accidents in a crowded urban area likely differ from such patterns in rural areas. In other words, the patterns that may predict traffic accidents are heterogeneous. Given how challenging it is to learn such patterns in the first place (due to the overwhelming imbalance between accident, no-accident), it is even more challenging to learn all these spatially heterogeneous patterns in the same model.

Researchers have tackled the accident prediction problem with a variety of approaches. A large body of literature has explored solutions with non-machine learning methods, or has proposed straightforward applications of data mining techniques to predict accidents [1–5, 8]. Such techniques do not address the challenges of sparsity and heterogeneity. More recently, researchers have proposed deep learning techniques [6, 12] and have taken advantage of LSTM methods and attention mechanisms to learn temporal patterns, and have used convolutional methods to learn local spatial patterns [21]. Wang et al. [15] proposed GSNet to capture multi-scale spatial-temporal dependencies by applying a geographical module and a semantic module. Zhou et al. [18] handled the sparsity issue using a transformation strategy to discriminate the risk values, dominated by zero or near-zero values. None of these methods directly address the spatial heterogeneity with mixed urban and rural areas, and only rely on the model to learn such patterns from spatial features. Notably, Yuan et al. [16] proposed a Hetero-ConvLSTM model to predict traffic

*University of Iowa, {bangan, xunzhou, nickstreet}@uiowa.edu
†Northern Illinois University, avahediankhezerlou@niu.edu
‡Worcester Polytechnic Institute, yli15@wpi.edu
Xun Zhou is the corresponding author

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 06/06/23 to 130.215.207.14 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

accidents in a grid setting. Their method addresses the heterogeneity challenge by an ensemble of predictions from 21 pre-selected sub-regions in a sliding-window manner. However, the pre-selected windows could not fully capture heterogeneity as they ignore the underlying spatial patterns. Moreover, it does not utilize the shared knowledge across windows and has a high computational cost due to the ensembles.

In this paper, we propose **HintNet**, a **Hi**erarchical K**N**owledge **T**ransfer **Net**work model to address the spatial heterogeneity problem. HintNet first employs a hierarchical spatial partitioning method to systematically group regions with potentially similar risk patterns together without needing prior knowledge of the regions. Then, separate models for each level of the hierarchy are trained, allowing the unique patterns of each level to be learned separately. Moreover, we argue that despite heterogeneity, there is also a pattern of traffic accidents that is common among all levels. We develop a knowledge transfer method to allow the models to share knowledge across the different levels of the hierarchy. We take advantage of the expansive set of measured and derived features from numerous datasets from the state of Iowa to train HintNet. Our extensive evaluations show that HintNet is successful in outperforming the state-of-the-art baselines. Moreover, our evaluations show that our proposed knowledge transfer mechanism can improve the prediction accuracy and shorten the training process of the models significantly. Our contributions are as follows:

- We propose a hierarchical space partitioning framework to automatically group regions with potentially different accident patterns.
- We propose a deep neural network to predict the traffic accidents of a region based on spatial, temporal and spatio-temporal features with jointly-trained graph convolution and LSTM modules.
- We propose a knowledge transfer mechanism to share the common pattern of accidents across regions among all the models to expedite the training and improve accuracy.

The rest of the paper is organized as follows: In the next section, we discuss the related work. Next, we present an overview of the dataset and extracted features as well as preliminary concepts and a problem formulation. Then, we present our solution, HintNet, followed by the experiments and the conclusion.

2 Related Work

A large body of work has taken a straightforward approach of simply applying existing prediction models to the accident prediction problem including ANNs and decision trees [4, 5], random forest ensemble [13], or Probabilistic Neural Networks [1]. Caliendo et al. developed multiple regression models based on Poisson, Negative Binomial, and Multinomial analyses to predict the number of accidents in given roads [3]. Many other works used regression and correlation methods to predict the number of accidents in general and special cases [2, 8]. However, these works only applied existing methods to the problem, and do not offer methodological contributions that address the challenges specific to the traffic accident prediction problem.

The use of deep learning [6, 21] to predict traffic accidents is relatively recent. However, most of these early attempts only use data from a single view (e.g., spatial or temporal) for prediction, therefore unable to fully capture the spatiotemporal patterns. Wang et al. [15] proposed GSNet with a geographical module and a semantic module to capture a diverse set of features to learn the patterns. The geographic features aim to allow the model to learn the heterogeneity in space. However, their method does not fully address the heterogeneity issue, as they are still relying on the model to capture it through learning patterns in the features. Zhou et al. [18] offers a novel transformation of zero-accident instances to handle the sparsity issue. However, they also do not fully address the heterogeneity problem and rely on a single model to learn the spatial heterogeneity directly from features, even though the number of samples with accidents is limited due to sparsity. Yuan et al. proposed Hetero-ConvLSTM [16], which explicitly addresses spatial heterogeneity in accident prediction. This deep learning method divides the spatial field into 21 distinct regions. They then build an ensemble method to predict the number of accidents. This method addresses the temporal patterns and spatial heterogeneity. However, the manually-selected regions do not necessarily reflect different prediction patterns.

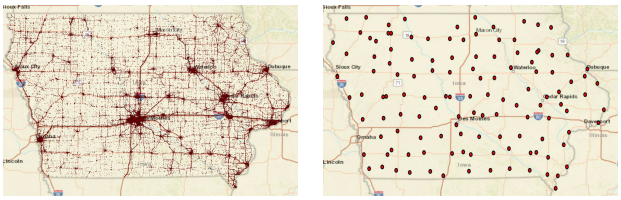
Distinct from all the above methods, our HintNet method fully addresses the spatial heterogeneity problem using an automatically generated hierarchical partitioning of the space, a deep learning network with spatial, temporal, and spatio-temporal features and using a knowledge transfer framework to train a diverse set of models to capture the heterogeneous patterns of accidents in space and time.

3 Preliminary Concepts and Overview

In this section, we introduce the data sources, extracted features, and our problem formulation. Our data covers the entire state of Iowa, which is a suitable place to study traffic accident forecasting problem due to the heterogeneous environment with both rural and urban areas.

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 06/06/23 to 130.215.207.14 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy



(a) traffic accidents in Iowa (b) COOP Stations

Figure 1: Visualization of traffic accidents and COOP data

3.1 Data Sources The data we collected are all within the range of Iowa from the year 2016 to 2018. We collect data from the following sources: **(1) Vehicle Crash data** is collected by the Iowa Department of Transportation(DOT)[1]. The data contains the 168,964 crash records from the year 2016 to 2018. The records include the time and location of each crash. Figure 1 (a) shows the mapping of the traffic accident records in the state of Iowa. **(2) RWIS (Roadway Weather Information System)**[2] is an atmosphere monitoring system with 86 observation stations located at the state primary roads. **(3) COOP (National Weather Service Cooperative Observer Program)**[3] is maintained by National Weather Service to monitor weather information. Unlike RWIS, COOP concentrates on weather data such as precipitation, snowfall, and snow depth. Figure 1 (b) demonstrates the locations of the observation stations. **(4) POI.** The Point-of-Interest data are collected from HERE MAP API[4]. We collected the 13 categories of POI with their latitude and longitude. **(5) Iowa Road Networks.** From Iowa DOT OPEN DATA[5], we obtained Iowa road network data with basic road information. It consists of the speed limit, estimated Annual Average Daily Traffic volume for the primary roads and secondary roads. **(6) Traffic Camera Data.** The real-time traffic condition data were collected from 128 camera stations along state highways.

3.2 Definitions and Feature Extraction Next, we define concepts needed to formulate our problem and then explain the features extracted for the prediction task. Finally, we present our problem definition.

Definition 3.1. A spatio-temporal field $L \times T$, where $L = \{l_1, l_2, ..., l_m\}$ is a grid, where each grid cell l_i is a $d \times d$ square area. $T = \{t_1, t_2, ..., t_n\}$ is a study period partitioned into equal time intervals (e.g., hours, days).

We map all the features and the accidents onto the grid L and over time T. We use $C(l,t)$ to denote the total accident count in location l during time t. Depending on the dimensions of the features, we have spatial, temporal, and spatio-temporal (ST) features, as defined later in this section.

Definition 3.2. Road Network Mask Map H is a binary mask layer created by mapping the road network with primary and secondary roads onto the grids. We use a spatial mask to indicate if a grid cell contains any road segments (1) or not (0).

Definition 3.3. Temporal Features Temporal features F_T include the day of the week, day of the year, and the month of the year, whether this is a weekend, and whether this is a holiday. $F_T^i(l,t)$ represents the i-th temporal feature at time interval t at location l. Note all cell locations l in the study area share the same temporal features in each time interval t.

Definition 3.4. Spatial Features Spatial features F_S include static features that do not change over time, where $F_S^i(l,t)$ represents the i-th spatial features for location l at time interval t. These features include the point of interest (POI), basic road network information, and spectral features [16]. Specifically, every POI feature is represented by the frequency of each POI category in a grid cell l. Road network features include basic road conditions such as Annual Average Daily Traffic (AADT), average traffic speed, etc. To better address the spatial heterogeneity problem, we use an idea proposed by Yuan et al. [16] and apply the spectral analysis on the road networks to generate 10 spectral features for each grid cell, which contain spatial connectivity relationships between different locations through the road network.

Definition 3.5. Spatio-Temporal (ST) Features Spatio-Temporal features F_{ST} are those, which vary both in space and time, where $F_{ST}^i(l,t)$ represents the i-th ST feature for location l at time interval t. F_{ST} includes daily weather and traffic conditions in each location l and each time slot t. Weather features consist of precipitation, snowfall, snow depth, etc. The weather features are continuously distributed over the entire space. The traffic condition features include average traffic speed, normal vehicle traffic volume, and truck traffic volume for each location and time interval.

Missing value imputation: Many ST features are only collected at sampling sites or stations. There are also missing values due to data quality issues. To utilize the data for the entire study area, we use spatial interpolation methods to impute the missing values. Specifically, we use Ordinary Kriging [7] to estimate the weather-related attributes for locations without a station and a Universal Kriging [7] with network distance to estimate traffic-related features.

[1] https://icat.iowadot.gov/#
[2] https://mesonet.agron.iastate.edu/RWIS/
[3] https://mesonet.agron.iastate.edu/request/coop/obs-fe.phtml
[4] https://developer.here.com/documentation/places/dev_guide/topics/categories.html
[5] https://data.iowadot.gov/datasets/f07494c9bc6048d8a34c50af400f22641

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 06/06/23 to 130.215.207.14 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

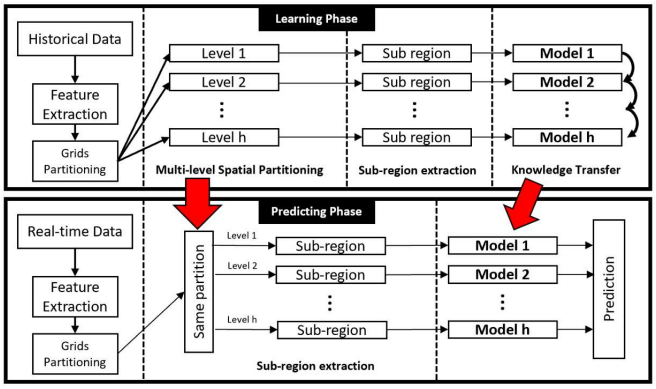


Figure 2: The HintNet Solution Overview.

In total, we extracted 47 features, including 29 spatial features, 5 temporal features, and 13 ST features for each grid cell l and time interval t.

3.3 Problem Definition

Now we are ready to define the problem formally.

• Given:
– A spatial-temporal field $L \times T$
– A road network mask map H
– Traffic accident count tensor C for a time window $[t-n, t-1]$ for all $l \in L$, $n < t$
– A set of feature tensors $F = \{F_T, F_S, F_{ST}\}$ for the same time window for all the locations $l \in L$
• Find:
– Predicted accident count in every $l \in L$ for t: $\hat{C}(l,t)$
• Objective:
– Minimize the prediction error
• Constraints:
– All traffic accidents occur along road system.
– Spatial heterogeneity exist in the data.

Here n is the time length of the input features for each prediction. In this paper, we choose t as a single day and use seven consecutive days of data to predict for the eighth day ($n = 7$).

4 Proposed Solution

In this section, we present our solution HintNet. Figure 2 shows the overview of the proposed framework. In the learning phase, we first perform the Multi-level spatial partitioning to obtain levels of regions with different risks. Meanwhile, the sub-regions are extracted and used to train models on each risk level. Lastly, the knowledge learned from previous models is transferred to the next level by initializing model parameters. In predicting phase, the real-time data is extracted and mapped into grids, then we use partitioned results from the training phase to classify grids into the same levels. Finally, features are fed into corresponding well-trained models to make final predictions.

4.1 Multi-level Risk-Based Spatial Partitioning

To address the limitations of related work in capturing irregular spatial heterogeneity patterns, we propose a spatial partitioning method, namely, Multi-level Risk-Based Spatial Partitioning (M-RSP) to partition the grids into irregular-shaped regions based on the accident risk in a hierarchical manner. Specifically, M-RSP applies binary-partitioning iteratively. In each step, the study area is split into a risky region and less-risky region. In the next iteration, M-RSP repeats this binary-partitioning process on the previous risky region. By doing this iteratively, we obtain multiple levels of partitioned less-risk regions. Each level of less-risk regions represents a hierarchy of the risk distribution. To distinguish risk levels, we use a threshold η to compare with the total number of accidents in each partitioned region. If the accident count is greater than η, this region is risky. Otherwise, the region is less-risky.

The partitioning procedure for every single level, which we refer to as RSP (Risk-Based Spatial Partitioning), is inspired by DBSCAN [11], and designed for binary-partitioning on grids with similar risk. Given a cell location l, its **neighbors** are defined as cells within a **range** ϵ both horizontally and vertically in Manhattan distance. **Min_points** γ is a threshold used to identify grid cell as **high-risk**, if its accidents count is greater than γ. **Min_Risk** λ is used for filtering out noise cells within fewer accidents than this limit. A **critical cell** is defined as a cell with the number of high-risk neighbors including itself greater than a threshold **min_neighbors** β. A **border cell** is a non-critical cell with at least one critical cell in the neighbors. An **outlier** is not a critical cell and has no critical cells among its neighbors. In RSP, grid cells are classified as Critical cells, Border cells, or Outliers. Specifically, for each level, RSP starts with checking a random cell. If this cell is identified as a Critical cell, and then its neighbors will also be checked until no Critical cell is identified. The border cell and critical cell are assigned with a partition label. The algorithm repeats this process to the next random unclassified grid cell until all grids are classified. Importantly, We have this unique design of identifying Critical cells by counting their high-risk neighbors. The reason behind this design is that if we simply count the total accidents within a region, some low-risk grid cells with an extremely high-risk neighbor cell will be classified as high-risk cells as a result of over-influence from that high-risk neighbor. Lastly, grid cells with no road are filtered out by using the mask map H.

The overall M-RSP algorithm calls for the single-level RSP procedure iterative for partitioning on each level for β iterations until all the levels are generated. Algorithm 1 shows the details of the entire M-RSP,

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 06/06/23 to 130.215.207.14 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Algorithm 1: Multi-level Risk-based Spatial Partitioning (M-RSP)

Input: accidents matrix A, mini_points γ, min_risk λ, threshold η, epsilon ϵ
Output: matrix with assigned level label

```
1  iter = (2 * ε + 1)^2
2  Initialize zero map matrix z
3  Initialize result as −1 map matrix
4  for each β from 0 to iter do
5      Partitions = RSP(A, ε, γ, λ, β)
6      if β == iter then
7          η = +∞
8      for each p in Partitions do
9          for each g in p do
10             if p[g] == 0 and z[g] == 0 then
11                 z[g] = 1
12                 result[g] = β − 1
13         ctr = CountAccidents(p, A)
14         if ctr <= η then
15             for each g in p do
16                 if z[g] == 0 then
17                     z[g] = 1
18                     result[g] = β
19 result += 1
20 return result
```

where Line 1 determines the maximum number of iterations, which equals to maximum neighbors controlled by ϵ. Line 5 performs the partitioning for a single level using RSP. Line 6 checks for the last iteration to partition all left regions by setting η as infinity. The remaining algorithm implements binary-partitioning by checking with η. In the end, the result is incremented by one because the noise level was initialized as negative ones. Figure 3 illustrates this process of multi-level partitioning on the entire Iowa dataset. In the binary tree, levels represent partitioned maps with β from 0 to 9 when ϵ is set as 1. The parent node represents the unassigned cells from the previous level. The left child represents the assigned grids in the current level, and the right child node represents unassigned grids. The right-most map on the top is the final risk-based partitioning map. Lastly, depending on the granularity of the partition result we need, we can aggregate every k levels together into a single partition, where k is a tunable hyperparameter in our framework. The finest granularity is when $k = 1$, which is the same as using the original partitioned result.

4.2 HintNet deep learning solution Given the partitioned regions on each level of the hierarchy, separate models are trained on each level to reduce the spatial heterogeneity issue. However, simply applying separate models ignores the potential connectivity between regions on each level of the hierarchy. Thus, we develop a knowledge transfer mechanism to allow models to share learned knowledge across different levels. Figure 4 shows the basic structure of HintNet. The inputs include spatial features, temporal features, spatial-temporal features, and an adjacency matrix from the same region. The output of HintNet is the predicted number of accidents. Firstly, The Graph Convolution(GC) is carefully designed to capture the local spatial auto-correlations along with the road system. Meanwhile, the training process should remain reasonably efficient. To achieve that goal, sub-regions are extracted from each level. In every time interval t, we treat each cell l and its surrounding neighbor cells as one $w \times w$ image, where the size w, a hyperparameter, controls the filter size of Graph Convolution. The sub-region grids are converted into a graph, where cells are treated as vertices and correlations between cells are treated as edges. Inspired by the dynamic CNN [17] proposed by Zhang et al., We use the Pearson correlation coefficient in Equation 4.1 to quantify the accident correlations between grid cells to capture the spatial dependencies.

$$a_{XY} = \frac{\sum_{i=1}^{n}(X_i - \overline{X})(Y_i - \overline{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \overline{X})^2 \sum_{i=1}^{n}(Y_i - \overline{Y})^2}} \tag{4.1}$$

where the adjacency matrix A is a symmetric $w^2 \times w^2$ matrix such that its element a_{XY} is the accident correlation between location X and Y. Thus, for each cell l and time interval t, a feature image tensor $X_l^t \in \mathbb{R}^{w \times w \times d}$ is retrieved, where d is the dimension of the features. The inputs of the Graph Convolution layer are X_l^t for each cell l and corresponding adjacency matrix A, and the output is output feature matrix H_l.

$$H_l^i = f(H_l^{i-1}, A) = \sigma(AH_l^{i-1}W_i), \tag{4.2}$$

Where H_l^i is the output feature matrix of the region l in the i-th layer. In this way, Graph Convolution filters focus more on regions with higher correlations along with the road network and ignore the regions with irrelevant traffic accident patterns.

Secondly, we use Long Short-Term Memory(LSTM) [19] as building blocks. To capture the spatial dependencies, Graph Convolution layers are first applied on spatial-temporal features to obtain filtered feature matrix H_l in each time interval t. Afterward, we concatenate the last output feature matrix H_l with temporal

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 06/06/23 to 130.215.207.14 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

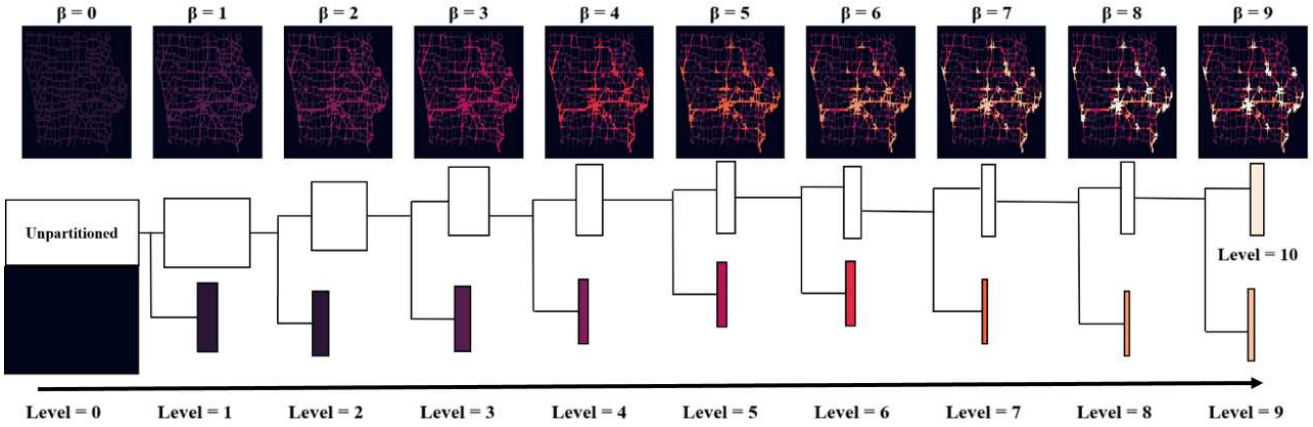


Figure 3: **Multi-level partitioning**. White box - unpartitioned regions that will be checked in the next level. Colored box - partitioned regions, size indicating the number of partitioned grid cells in corresponding level.

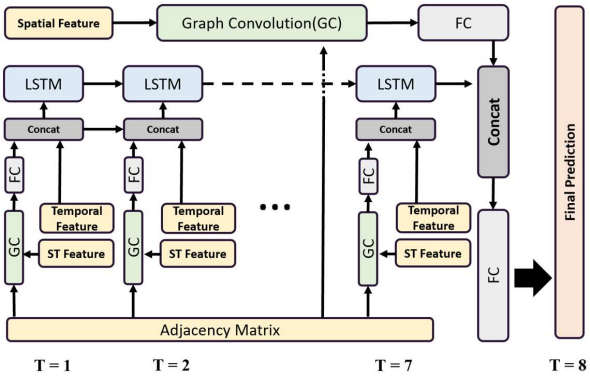


Figure 4: HintNet Deep Learning framework.

features and feed it into a fully connected layer, and then the output is used as input of each LSTM state. Concurrently, Spatial Features are also fed into a Graph Convolution layer, and another fully connected layer is applied on its output feature matrix to get low dimension representation. Lastly, the representation of spatial features is concatenated with the output of the last LSTM state, and then we fed the concatenation into the last fully connected layer to make final predictions.

Cross-level Knowledge Transfer We argue that there exist some common traffic accident patterns in all levels of hierarchy, especially for particular adjacent levels. For example, the risk factors recognized in downtown areas such as holiday events can be transferred to nearby regions, because they tend to share similar traffic patterns. To transfer learned knowledge across levels, we transfer model parameters learned from the previous level to the model in the next level. Compared with initializing model parameters randomly, we argue that the transferred parameters learned from other levels carry the experience of forecasting accidents and contribute to the training process of other levels. The right part of Figure 2 shows this process. In our case, the knowledge is transferred from the level of urban areas to the level of rural areas (i.e., from leaf to root). During the training phase, we apply the gradient descent to update parameters θ, with a learning rate α. The training process uses the mean square error (MSE) as a loss function.

$$Loss = \frac{1}{T}\sum_{t=1}^{T}(Y_t - \hat{Y}_t)^2, \tag{4.3}$$

Where Y_t is the ground truth and $\hat{Y}_t$ is the predicted values of all grid cells at time interval t.
Algorithm 2 shows the training process of the cross-level knowledge transfer. The output contains trained models on each level, and they are used in Predicting Phase. In the Predicting Phase, real-time features in each risk level are fed into corresponding trained models and make predictions on each level. The final prediction of HintNet is the integration of predictions on each level.

5 Evaluation

In this section, we demonstrate the effectiveness of our proposed method comparing with baselines.

5.1 Experiment Settings In this part, we will explain the basic setting of our experiments.
Data Preprocessing: We use the data from the first two years (2016-2017) as a training set, and the validation set is randomly selected from 20% of the training set. The data from the year 2018 is used as a testing set. Besides, the whole state of Iowa area is partitioned into 5km by 5km grids.

Evaluation Goals: (1) Does the proposed framework outperforms baseline methods with different levels of heterogeneity? (2) Which features have the most impact on prediction accuracy in different levels. (3) Which granularity k of partitioning results in the best performance? (4) Does the knowledge transfer mecha-

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 06/06/23 to 130.215.207.14 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Algorithm 2: Cross-level Training

Input: Multi-level partitioning τ, predictor $f()$, True accidents Y, learning rate α, Epoch e

Output: trained models on each level

initialize parameter θ in *pool* for each level
for *each level v in* τ **do**
  if *not 1st v* **then**
    $\theta = pool$[previous v]
  for *each iteration in e* **do**
    $F_T, F_S.F_{ST}, A = \text{SubregionExtract}(v)$
    $\hat{Y} = f_\theta(F_T, F_S.F_{ST}, A)$
    $Loss = \frac{1}{T}\sum_{t=1}^{T}(Y_t - \hat{Y}_t)^2$
    Calculate gradient $\nabla g(\theta)$ by $Loss$
    Update $pool[v] = pool[v] + \alpha \nabla g(\theta)$
return *pool*

nism improve the model performance?

Metrics: We evaluate the performance of the models by measuring the mean square error (MSE) Besides, we use the number of model parameters to demonstrate the resources usage between the proposed model and baselines.

Baselines: We compare our proposed framework with the following baselines: (1) Least Square Linear Regression (2) FC-LSTM, two layers of LSTM (2) Decision Tree Regression. The max depth is set to 30. (3) ConvLSTM [14]. We use a two-layer structure and the hidden dimension is equal to the number of features. (4) Hetero-ConvLSTM. Each ConvLSTM in hetero-ConvLSTM uses the same parameter setting as the ordinary ConvLSTM baseline. We use multiple moving windows with fixed size 32×32. The number of windows depends on the size of the study region. (5) GSNet. we modify the weighted loss function part to fit into our case. (6) Historical Average. Historical average daily accident counts. The mask map is also applied on all baseline predictions to filter out cells without roads.

5.2 Performance Comparison We compare the performance between the proposed method and baselines given with different levels of spatial heterogeneity. We test the models in 4 types of regions with grid size 16×16, 32×32, 64×64, and 128×64 separately. Figure 5 shows the corresponding grid map representing homogeneous, less-homogeneous, heterogeneous, severely heterogeneous regions respectively. To test all methods on them, the filter size of ConvLSTM and the sub-region size in our method is set as 5×5 in the homogeneous region (16×16) and less-homogeneous region (32×32)

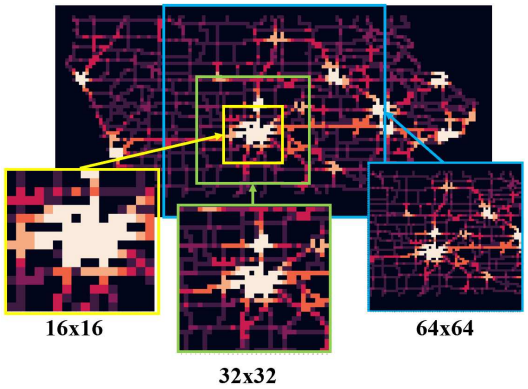


Figure 5: Illustration of four test regions

grid map. For heterogeneous region (64×64) and severe-heterogeneous region (128×128), the both sizes are set as 7×7. Besides, GSNest is infeasible to be applied on the largest region (128×64) because of its oversized graph convolution part. Therefore, we applied two independent GSNet models on the left-half and right-half parts of the study area, and calculate the total prediction error. For Hetero-ConvLSTM, the number of moving windows used for each region are 1, 1, 9, 21 respectively, as a result of its fixed window size.

Table 1 compares models in all heterogeneity levels, where HintNet outperforms all baseline methods. State-of-the-art methods like GSNet and ConvLSTM achieve decent prediction in smaller regions with less spatial heterogeneity but perform poorer in large, heterogeneous regions. In this case, HintNet outperforms GSNet 43%. On the opposite, our method and hetero-ConvLSTM address the heterogeneity problem and perform stably in all regions. Nevertheless, Hetero-ConvLSTM suffers from an excessive number of sub-models, square-shaped partitioning with a fixed size, and dis-connectivity between sub-models. In a less-homogeneous region, our method outperforms hetero-ConvLSTM by 26% in prediction error. **Model Complexity:** The experiments show that HinNet achieves better performance with a relatively smaller model size. The results of model complexity are included in supplementary materials. The supplementary material and code is available at: `https://github.com/BANG23333/HintNet`.

5.3 Ablation study In this section, we first evaluate HintNet's performance on each level of the hierarchy, then compare the impact of features and parameters.

Improvements on each level: We study how our proposed model performs in different levels of the partition. The Historical Average(HA) is generally a good reference value to measure the overall improvements of HintNet in each level. We calculate the percent of improvements between our proposed model and the historical average in each level. Figure 6 shows the trend of improvements from rural regions to urban regions.

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 06/06/23 to 130.215.207.14 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Table 1: Model Comparison

	LR	DTR	LSTM	ConvLSTM	GSNet	Hetero-ConvLSTM	HA	HintNet
16×16	0.221	0.388	0.264	0.197	0.179	0.197	0.220	**0.174**
32×32	0.075	0.133	0.085	0.081	0.080	0.081	0.074	**0.060**
64×64	0.032	0.085	0.033	0.044	0.036	0.028	0.032	**0.025**
128×64	0.026	0.033	0.027	0.039	0.037	0.024	0.027	**0.021**

Table 2: Error Comparison with different k

	k = 3	k = 2	k = 1
MSE	0.063	0.021	0.135

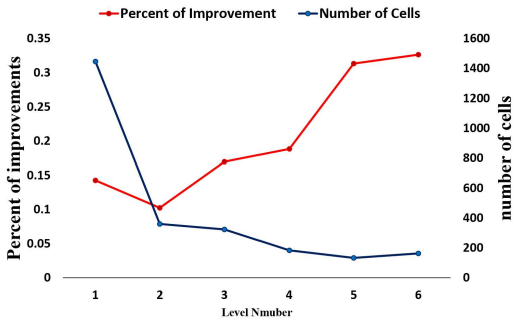


Figure 6: Improvement across levels

The red line represents the percentage of improvement and the blue line represents the number of grid cells involved in each level. As we can see, the improvements grow steadily as the level number grows. This indicates that HintNet makes relatively better predictions in urban areas and suburban areas, but only slightly exceeds HA in rural areas partially due to data sparsity.

Impact of feature group: To determine the effectiveness of different feature groups on the results, we examine the results by adding feature groups one by one. As Table 3 illustrated, the model with only spatial features (S) has slightly better performance than the historical average. With extra temporal features (T) such as calendar data, our model makes a great improvement on level 6 which represents the downtown areas. It reveals that calendar features enable the model to capture the temporal patterns in areas with frequent human activities such as traffic jams in holidays. Interestingly, the spatial-temporal (ST) features like weather information brings down the errors on level 5, level 6, and level 4 significantly. This indicates that the dynamic weather changes play a significant role in predicting accidents in state highways and residential areas which covers the large road systems with high-speed traffic volumes. Lastly, the prediction on extreme rural areas including level 2 and level 1 is still challenging due to the randomness of accidents.

Impact of risk-tree level aggregation granularity k: To investigate which degree of partitioning

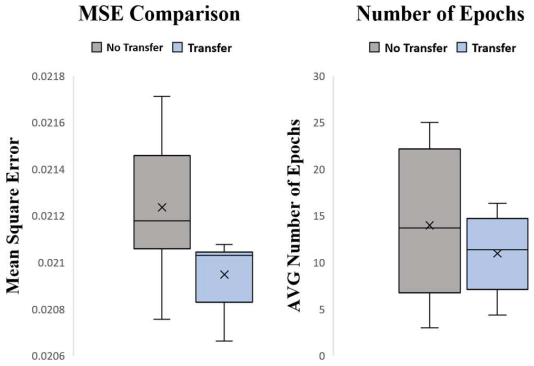


Figure 7: Knowledge Transfer

granularity leads to best model accuracy. We test the model when k is set to be 3, 2, and 1. As shown in Table 2, when $k = 2$ we obtain the best result. When the $k = 3$, there are only 3 partitioned levels. The cells with different accident patterns are mixed in a single group, so the model cannot address the spatial heterogeneity appropriately. Similarly, the model has the worst results when $k = 1$. In this case, the over-fined granularity results in a limited number of samples in each level, and it becomes extremely difficult for our model to learn from levels with more variability.

5.4 Impact of cross-level knowledge transfer

We assume that the inherent knowledge learned from levels helps the training process on other levels. We examine the benefits of using the knowledge transferring mechanism based on its accuracy and efficiency. To check the improvements in prediction accuracy, we train our models with and without knowledge transfer mechanism under the same parameter setting for 10 times, and then we draw a box-plot from their mean square errors of testing sets. Figure 7 shows that models with knowledge transfer have lower average and lower variance. On the other hand, to test the improvements on model efficiency, we first use the training set and validating set to get the best validating errors on each level and use them as target lines. Next, we train the models with and without knowledge-transferring for 10 times to reach the same target validating errors on each level. We record the average number of epochs the models take to reach the target error. Lastly, we draw a

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited

Downloaded 06/06/23 to 130.215.207.14 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

Table 3: Impact of feature groups

	level 6	level 5	level 4	level 3	level 2	level 1	All levels
S	0.676	0.164	0.098	0.057	0.036	0.007	0.024
S+T	0.648	0.150	0.090	0.053	0.036	0.007	0.022
S+T+ST	0.625	0.118	0.066	0.049	0.034	0.007	0.021

box plot based on their level-average epoch cost. We can find that models with knowledge transfer mechanism have less average epoch cost and less variance. The experiments results show that knowledge transfer can further improve and stabilize prediction accuracy and training efficiency.

6 Conclusion

In this paper, we performed a comprehensive study on the traffic accident forecasting problem. Traffic accident prediction is important to transportation management and public safety, but it is very challenging due to spatial heterogeneity and rareness. We proposed a HintNet model to partition areas into multi-level subregions based on their accident risk and learn models for each level. Meanwhile, A knowledge transfer mechanism is applied across different levels. The experiments show that the HintNet is a promising solution to accident prediction problems, and HintNet outperforms the state-of-art method up to 12.5% on prediction error.

7 Acknowledgements

Bang An and Xun Zhou are partially supported by an ISSSF grant from the University of Iowa and the SAFER-SIM UTC under US-DOT award 69A3551747131. Yanhua Li was supported in part by NSF grants IIS-1942680 (CAREER), CNS1952085, CMMI-1831140, and DGE-2021871.

References

[1] Joaquín Abellán, Griselda López, and Juan De OñA. Analysis of traffic accident severity using decision rules via decision trees. *Expert Systems with Applications*, 40(15):6047–6054, 2013.

[2] Ruth Bergel-Hayat, Mohammed Debbarh, Constantinos Antoniou, and George Yannis. Explaining the road accident risk: Weather effects. *Accident Analysis & Prevention*, 60:456–465, 2013.

[3] Ciro Caliendo, Maurizio Guida, and Alessandra Parisi. A crash-prediction model for multilane roads. *Accident Analysis & Prevention*, 39(4):657–670, 2007.

[4] Li-Yen Chang. Analysis of freeway accident frequencies: negative binomial regression versus artificial neural network. *Safety science*, 43(8):541–557, 2005.

[5] Li-Yen Chang and Wen-Chieh Chen. Data mining of tree-based models to analyze freeway accident frequency. *Journal of safety research*, 2005.

[6] Quanjun Chen, Xuan Song, Harutoshi Yamada, and Ryosuke Shibasaki. Learning deep representation from big and heterogeneous data for traffic accident inference. In *Thirtieth AAAI conference on artificial intelligence*, 2016.

[7] Noel Cressie. *Statistics for Spatial Data*. John Wiley Sons, Incorporated, New York, 1993.

[8] Daniel Eisenberg. The mixed effects of precipitation on traffic crashes. *Accident analysis & prevention*, 36(4):637–647, 2004.

[9] Burd et al. Travel time to work in the united states: 2019. *American Community Survey Reports, United States Census Bureau*, 2021.

[10] Burrows et al. Commuting by public transportation in the united states: 2019. *American Community Survey Reports, United States Census Bureau*, 2021.

[11] Ester et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, page 226–231. AAAI Press, 1996.

[12] Huang et al. Deep dynamic fusion network for traffic accident forecasting. In *Proceedings of the 28th ACM ICKM*, CIKM '19, pages 2673–2681. ACM, 2019.

[13] Lin et al. A novel variable selection method based on frequent pattern tree for real-time traffic accident risk prediction. *Transportation Research Part C: Emerging Technologies*, 55:444–459, 2015.

[14] Shi et al. Convolutional lstm network: A machine learning approach for precipitation nowcasting. NIPS'15, Cambridge, MA, USA, 2015. MIT Press.

[15] Wang et al. Gsnet: Learning spatial-temporal correlations from geographical and semantic aspects for traffic accident risk forecasting. In *Proceedings of the AAAI*, volume 35, pages 4402–4409, 2021.

[16] Yuan et al. Hetero-convlstm: A deep learning approach to traffic accident prediction on heterogeneous spatio-temporal data. In *ACM SIGKDD*, 2018.

[17] Zhang et al. Trafficgan: Off-deployment traffic estimation with traffic generative adversarial networks. In *2019 IEEE International Conference on Data Mining (ICDM)*, pages 1474–1479, 2019.

[18] Zhou et al. Riskoracle: A minute-level citywide traffic accident forecasting framework. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

[19] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8), 1997.

[20] NHTSA Media. 2020 fatality data show increased traffic fatalities during pandemic, Jun 2021.

[21] Alameen Najjar, Shun'ichi Kaneko, and Yoshikazu Miyanaga. Combining satellite imagery and open data to map road safety. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

Copyright © 2022 by SIAM
Unauthorized reproduction of this article is prohibited