SpatialRank: Urban Event Ranking with NDCG Optimization on Spatiotemporal Data Supplementary Material

Bang An
Department of Business Analytics
University of Iowa
Iowa City, IA 52242
bang-an@uiowa.edu

Xun Zhou *
Department of Business Analytics
University of Iowa
Iowa City, IA 52242
xun-zhou@uiowa.edu

Yongjian Zhong
Department of Computer Science
University of Iowa
Iowa City, IA 52242
yongjian-zhong@uiowa.edu

Tianbao Yang
Department of Computer Science and Engineering
Texas A&M University
College Station, TX 77843
tianbao-yang@tamu.edu

1 Appendix A Feature Engineering

In this section, we explain how we generate features. Features are generated on partitioned grid cells and at different time intervals.
Temporal Features F_T Such calendar features and Weather features are generated from the date of Vehicle Crash or Crime Records, where all grid cells share a vector of temporal features in a time interval. calendar features include the day of the year, the month of the year, holidays, and so on. Weather features include temperature, precipitation, snowfall, wind speed, etc.
Spatial Features F_S are generated based on each grid cell and remain the same over different time intervals. First, POI features are the number of POI data in each grid cell for different categories. For example, one of the POI types is shopping, we count the number of shopping instances in each grid cell. Second, basic road condition features are extracted from road network data, in which we calculate the summation or average of provided data for road segments in each grid cell. Third, we use top eigenvectors of the Laplacian matrix of road networks as spatial graph features[12], which represent the topological information for each grid cell.
Spatio-Temporal Features F_{ST} such as real-time traffic conditions are estimated by taxi GPS data and Bus GPS data. Spatio-temporal features include pick-up volumes, drop-off volumes, traffic speed, etc.

Feature Summary In total, 36 features are extracted, including 12 temporal features, 18 spatial features, and 6 spatial-temporal features for each location s and time interval t.

*corresponding author

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

2 Appendix B Methodology

2.1 Symbol Table

Symbol Table	
Symbol	Explainations
$\mathcal{S}$	Spatial filed, study area
s	A partitioned location, grid cell
T	Temporal filed, study period
t	Time interval (e.g. hours, days)
F_T	Temporal features (weather, time)
F_S	Spatial features (e.g. POI)
F_ST	Spatiotemporal features (e.g. traffic conditions)
Z	Discounted Cumulative Gain (DCG) score
a	Pearson correlation coefficient
NDCG	Normalized Discounted Cumulative Gain
L-NDCG	Local Normalized Discounted Cumulative Gain
Prec	top-K precision
r()	Ranking function
$\mathcal{N}$	Neighbour querying

3 Appendix C Experiments

Parameter Configuration. For each method, we train the network for 100 epochs and save the model with the best performance on the validating set. We use the Adam optimizer [4] with settings $\alpha = 0.0001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$. We tune trade-off hyperparameter σ with value 0, 0.05, 0.1, 0.2, and 0.3. The R in the spatial neighbor definition is set as 2 grid cells for computation efficiency. We use an initial warm-up strategy by optimizing cross-entropy at the first 20 epochs to obtain a good initial solution because merely optimizing NDCG might land in the local minimum if a poor initial solution is given. The same warm-up process is also applied to optimization baselines. The learning rate is set at 0.001 in the warm-up and changed to 0.0001 afterward. The batch size is 64.

Data. For the state of Iowa, we collect data from 2016 to 2018. 80% of data from the year 2016 and year 2017 is randomly selected as a training set, and the remaining data is used as the validating set. The data from the year of 2018 is used as the testing set. The area is partitioned by 5 km $\times$ 5 km cells. The grid size is 128×64. Normalization is used to transform data into the range $[0, 1]$.

Platform. We run the experiments on a High-Performance Computer system where each node runs has an Intel Xeon E5 2.4 GHz and 256 GB of Memory. We use a GPU node with Nvidia Tesla V100 Accelerator Cards with the support of Pytorch library [10] to train the deep learning models.

Baselines. First, we compare our methods with daily **Historical Average (HA)**. Next, we consider popular machine-learning methods. **Long Short-term Memory (LSTM)** [3] is a recurrent neural network architecture with feedback connections, we stack two fully-connected LSTM layers. **Convolutional LSTM Network (ConvLSTM)** [8] is a recurrent neural network with convolution layers for spatial-temporal prediction. We use a stacked two-layer network. Thirdly, we compare with recent methods designed to tackle the traffic accident occurrence prediction problem. **GSNet** [11] is a deep-learning method utilizing complicated graph information. **Hetero-ConvLSTM** [12] is an advanced deep-learning framework to address spatial heterogeneity. It applies multiple ConvLSTM on pre-defined sub-regions with size 32×32. We use the same parameter setting in the experiments. **HintNet** [1] is a recent work capturing heterogeneous accident patterns by a hierarchical-structured learning framework. Finally, using our proposed network, we compared our optimization approach

TABLE 1: PERFORMANCE COMPARISON

IOWA	K=30			K=40			K=50		
	NDCG	PREC	L-NDCG	NDCG	PREC	L-NDCG	NDCG	PREC	L-NDCG
HA	.314±0	.167±0	.326±0	.326±0	.134±0	.261±0	.333±0	.111±0	.231±0
LSTM	.503±3‰*	.278±3‰*	.573±3‰*	.522±1‰*	.209±1‰*	.518±3‰*	.519±5‰*	.187±1‰*	.474±1‰*
CONVLSTM	.490±3‰*	.282±2‰*	.583±1‰*	.507±3‰*	.207±1‰*	.513±4‰*	.511±3‰*	.189±3‰*	.474±8‰*
GSNET	.493±2‰*	.265±1‰*	.569±3‰*	.509±3‰*	.222±3‰*	.527±3‰*	.526±5‰*	.207±1‰*	.510±3‰*
HETERO-CONVLSTM	.518±1‰*	.289±2‰*	.617±4‰	.523±5‰*	.258±1‰*	.589±5‰*	.543±3‰*	.226±1‰*	.534±5‰*
HINTNET	.512±5‰	.289±3‰*	.617±1‰	.542±5‰	.243±4‰*	.590±9‰	.556±3‰*	.209±2‰*	.534±8‰*
SPATIALRANK #	.530±3‰	.300±2‰	.617±4‰	.556±3‰	.264±2‰	.594±3‰	.571±3‰	.223±1‰	**.552**±4‰
SPATIALRANK	**.540**±3‰	**.309**±2‰	**.618**±3‰	**.563**±6‰	**.268**±3‰	**.600**±3‰	**.585**±3‰	**.232**±2‰	.550±6‰

* $\beta = 0.5$ ‰: $\times 10^{-3}$

TABLE 2: OPTIMIZATION COMPARISON

IOWA	K=30			K=40			K=@50		
	NDCG	PREC	L-NDCG	NDCG	PREC	L-NDCG	NDCG	PREC	L-NDCG
CE	.531±4‰	.306±4‰	.554±2‰	.555±1‰	.246±2‰	.556±2‰	.548±2‰	.205±1‰	.502±1‰
APPROXNDCG	.528±4‰	.304±3‰	.561±1‰	.551±3‰	.250±4‰	.557±6‰	.554±6‰	.212±2‰	.508±3‰
SONG	.529±3‰	.308±2‰	**.618**±1‰	**.563**±3‰	.264±4‰	.584±1‰	.581±3‰	.227±4‰	.536±6‰
SPATIALRANK	**.540**±3‰	**.309**±2‰	**.618**±3‰	**.563**±6‰	**.268**±3‰	**.600**±3‰	**.585**±3‰	**.232**±2‰	**.550**±6‰

‰: $\times 10^{-3}$

with other NDCG optimization solutions. **Cross Entropy (CE)** is a well-accepted objective function. **ApproxNDCG** [6] approximates the indicator function in the computation of ranks. **SONG** [7] is an efficient stochastic method to optimize NDCG.

3.1 Performance Comparison

We compare the results between SpatialRank and the baselines on the Iowa traffic accident dataset and observe that our full version of SpatialRank still outperforms other compared baselines in all three metrics. The results are shown in Table 1. SpatialRank$^{\#}$ with β pre-defined (as 0.5) rather than automatically learned is the second best. Consistent with the results on the other two datasets, SparkRank outperforms all the baselines and the dynamic convolution layers are effective in improving the model's performance.

3.2 Optimization Comparison

We perform the comparison of optimization methods on the Iowa traffic accident data. We use the same network architecture but with different optimization solutions including Cross Entropy (CE), ApproxNDCG, and SONG. The results are shown in Table 2. Methods designed to optimize NDCG consistently perform better than Cross Entropy. SpatialRank substantially outperforms SONG and ApproxNDCG and made a noticeable improvement on L-NDCG as it is considered in the objective function.

3.3 Computation Cost Comparison

We conduct comparisons with SOTA methods on average training time in seconds per epoch and inference time on the testing dataset in Table 3. The Chicago crime dataset and the Chicago accident dataset have the same input feature; thus, have equivalent training costs. In summary, SpatialRank trains faster than two SOTA baselines HintNet and GSNet on both datasets. It is only slower than HeteroConvLSTM but the training times of the two are on the same order of magnitude. The training phase of SpatialRank is slow because of computing nested L-NDCG loss function. Without extra cost on proposed optimization techniques, SpatialRank is significantly faster than all baselines in the inference phase. Given the improvement in prediction performance, we believe the cost of training time is acceptable, which will not affect the predicting efficiency of the proposed method.

TABLE 3: COMPUTATION COST

TRAINING TIME	COST IN SECONDS			
	SPATIALRANK	HINTNET	HETEROCONVLSTM	GSNET
CHICAGO	88.2	132.1	**47.7**	98.8
IOWA	76.5	117.5	**41.6**	83.5
INFERENCE TIME	SPATIALRANK	HINTNET	HETEROCONVLSTM	GSNET
CHICAGO	**5.6**	51.2	41.4	21.1
IOWA	**5.1**	41.7	12.3	16.2

TABLE 4: ABLATION STUDY ON WEIGHTING

CHICAGO ACCIDENT	K=30			K=40			K=50		
	NDCG	PREC	L-NDCG	NDCG	PREC	L-NDCG	NDCG	PREC	L-NDCG
NO-WEIGHT	0.255	0.441	**0.622**	0.265	0.417	0.613	0.274	0.401	0.595
SPATIALRANK	**0.257**	**0.444**	0.621	**0.268**	**0.420**	**0.614**	**0.278**	**0.403**	**0.599**
IOWA	NDCG	PREC	L-NDCG	NDCG	PREC	L-NDCG	NDCG	PREC	L-NDCG
NO-WEIGHT	0.531	0.304	0.617	0.557	0.264	0.591	0.573	0.225	0.546
SPATIALRANK	**0.540**	**0.309**	**0.618**	**0.563**	**0.268**	**0.600**	**0.585**	**0.232**	**0.550**
CHICAGO CRIME	NDCG	PREC	L-NDCG	NDCG	PREC	L-NDCG	NDCG	PREC	L-NDCG
NO-WEIGHT	0.364	0.484	0.660	0.379	0.466	0.642	0.390	**0.450**	**0.649**
SPATIALRANK	**0.373**	**0.491**	**0.665**	**0.380**	**0.467**	**0.647**	**0.392**	0.446	0.644

3.4 Ablation study

We perform an ablation study on the parameter σ on the Iowa traffic accident dataset. The results are shown in Table 5. For K=30, 40, and 50, σ = 0.05 always gives the best performance. For K=50 there is a tie in Precision@K between $\sigma = 0.05$ and $\sigma = 0.1$. Compared with the Chicago datasets, the best σ changed from 0.1 to 0.05, suggesting that local ranking might be less challenging in the Iowa data. This makes sense as the Iowa data has a coarser resolution (5km), making it easier to separate potential hotspots from surrounding grid cells.

3.5 Cross-K function

We use the Cross-K function[2] with Monte Carlo Simulation[9] to evaluate the accuracy of predicted locations. The Cross-K function measures the spatial correlation between the predicted accident locations and true locations in our case. Specifically, we calculate the average density of predictions within every distance d of a true event in each day as shown in Eq.1:

TABLE 5: ABLATION STUDY

IOWA	K=30			K=40			K=50		
	NDCG	PREC	L-NDCG	NDCG	PREC	L-NDCG	NDCG	PREC	L-NDCG
$\sigma = 0.0$	0.532	0.305	0.610	0.560	0.261	**0.597**	0.578	0.228	0.554
$\sigma = 0.05$	**0.539**	**0.311**	**0.626**	0.559	**0.267**	**0.597**	0.579	**0.231**	**0.560**
$\sigma = 0.1$	0.532	0.307	0.603	0.560	0.264	0.572	0.578	0.229	0.555
$\sigma = 0.2$	0.528	0.303	0.585	**0.561**	0.262	0.566	**0.580**	0.227	0.526
$\sigma = 0.3$	0.523	0.300	0.579	0.559	0.263	0.572	**0.580**	0.227	0.530

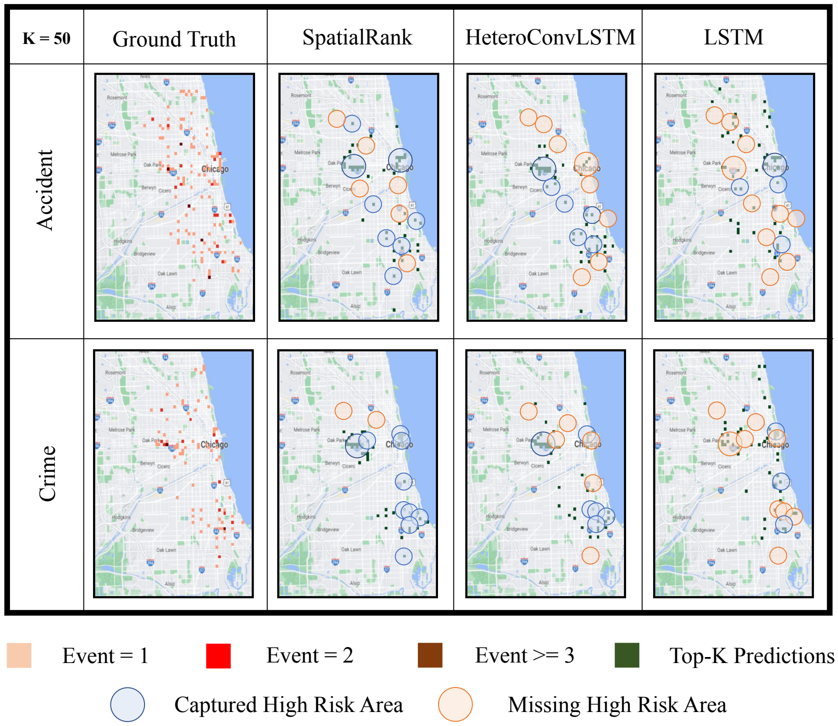

Figure 1: Case Study on Chicago Jan. 25^{th}, 2021.

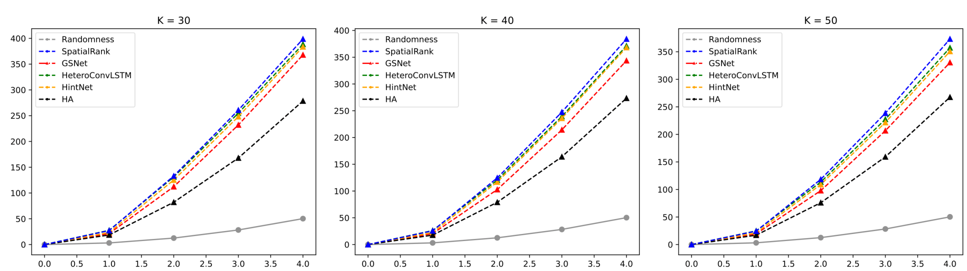

Figure 2: Comparison of Cross-K function for Chicago Crime.

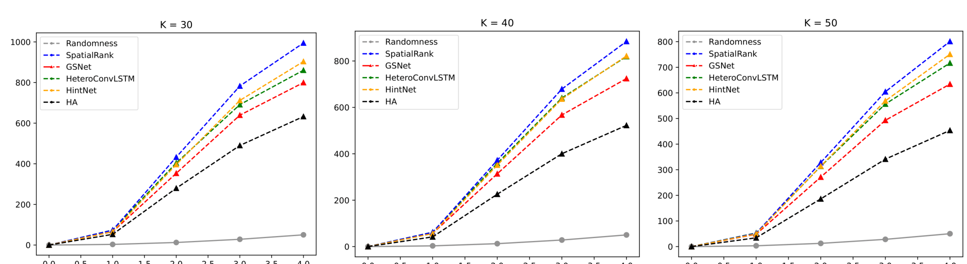

Figure 3: Comparison of Cross-K function for Iowa accident data.

$$\hat{K}(d) = \lambda_j^{-1} \sum_{i \neq j} I(d_{ij} \leq d)/n, \tag{1}$$

where λ is global density of event j, and $I()$ is an identity function which equals one if real distance d_{ij} is smaller than d, else equals zero. n is the number of events i. The results on the Chicago crime data and Iowa accident datasets are shown in Figure 2 and Figure 3, respectively. The grey curve represents the complete spatial randomness and we use it as a reference baseline. Higher curves are better. Our SpatialRank (blue) achieves the best predictions in both datasets, which indicates that the predictions of SpatialRank are significantly spatially correlated with ground truth.

3.6 Case Study

We present a successful prediction on Chicago using SpatialRank on Jan 25^{th} 2021 in Figure 1. There was a severe winter storm in the area of Chicago and 147 people were injured [5] and 99 severe crimes occurred. We compare predictions of SpatialRank with ground truth, Hetero-ConvLSTM, and LSTM. The top 50 riskiest locations in the predictions are chosen and labeled on the map. For easier visualization, the number of events greater than three in the ground truth map is changed to three, representing the riskiest location. To understand how those methods prioritize the riskiest locations, We define locations with events greater than 1 as high-risk areas. The blue circle indicates this high-risk location is predicted correctly. Missing High-risk location is indicated as an orange circle. We can observe that SpatialRank captures most High-risk locations, while others are not able to find the potential risk near the downtown area. As a result, the overall NDCG score is improved in SpatialRank.

References

[1] B. An, A. Vahedian, X. Zhou, W. N. Street, and Y. Li. Hintnet: Hierarchical knowledge transfer networks for traffic accident forecasting on heterogeneous spatio-temporal data. *Proceedings of the 2022 SIAM International Conference on Data Mining*, 2022.

[2] P. M. Dixon. *Ripley's K Function*. John Wiley & Sons, Ltd, 2014.

[3] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[4] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. 2014.

[5] NBC. Winter storm warning issued for chicago area ahead of forecasted snow storm. In *media*, 2021.

[6] T. Qin, T.-Y. Liu, and H. Li. A general approximation framework for direct optimization of information retrieval measures. *Information Retrieval*, 13(4):375–397, 2010.

[7] Z. Qiu, Q. Hu, Y. Zhong, L. Zhang, and T. Yang. Large-scale stochastic optimization of ndcg surrogates for deep learning with provable convergence. In *International Conference on Machine Learning*, 2022.

[8] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-k. Wong, and W.-c. Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. 2015.

[9] R. Tao and J.-C. Thill. Flow cross k-function: a bivariate flow analytical method. *International journal of geographical information science : IJGIS*, 33(10):2055–2071, 2019.

[10] Paszke et al. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019.

[11] B. Wang, Y. Lin, S. Guo, and H. Wan. Gsnet: Learning spatial-temporal correlations from geographical and semantic aspects for traffic accident risk forecasting. In *2021 AAAI Conference on Artificial Intelligence (AAAI'21)*, 2021.

[12] Z. Yuan, X. Zhou, and T. Yang. Hetero-convlstm: A deep learning approach to traffic accident prediction on heterogeneous spatio-temporal data. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 984–992, 2018.