Neural Network based Continuous Conditional Random Field for Fine-grained Crime Prediction

Fei Yi[1,4] , **Zhiwen Yu**[1] , **Fuzhen Zhuang**[2,3] and **Bin Guo**[1]
[1]Northwestern Polytechnical University, Xi'an, Shaanxi, China
[2]Key Lab of IIP of CAS, Institute of Computing Technology, CAS, Beijing, China
[3]University of Chinese Academy of Sciences, Beijing, China
[4]Baidu Inc, The Business Intelligence Lab, Baidu Research
yifeinwpu@gmail.com, zhiwenyu@nwpu.edu.cn, zhuangfuzhen@ict.ac.cn, guobin.keio@gmail.com

Abstract

Crime prediction has always been a crucial issue for public safety, and recent works have shown the effectiveness of taking spatial correlation, such as region similarity or interaction, for fine-grained crime modeling. In our work, we seek to reveal the relationship across regions for crime prediction using Continuous Conditional Random Field (CCRF). However, conventional CCRF would become impractical when facing a dense graph considering all relationship between regions. To deal with it, in this paper, we propose a Neural Network based CCRF (NN-CCRF) model that formulates CCRF into an end-to-end neural network framework, which could reduce the complexity in model training and improve the overall performance. We integrate CCRF with NN by introducing a Long Short-Term Memory (LSTM) component to learn the non-linear mapping from inputs to outputs of each region, and a modified Stacked Denoising AutoEncoder (SDAE) component for pairwise interactions modeling between regions. Experiments conducted on two different real-world datasets demonstrate the superiority of our proposed model over the state-of-the-art methods.

1 Introduction

Works on smart city applications related to mobile computing [Wang *et al.*, 2017; Liu *et al.*, 2019; Yu *et al.*, 2015], social economics [Yu *et al.*, 2016; Liu *et al.*, 2016; Fu *et al.*, 2014] and public safety [Yu *et al.*, 2018; Yi *et al.*, 2018] have inspired the implementation of advanced technologies in crime prevention [Wang *et al.*, 2013; Du *et al.*, 2016]. Specifically, the consideration of spatial correlation among different regions has been proved effective, where [Wang *et al.*, 2016] studied the taxi trajectory-based region relationship and [Zhao and Tang, 2017] modeled a distance-based region similarity for spatial-temporal crime prediction. However, the complexity of different type of spatial correlation between regions would eventually make the fine-grained crime prediction a difficult problem.

As discussed in [Qin *et al.*, 2009], CCRF is a powerful model that is typically designed to model effects of interactions among instances. And we could leverage this advantage in modeling region relationship for crime prediction by regarding each region in a city as an instance. However, traditional CCRF would face problems like complex gradient derivation and capacity for large-scale dataset when pairwise interactions across instances are all considered [Ristovski *et al.*, 2013]. To solve these problems, we take advantages from back-propagation algorithm in model training by introducing neural network components into CCRF model. Specifically, mean-field theory [Koller and Friedman, 2009] has proved that the inference process of CCRF model could be transferred into an iterative procedure, and we further reformulate it into neural network layers and propose a Neural Network based CCRF model (NN-CCRF) in our work.

Inspired by recent works [Zheng *et al.*, 2015; Xu *et al.*, 2017] on incorporating CRF with neural network for discrete labeling problems, we proceed to alleviate the limit of integrating CCRF with neural network for structured regression problems. In details, traditional CCRF model consists of two parts, the unary potential and the pairwise potential. Commonly, unary potential models relationship between inputs and outputs of each instance, and pairwise potential constraints outputs of each instance according to a predefined correlation matrix calculated by some kernel functions (e.g., Gaussian kernel, RBF kernel). Existing works only transform pairwise potential into neural network framework, and adopt predefined kernel functions to calculate the correlation matrix for pairwise interaction modeling. In our work, the proposed NN-CCRF model not only formulates pairwise potential into neural network, but also reformulates the unary potential into a Long Short-Term Memory (LSTM) neural network. Furthermore, we propose to learn the correlation matrix between instances using Stacked Denoising AutoEncoder (SDAE) rather than predefined kernel functions, which is more effective to understand the spatial correlation between regions in a data-driven manner. And our work mainly makes the following contributions:

- We propose a Neural Network based Continuous Conditional Random Field (NN-CCRF) model for fine-grained crime prediction, which applies Long Short-Term Memory (LSTM) as the unary potential and leverages Stacked Denoising AutoEncoder (SDAE) to learn spatial correlations across regions.

- We formulate the inference process of NN-CCRF model into a sequential neural network, which helps us to train the whole model in an end-to-end manner leveraging the advantages of back-propagation algorithm.
- We conduct experiments on two real-world crime records collected from Chicago and New York respectively. Considering different types of criminal incidences and disjointed grids in the city as fine-grained regions, our NN-CCRF model outperforms the state-of-the-art approaches with respect to crime prediction and ranking accuracy.

By achieving the above contributions, our work is of great importance for researchers to understand the mechanism of fusing/transforming traditional CCRF model into neural networks for crime prediction.

2 NN-CCRF Model for Crime Prediction

2.1 Problem Formulation

In our work, we propose to learn a neural network based nonlinear mapping model $M : \mathcal{I} \rightarrow \mathcal{O}$ from the input historical crime number $\mathcal{I}$ to the output future crime number $\mathcal{O}$. More formally, let $\mathcal{Q} = \{(\mathcal{H}_i, \mathcal{F}_i)\}_{i=1}^{q}$ be a training set of q pairs, where $\mathcal{H}_i \in \mathcal{I}$ represents the historical crime numbers and $\mathcal{F}_i \in \mathcal{O}$ donates the corresponding future crime numbers. To deal with fine-grained crime prediction, we divide city landscape into many small regions. Therefore, $\mathcal{H}_i$ is a $N \times T$ matrix and $\mathcal{F}_i$ forms as a $N \times 1$ vector, where N donates the number of disjointed regions in a city and T represents the length of historical time steps (e.g., T days, weeks, or months). That is, our model aims to predict future crime numbers leveraging historical T time steps of data records. Further, considering a $N \times N$ correlation matrix that can potential influence the crime distribution across all regions, our model is also required to capture spatial relationship for crime prediction.

2.2 Neural Network Based CCRF Model

The proposed Neural Network based Continuous Conditional Random Field (NN-CCRF) model is illustrated in Figure 1, which takes advantages from both CCRF model and NN algorithms. Specifically, we demonstrate a conventional CCRF model in the middle, where each gray node tagged with $\mathbf{x}_i$ represents the historical crime numbers with T time steps and white node in y_i donates the corresponding future crime numbers of i-th region. The *unary feature function* and *correlation matrix learning* components are proposed based on neural network algorithms to solve CCRF model, and we define our NN-CCRF model as:

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{\mathcal{Z}(\mathbf{x})}\exp\{\psi_u(\mathbf{y},\mathbf{x},\mathbf{h}^*) + \sum_{i,j}\psi_p(y_i,y_j,\mathbf{K}^*)\}, \tag{1}$$

where $\psi_u(\mathbf{y},\mathbf{x},\mathbf{h}^*)$ is the unary potential function, and we adopt LSTM [Hochreiter and Schmidhuber, 1997; Gers *et al.*, 1999] with hidden states $\mathbf{h}^*$ to represent the mapping from input $\mathbf{x}$ to output $\mathbf{y}$ as follows:

$$\begin{aligned} \psi_u(\mathbf{y},\mathbf{x},\mathbf{h}^*) =& -(\mathbf{y} - R(\mathbf{x},\mathbf{h}^*))^2, \\ R(\mathbf{x},\mathbf{h}^*) =& \sigma(W_h\mathbf{h}_t^* + b_h|\mathbf{x}, W_*, U_*, b_*), \end{aligned} \tag{2}$$

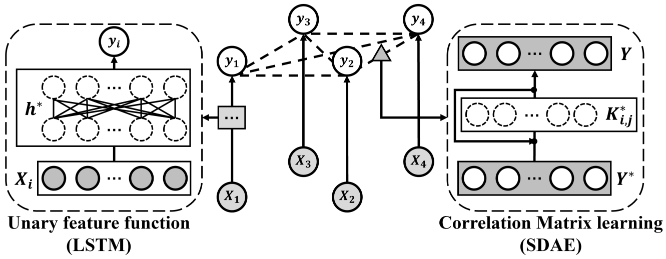


Figure 1: The proposed NN-CCRF model.

where $R(\mathbf{x},\mathbf{h}^*)$ is the preliminary estimations on $\mathbf{y}$ without considering the pairwise spatial correlations, $\sigma(\bullet)$ is the Sigmod function, and W_* and U_* are weight matrices, and b_* donate the bias vectors within LSTM components that are specified as follows in iteratively updating $\mathbf{h}_t^*$:

$$\begin{aligned} f_t =& \sigma(W_f\mathbf{x}_t + U_f\mathbf{h}_{t-1}^* + b_f), \\ i_t =& \sigma(W_i\mathbf{x}_t + U_i\mathbf{h}_{t-1}^* + b_i), \\ o_t =& \sigma(W_o\mathbf{x}_t + U_o\mathbf{h}_{t-1}^* + b_o), \\ g_t =& \tanh(W_g\mathbf{x}_t + U_g\mathbf{h}_{t-1}^* + b_g), \\ c_t =& f_t \odot c_{t-1} + i_t \odot g_t, \\ \mathbf{h}_t^* =& o_t \odot \tanh(c_t). \end{aligned} \tag{3}$$

Besides, the pairwise potential function provides a spatial-dependent smoothing term that encourage correlated regions to have similar crime numbers as defined:

$$\psi_p(y_i, y_j, \mathbf{K}^*) = -\mathbf{K}_{i,j}^*(y_i - y_j)^2, \tag{4}$$

where $\mathbf{K}_{i,j}^*$ represents the spatial-dependent correlation between y_i and y_j, which constraints and smooths the preliminary estimations (e.g., $R(\mathbf{x}_i,\mathbf{h}^*)$ and $R(\mathbf{x}_j,\mathbf{h}^*)$) to have better overall results. And one critical problem is how to deduce $\mathbf{K}_{i,j}^*$ for each pair of y_i and y_j. In our work, we leverage a modified SADE [Vincent *et al.*, 2010] framework to learn the spatial correlation matrix $\mathbf{K}^*$ used in pairwise potential function.

Specifically, SDAE is a feedforward neural network that matches the corrupted input and output (ground-truth) by encoding and decoding raw input data in a sequential manner. In our work, suppose we have corrupted input $\hat{\mathbf{y}}$ (inferred by unary potential function) and ground-truth $\mathbf{y}$, SDAE first encodes $\hat{\mathbf{y}}$ into image $\mathbf{z}$ and then decodes $\mathbf{z}$ to produce $\mathbf{y}'$ as predictions on $\mathbf{y}$ as follows:

$$\begin{aligned} \mathbf{z} =& \sigma(W_z\hat{\mathbf{y}} + b_z), \\ \mathbf{y}' =& \sigma(W_y\mathbf{z} + b_y), \end{aligned} \tag{5}$$

and the objective function of conventional SDAE is shown as:

$$\min_{\{W\},\{b\}} ||\mathbf{y} - \mathbf{y}'||_F^2 + \lambda||W||_F^2, \tag{6}$$

where W donate the weight matrices mapping $\hat{\mathbf{y}}$ to $\mathbf{y}$, which can be regarded that W achieves the goal of measuring correlation between y_i and y_j. And if we adopt each encoding or decoding layer with a same weight matrix, which indicates that our modified SDAE would have n identical layers, we will eventually learn the spatial correlation matrix $\mathbf{K}^*$ similarly by minimizing the following objective function:

$$\min_{\{\mathbf{K}^*\}} ||\mathbf{y} - \sigma_n(\mathbf{K}^* R(\mathbf{x}, \mathbf{h}^*))||_F^2 + \lambda ||\mathbf{K}^*||_F^2, \tag{7}$$

where $\sigma_n(\mathbf{K}^* R(\mathbf{x}, \mathbf{h}^*))$ is the n times encode-decode results of unary predictions in replace of $\mathbf{y}'$ according to n identical layers of modified SDAE, and $\mathbf{K}^*$ is exactly identical to W used in conventional SDAE, and we just replace it in each layer. Finally, our proposed NN-CCRF model manages to maximize the following function according to aforementioned LSTM and SDAE components:

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{\mathcal{Z}(\mathbf{x})} \exp\{ - (\mathbf{y} - \sigma(W_h \mathbf{h}_t^* + b_h | \mathbf{x}, W_*, U_*, b_*))^2 - \sum_{i,j} \mathbf{K}^*_{i,j} (y_i - y_j)^2 \}. \tag{8}$$

In general, solving such a probability density function that contains neural network component requires a lot mathematical derivation, even there is only a simple one layer neural network as discussed in [Baltrušaitis *et al.*, 2014], not to mention that in our work we will consider a LSTM model with many layers and a SDAE. And the main contribution of our work is to reformulate this function into a neural network framework and learn the parameters in an end-to-end manner, instead of manually deducing the gradients of every parameters, which reduces a lot gradient derivation as well as mathematical analysis.

2.3 End-to-End Model Inferencing

To avoid complicated gradient derivation as discussed in [Ristovski *et al.*, 2013; Qin *et al.*, 2009] for learning CRF models, CRFasRNN [Zheng *et al.*, 2015] is proposed to transfer part of CRF into neural network for fast learning in an end-to-end training manner. However, CRFasRNN is limited in only transforming pairwise potential in a RNN manner with a predefined kernel function for inferring pairwise correlation matrix. In our work, we not only reformulate unary and pairwise potential into neural networks, but also simultaneously learn the pairwise correlation matrix instead of adopting a predefined kernel function.

Directly solving the probability dense function in Equation (8) is impractical and complex. We turn to apply mean-field theory [Koller and Friedman, 2009] to approximate this distribution $P(\mathbf{y}|\mathbf{x})$. The objective of mean-field inference is to approximate distribution $P(\mathbf{y}|\mathbf{x})$ with distribution $Q(\mathbf{y}|\mathbf{x})$ that can be expressed as a product of independent marginals $Q(\mathbf{y}|\mathbf{x}) = \prod_{i=1}^{N} Q_i(y_i|\mathbf{x})$, where N represents the total number of regions in our work. To achieve this, we have to minimizes the Kullback-Leibler (KL) divergence between these two distributions P and Q:

$$\begin{aligned} KL(Q||P) = &\sum_{i,y_i} Q_i(y_i) \log Q_i(y_i) + \sum_{y_i} \psi_u(y_i|\mathbf{h}^*) \\ &\sum_{y_i,y_j} \psi_p(y_i, y_j|\mathbf{K}^*) Q_i(y_i) Q_j(y_j) \\ &+ \log \mathcal{Z}(\mathbf{h}^*, \mathbf{K}^*), \end{aligned} \tag{9}$$

Algorithm 1 Mean-field inference for CCRF

```
for $i$ in $N$ do
    $y_i^* \leftarrow R(\mathbf{x}_i|\mathbf{h}^*)$                              ▷ Unary estimation
end for
for $i$ in $Iter$ do
    $\hat{y_i^*} \leftarrow \sum_{i\neq j} \mathbf{K}^*_{i,j} y_j^*$              ▷ Pairwise interaction
    $y_i^* \leftarrow R(\mathbf{x}_i|\mathbf{h}^*) + \hat{y_i^*}$     ▷ Adding unary estimation
end for
```

where $\mathbf{h}^*$ and $\mathbf{K}^*$ donate the parameters in our model. Following derivations discussed in [Zheng *et al.*, 2015; Krähenbühl and Koltun, 2011], we are able to solve above function and obtain the following compact update functions for model inference:

$$y_i^* = R(\mathbf{x}_i|\mathbf{h}^*) + \sum_{j\neq i} \mathbf{K}^*_{i,j} y_j^*. \tag{10}$$

where y_i^* is the estimated crime number for i-th region. And the goal of mean-field approximation for regression problem is to iteratively update each y_i^* according to Equation (10) to minimize the mean absolute error between estimated y_i^* and ground truth y_i. Specifically, the mean-field approximation inference algorithm is illustrated in Algorithm 1.

There are two main stages during the whole mean-field inference according to Algorithm 1. To begin with, the algorithm first gives estimations on each region using unary feature function. After that, the algorithm passes the estimated values across all regions considering their pairwise interactions. The final estimation of each region is the combination of unary estimation and pairwise interaction. To learn parameters in unary feature function and the corresponding correlation matrix $\mathbf{K}^*$, we transform the mean-field inference into a sequential neural network framework as illustrated in Figure 2, in which there are two distinct neural networks include a LSTM block for unary potential and SDAE block for learning correlation matrix used in pairwise potential.

In specific, the LSTM block is used in replacement of unary potential, which aims to learn relationship between input features X and output Y and we consider three layers in LSTM in our work. Besides, since there are totally N regions, the model has to apply a $N \times N$ pairwise correlation matrix to constraint the values on each y_i and y_j, and we leverage SDAE to learn such $N \times N$ matrix which is used in pairwise

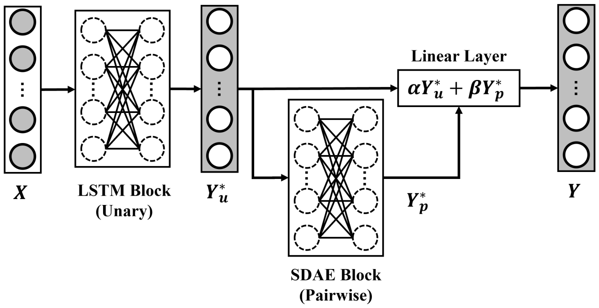


Figure 2: Neural network based mean-field inference

potential. Particularly, we notice that the correlation $K^*_{i,j}$ remains invariant when iteratively updating Y^*_u, therefore, in our work, each layer in SDAE should be the identically same and forms as a $N \times N$ matrix to keep the consistency to Algorithm 1. Further, the number of SDAE layers should be identical to the number of iterations in stage two of mean-field inference (the pairwise interaction phase). Hence, by changing the number of SDAE layers, we could control the iteration numbers for mean-field inference thus influencing the degree of applying the pairwise interaction in our model. Besides, in order to balance the importance of unary and pairwise estimations, we add a linear combination in the end of the procedure. Since we apply neural networks in mean-field inference for CCRF, we are able to learn the model's parameters in an end-to-end manner.

Algorithm 2 gives a detailed procedure of the end-to-end model training for our NN-CCRF according to the proposed neural network based mean-field inference. Specifically, the hyper-parameter η, λ, N, and $epoch$ represent the number of hidden states in LSTM block, the number of layers in SDAE block, the number of regions, and the number of iterations for training respectively. The whole procedure could be divided into two steps, line 3 to line 11 contain the procedure of initialing a NN-CCRF model according to neural network based mean-field inference, and line 12 to line 19 are the end-to-end training procedure of NN-CCRF model. Without calculating any parameters' gradients and considering mathematical constraints, this end-to-end parameter learning algorithm is simple but efficient. We implement this algorithm using a popular deep learning toolkit Pytorch [Chollet, 2017].

3 Experiments

3.1 Data Preprocessing

We totally collect 1,072,208 crime records in Chicago city from Jan. 1 2013 to Dec. 31 2015, and 1,417,083 in New York City from Jan.1 2015 to Dec. 31 2016. Specifically, each crime record is attached with timestamp, location, and crime type information. In order to predict a fine-grained crime number across the city, we first divide the whole city landscape into N disjointed regions, and each region is a 1km×1km grid resulting in 35 (7×5) and 63 (7×9) grids for Chicago and New York respectively. Then, we select one day as the minimum time step, and aggregate the crime incidence into different regions in different time steps. Knowing that it is also acceptable to select different length (e.g., 500m, 200m) for regions and different time step (e.g., week, month).

Noticing that data sparsity could become an issue in our work for predicting specific crime types, we manage to aggregate similar crime types into classes (crime against person or property) to solve such data sparsity problem. For example, in Chicago dataset, the ratio of non-zeros counts is 0.07, 0.16, 0.04, 0.4 and 0.05 for crime types assault, battery, robbery, theft, and burglary respectively, while that becomes 0.24 and 0.43 for crime against person and property after we aggregate them.

Algorithm 2 End-to-End model training for NN-CCRF

```
 1: Input: hyper-parameters $\eta$, $\lambda$, $N$, $epoch$, and training set
    $\mathcal{Q} = \{(\mathcal{H}_i, \mathcal{F}_i)\}_{i=1}^q$
 2: Output: NN-CCRF model $\mathcal{M}$
 3: $\mathcal{M} \leftarrow nn.Sequential()$            ▷ create a NN-CCRF
 4: $lstm \leftarrow nn.LSTM(\eta)$                     ▷ create a LSTM block
 5: $sdae \leftarrow nn.Sequential()$                   ▷ create a SDAE block
 6: $kernel \leftarrow nn.Linear(N, N)$                 ▷ create a SDAE layer
 7: for $i$ in $\lambda$ do
 8:     $sdae.append(kernel)$                          ▷ fill in SDAE block
 9: end for
10: $linear \leftarrow nn.Linear(2, 1)$                 ▷ create a linear layer
11: $\mathcal{M}.append([lstm, sdae, linear])$          ▷ fill in NN-CCRF
12: for $i$ in $epoch$ do
13:     $\mathcal{F}^* \leftarrow \mathcal{M}.forward(\mathcal{H})$
14:     $loss \leftarrow MSELoss(\mathcal{F}^*, \mathcal{F})$
15:     if $loss$ is minimized then
16:         break
17:     end if
18:     $loss.backward()$
19: end for
20: return $\mathcal{M}$
```

3.2 Experiment Setup

In our work, we do not only evaluate the prediction precision, but also try to measure the ranking performance. Hence, we apply the following two metrics. The averaged Root Mean Square Error (RMSE):

$$RMSE = \sqrt{\frac{1}{t \times N} \sum_{j=1}^{t} \sum_{i=1}^{N} (y^*_{i,j} - y_{i,j})^2}, \tag{11}$$

and the averaged Hitting Rate@K (HR@K):

$$HR@K = \frac{1}{t \times K} \sum_{j=1}^{t} \sum_{i=1}^{K} I(list^*_{i,j}, list_{i,j}), \tag{12}$$

where N is the number of regions, t is the number of testing days, such that $t = 5$ represents we totally conduct experiments on distinct 5 days. And $list^*$ and $list$ are the ordered region sequence according to Y^* and Y respectively, $I(\cdot)$ donates a binary function which outputs 1 when the inputs are identical otherwise 0. Specifically, smaller RMSE represents better performance on prediction precision, and higher HR@K values indicate better performance the model could rank all regions in the same order identical to the real ranking list. We list the following methods for comparison in our work in details.

- **History**: This method is the baseline method in our work, which simply regards crime numbers in time t_{k-1} as the predicted future crimes in time t_k.
- **LR**[Montgomery *et al.*, 2012]: Linear Regression model is applied to the whole dataset merged from all regions, and it assumes that future crime numbers are linearly related to historical crime numbers.

Model	**Datasets**									
	Crime against Person					**Crime against Property**				
	CHI_{13}	CHI_{14}	CHI_{15}	NY_{15}	NY_{16}	CHI_{13}	CHI_{14}	CHI_{15}	NY_{15}	NY_{16}
History	0.7505	1.1139	0.8713	1.6410	1.6466	1.4541	1.5105	1.3171	1.7294	1.7090
LR	0.6159	0.8869	0.6647	1.4497	1.4071	1.6034	1.6354	1.4157	2.0137	2.0616
LSTM	0.6298	0.8932	0.6365	1.2720	1.3338	1.3969	1.4409	1.2165	1.5148	1.4433
CRFasRNN	0.6098	0.8092	0.6841	1.3473	1.3263	1.1478	1.1085	1.0945	1.4680	1.4075
TCP	0.5970	0.8429	0.6511	1.3642	1.3584	1.3718	1.1906	1.1051	2.3182	1.7097
NN-CCRF	**0.5838**	**0.7581**	**0.6272**	**1.2632**	**1.2805**	**1.0278**	**1.0602**	**1.0751**	**1.3077**	**1.2774**

Table 1: Prediction performance comparison (RMSE).

- **LSTM**[Gers *et al.*, 1999]: Long Short-Term Memory is one type of Recurrent Neural Network (RNN) that is widely used in time-series modeling and prediction. It introduces hidden states to capture the dynamic of time-series and usually provide reasonable results. In our work, we build a three layers LSTM with predefined hidden state η with H dimensions.

- **CRFasRNN**[Zheng *et al.*, 2015]: This model is proposed for classification tasks, and we reformulate its object function to suit for regression problem. CRFasRNN only transform pairwise updating into RNN framework, and still adopts predefined kernel functions to calculate the correlation matrix between regions. Here we adopt two kernels $\exp(-\frac{||p_i-p_j||^2}{2\theta_\alpha^2})$ and $\exp(-\frac{||f_i-f_j||^2}{\theta_\beta^2})$, where p_i and p_j are the location of i-th and j-th region, f_i and f_j donate the corresponding Point-Of-Interest (POI) feature vector of these regions.

- **TCP**[Zhao and Tang, 2017]: Temporal-spatial Correlation for Crime Prediction is a multi-task learning framework that considers both parametric temporal and spatial constraints in crime prediction. Its spatial correlation is predefined using power law exponential function that $\mathbf{K}_{i,j} = d_{i,j}^{-\phi}$, where $d_{i,j}$ is the distance between i-th and j-th region, ϕ is a predefined regularization parameter controlling the degree of spatial correlation.

- **NN-CCRF**: Our proposed Neural Network based Continuous Conditional Random Field model regards traditional CCRF model as sequential neural network, which applies LSTM and SDAE in unary feature function and pairwise interaction learning. Using mean-field inference, the model is trained in an end-to-end manner. And the correlation matrix $\mathbf{K}^*$ is simultaneously learned from SDAE instead of predefined using some kernel functions.

3.3 Prediction Performance

We test prediction accuracy on different subsets from our collected data, including $CHI_{\{13-15\}}$ and $NY_{\{15,16\}}$ representing dataset from different years of Chicago (e.g., CHI_{13} is the dataset from year 2013 of Chicago) and New York respectively. The detailed experimental results from our proposed NN-CCRF model and other models are demonstrated in Table 1.

Before further analysis on model performances, we observe that as a simple inference model, History based method outperforms LR with respect to crime against property, which indicates that only using yesterday's crime number would be more effective than combining last few days crime numbers for property criminal prediction. While for predicting crime against person, LR outperforms History and History performs the worst as expected. Among all compared approaches, our proposed NN-CCRF model performs the best on all datasets and crime types. Specifically, LR and LSTM models mainly focus on temporal correlations alone for crime prediction, while CRFasRNN, TCP and NN-CCRF additionally take spatial correlation across regions into consideration.

Comparing to LR and LSTM that only take temporal factor into account, CRFasRNN and TCP are more likely to achieve better performance due to the consideration of spatial correlation between regions. However, there are still negative effects, such as results in CHI_{15} and $NY_{15,16}$ for crime against person or property where they perform worse than LR or LSTM, one possible reason is that the spatial correlation considered in these two methods are all predefined, which may not always be suitable for different situations. Different from CRFasRNN and TCP, our proposed NN-CCRF model applies SDAE to learn the pairwise spatial correlation in a data-driven manner, which is more likely to capture the dynamics of spatial correlation under various situations. Hence providing better and more robust results.

We further analysis the running time across different models as shown in Figure 3. Our proposed NN-CCRF model costs reasonable running time comparing to other models. Specifically, our model costs equally comparing with LSTM and a little longer than TCP and CRFasRNN models. Both neural network based models, including NN-CCRF and LSTM, grow linearly but faster than other two models when increasing the training data size.

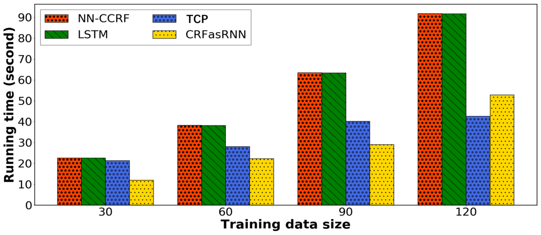
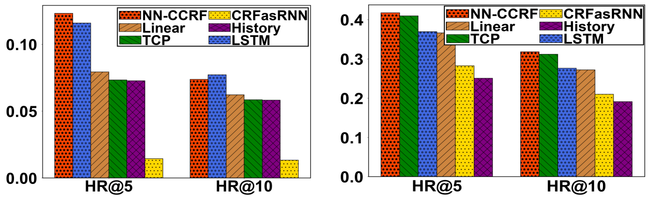
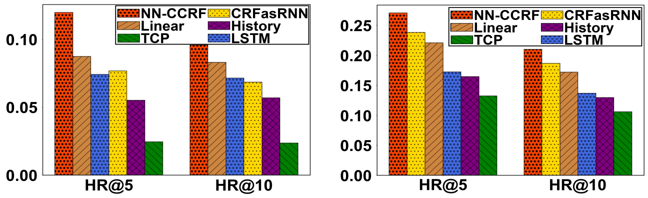


Figure 3: Running time on different models.



(a) person (b) property

Figure 4: Ranking performances on New York (HR@5,10).



(a) person (b) property

Figure 5: Ranking performances on Chicago (HR@5,10).

3.4 Ranking Performance

Apart from evaluating the prediction accuracy, we also conduct experiments on each subset to judge if the model could rank the most "dangerous" regions as demonstrated in Figure 4 and Figure 5. Specifically, each figure contains the ranking performance for two different crime types (person and property), and a higher value indicates better performance. According to the results, we observe that the performance of comparison models (e.g., LSTM, TCP, or CRFasRNN) fluctuates a lot under different situations, for example, LSTM performs the fourth place when evaluating on crime against person from New York dataset as shown in Figure 4(a), while it performs the second place as shown in Figure 4(b) and becomes the worst when testing on Chicago dataset as illustrated in Figure 5. Similarly, spatial-based models with predefined kernel functions like TCP and CRFasRNN also perform differently under different situations, which indicating that a static spatial correlation is insufficient for various conditions. However, our data-driven based NN-CCRF model could learn dynamic spatial correlations through SDAE component under different situations, achieving more robust performance in all conditions. Noticing that TCP performs better than ours when testing on crime against person from New York dataset with Hitting Rate at 10 as shown in Figure 4(a).

Dataset	λ η	4	6	8	10
CHI	32	**0.855**	0.928	0.919	0.907
	64	0.879	0.874	0.880	0.876
	128	0.875	0.874	0.873	0.874
NY	32	1.291	1.285	1.283	1.295
	64	1.236	1.236	1.236	1.236
	128	1.237	1.237	1.237	1.236

Table 2: Influence of different parameter settings on prediction.

Dataset	λ η	4	6	8	10
CHI	32	0.153	0.157	0.139	0.148
	64	0.156	0.156	0.157	0.153
	128	0.161	0.152	**0.169**	0.160
NY	32	0.207	0.212	0.224	0.208
	64	0.228	0.230	0.230	0.230
	128	0.229	0.230	0.230	0.230

Table 3: Influence of different parameter settings on ranking.

3.5 Hyper-parameters Effects

We test two major hyper-parameters (η for LSTM and λ for SDAE) as illustrated in Table 2 and 3, and the results are averaged from two crime types. We discover that the parameter setting for best prediction and ranking performance are not consistent. For example, our model achieves its best performance when $\eta = 32$ and $\lambda = 4$ for prediction with Chicago dataset, while that is $\eta = 128$ and $\lambda = 8$ for its best ranking performance. Besides, we discover that λ impacts more on model's performance when η is small, showing that spatial correlation plays an important role when temporal influence is weak, while its influence vanishes when η grows larger. In our work, considering the trade-off between accuracy and computational complexity, we take $\eta = 32$ and $\lambda = 4$ for training model using Chicago dataset, and that is $\eta = 64$ and $\lambda = 4$ for New York dataset.

4 Conclusion

In this work, we proposed to exploit the effectiveness of formulating conventional CCRF model into neural network framework for spatial correlation learning in fine-grained crime prediction. Specifically, we first reformulated the unary potential of CCRF into LSTM, and applied SDAE to learn pairwise interaction between instances. After that, we transformed the inference of CCRF model into an iterative process based on mean-field approximation theory. By achieving these, a Neural Network based CCRF (NN-CCRF) model was proposed, which is able to deal with large-scale and fine-grained crime prediction. Specifically, the proposed SDAE component made it much more effective and convenient to capture spatial correlation between disjointed regions, comparing to traditional predefined kernel functions for calculating the correlation matrix. Furthermore, we reformulated the inference process of NN-CCRF model into a sequential neural network, which could help us to train NN-CCRF in an end-to-end manner using back-propagation algorithm. Experiments conducted on two real-world datasets validated the superiority of our model compared to several state-of-the-art approaches with respect to prediction and ranking accuracy.

Acknowledgments

This work was supported in part by the National Science Fund for Distinguished Young Scholars (No. 61725205), the National Key R&D Program of China (No. 2017YFB1002000), and the National Natural Science Foundation of China (No. 61772428, 61773361, U1836206).

References

[Baltrušaitis *et al.*, 2014] Tadas Baltrušaitis, Peter Robinson, and Louis-Philippe Morency. Continuous conditional neural fields for structured regression. In *European conference on computer vision*, pages 593–608. Springer, 2014.

[Chollet, 2017] Francois Chollet. *Deep learning with python*. Manning Publications Co., 2017.

[Du *et al.*, 2016] Bowen Du, Chuanren Liu, Wenjun Zhou, Zhenshan Hou, and Hui Xiong. Catch me if you can: Detecting pickpocket suspects from large-scale transit records. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 87–96. ACM, 2016.

[Fu *et al.*, 2014] Yanjie Fu, Yong Ge, Yu Zheng, Zijun Yao, Yanchi Liu, Hui Xiong, and Jing Yuan. Sparse real estate ranking with online user reviews and offline moving behaviors. In *2014 IEEE International Conference on Data Mining*, pages 120–129. IEEE, 2014.

[Gers *et al.*, 1999] Felix A Gers, Jürgen Schmidhuber, and Fred Cummins. Learning to forget: Continual prediction with lstm. 1999.

[Hochreiter and Schmidhuber, 1997] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[Koller and Friedman, 2009] Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[Krähenbühl and Koltun, 2011] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *Advances in neural information processing systems*, pages 109–117, 2011.

[Liu *et al.*, 2016] Junming Liu, Leilei Sun, Weiwei Chen, and Hui Xiong. Rebalancing bike sharing systems: A multi-source data smart optimization. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1005–1014. ACM, 2016.

[Liu *et al.*, 2019] Hao Liu, Ting Li, Renjun Hu, Yanjie Fu, Jingjing Gu, and Hui Xiong. Joint representation learning for multi-modal transportation recommendation. *AAAI, to appear*, 2019.

[Montgomery *et al.*, 2012] Douglas C Montgomery, Elizabeth A Peck, and G Geoffrey Vining. *Introduction to linear regression analysis*, volume 821. John Wiley & Sons, 2012.

[Qin *et al.*, 2009] Tao Qin, Tie-Yan Liu, Xu-Dong Zhang, De-Sheng Wang, and Hang Li. Global ranking using continuous conditional random fields. In *Advances in neural information processing systems*, pages 1281–1288, 2009.

[Ristovski *et al.*, 2013] Kosta Ristovski, Vladan Radosavljevic, Slobodan Vucetic, and Zoran Obradovic. Continuous conditional random fields for efficient regression in large fully connected graphs. In *AAAI*, pages 840–846, 2013.

[Vincent *et al.*, 2010] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of machine learning research*, 11(Dec):3371–3408, 2010.

[Wang *et al.*, 2013] Tong Wang, Cynthia Rudin, Daniel Wagner, and Rich Sevieri. Learning to detect patterns of crime. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 515–530. Springer, 2013.

[Wang *et al.*, 2016] Hongjian Wang, Daniel Kifer, Corina Graif, and Zhenhui Li. Crime rate inference with big data. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 635–644. ACM, 2016.

[Wang *et al.*, 2017] Liang Wang, Zhiwen Yu, Bin Guo, Tao Ku, and Fei Yi. Moving destination prediction using sparse dataset: A mobility gradient descent approach. *ACM Transactions on Knowledge Discovery from Data*, 11(3):37, 2017.

[Xu *et al.*, 2017] Dan Xu, Elisa Ricci, Wanli Ouyang, Xiaogang Wang, and Nicu Sebe. Multi-scale continuous crfs as sequential deep networks for monocular depth estimation. In *Proceedings of CVPR*, volume 1, 2017.

[Yi *et al.*, 2018] Fei Yi, Zhiwen Yu, Fuzhen Zhuang, Xiao Zhang, and Hui Xiong. An integrated model for crime prediction using temporal and spatial factors. In *2018 IEEE International Conference on Data Mining*, pages 1386–1391. IEEE, 2018.

[Yu *et al.*, 2015] Zhiwen Yu, Huang Xu, Zhe Yang, and Bin Guo. Personalized travel package with multi-point-of-interest recommendation based on crowdsourced user footprints. *IEEE Transactions on Human-Machine Systems*, 46(1):151–158, 2015.

[Yu *et al.*, 2016] Zhiwen Yu, Miao Tian, Zhu Wang, Bin Guo, and Tao Mei. Shop-type recommendation leveraging the data from social media and location-based services. *ACM Transactions on Knowledge Discovery from Data*, 11(1):1, 2016.

[Yu *et al.*, 2018] Zhiwen Yu, Fei Yi, Qin Lv, and Bin Guo. Identifying on-site users for social events: Mobility, content, and social relationship. *IEEE Transactions on Mobile Computing*, 17(9):2055–2068, 2018.

[Zhao and Tang, 2017] Xiangyu Zhao and Jiliang Tang. Modeling temporal-spatial correlations for crime prediction. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 497–506. ACM, 2017.

[Zheng *et al.*, 2015] Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Su, Dalong Du, Chang Huang, and Philip HS Torr. Conditional random fields as recurrent neural networks. In *Proceedings of the IEEE international conference on computer vision*, pages 1529–1537, 2015.