Simulating multiple land use changes by incorporating deep belief network into cellular automata: a case study in BEIJING-TIANJIN-HEBEI region, China

Cong Ou
China Agricultural University
No.17 Tsing Hua East Road, Haidian District Beijing, P.R.China
oucong@cau.edu.cn

Jianyu Yang
China Agricultural University
No.17 Tsing Hua East Road, Haidian District Beijing, P.R.China
ycjyyang@cau.edu.cn

Zhenrong Du
China Agricultural University
No.17 Tsing Hua East Road, Haidian District Beijing, P.R.China
Iris_dudu@126.com

Pengshan Li
Chengdu Land Planning and Cadastre Center
No.3 Jiayuan Road,Qingyang District, Chengdu, P.R.China
targetlps@gmail.com

Dehai Zhu
China Agricultural University
No.17 Tsing Hua East Road, Haidian District Beijing, P.R.China
zhudehai@263.net

Abstract

Cellular automata (CA) has been widely used to simulate complex geographical processes. The core of a CA model is defining appropriate transition rules to constructing nonlinear relationship between spatial variables and geographical structure. This paper presents an intelligent approach to discover transition rules for CA model by using deep belief network (DBN). DBN is a novel neural network model which is composed of some stacks of restricted boltzmann machines. The nonlinear features and structures of land use structure from training data which determines transition rules in CA model is extracted by a layer-by-layer unsupervised pre-training and a supervised back-propagation fine-tuning procedure in DBN. A case study simulating the multiple land use changes in the BEIJING-TIANJIN-HEBEI region demonstrates that the proposed model can achieve high accuracy and overcome some limitations of existing CA models.
Keywords: deep belief network; transition rules; CA; simulation; land use changes;

1 Introduction

Cellular automata (CA) are examples of mathematical systems constructed from many identical components, each simple, but together capable of complex behaviour (S Wolfram, 1984). CA adopt a bottom-up approach, through which local individual behaviors can give rise to complex global patterns (Liu et al., 2008). Because of the simplicity, flexibility and intuitiveness of CA (Santé et al. 2010), a series of CA-based models were proposed (Batty, 1997; Clarke et al., 1997; Torrens and O'Sullivan, 2001; Verburg et al., 2004; Almeida et al., 2008). These studies have demonstrated the advantages of CA-based models for simulating complex geographical processes.

The core of CA is how to define the transition rules, which determine the state conversion of geographical processes (Liu et al., 2008a). With many spatial variables and parameters involved, it's difficult to obtain appropriate transition rules. Varies of methods were proposed to describe the transition rules of CA for different research areas or objects. Traditional methods, like multi-criteria evaluation (Wu and Webster, 1998) and SLEUTH model (Silva et al, 2002), are simple and their mechanism are clear. However, with a large set of variables, it is not efficient and reliable because the determination of parameters has certain subjectivity and randomness. And a major drawback of equation-based models is that they have difficulties in tackling a series of complex behaviors associated with natural systems. To overcome this problem, a series of machine-learning methods were proposed, such as logistic regression (Wu, 2002), artificial neural networks (Li and Yeh, 2002), support vector machines (Yang et al, 2008). Furthermore, artificial intelligence algorithms were put forward for explicitly obtain the transition rules of CA, like data mining approaches (Li and Yeh, 2004), genetic algorithm (Jenerette and Wu, 2001), ant colony optimization (Yang et al, 2012), etc. The methods above show a significant improvement in obtaining nonlinear transition rules for CA, but there still remains many problems like over-fitting, resulting in local optimization or difficult to interpret the inference process. Moreover, the geographical simulation of large-scale regions with fine resolution units has become an inevitable trend (He et al, 2013), which further makes the implementation of fast CA simulations difficult in traditional methods.

Recently, deep learning has become the dominant technique to learning more information from multivariable nonlinear systems. A deep architecture consists of feature detector units arranged in multiple layers: lower layers detect simple features and feed into higher layers, which in turn detect more complex features (Ji et al, 2014). DBN is a multilayer generative neural network along with a greedy layer-wise learning algorithm, and it has been successfully implemented in dimensionality reduction (Hinton, 2006), time series forecasting (Chao, 2011) and digit recognition (Bengio, 2007).

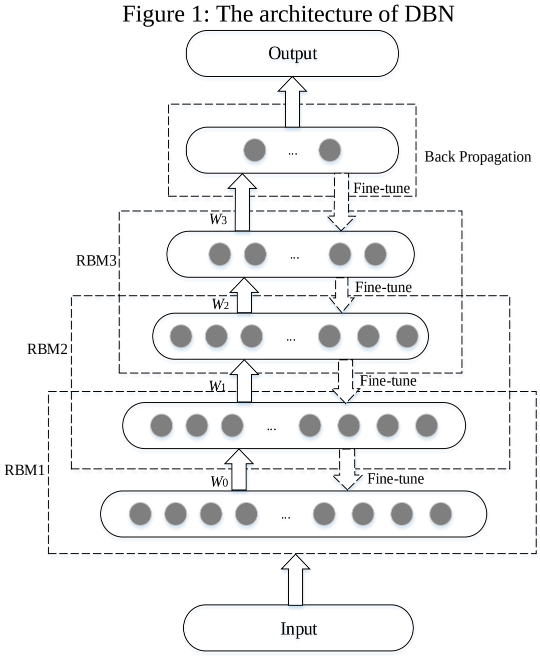
In this paper, we will explore the practicability and efficient of implementing deep belief network (DBN) to discover transitions rules of CA. And this proposed model was tested using regional land use changes data from BEIJING-TIANJIN-HEBEI (BTH) region, China, for the period 2005-2015.

2 Methodology

2.1 Deep belief network

Deep belief network (DBN) is probabilistic generative model and a feedforward neural network. A typical DBN architecture contain an unsupervised learning subpart by using restricted Boltzmann machines (RBMs) which is trained in a greedy manner and followed with a supervised fine-tuning subpart like back-propagation in the top-level for prediction (see Figure 1).

Figure 1: The architecture of DBN



As the basic component of DBN, restricted boltzmann machine (RBM) is a stochastic neural network and an energy based model in essence. In an RBM, there are only 2 layers so-called input layer (or visible layer) of i dimension representing observable data and output layer (or hidden layer) of j dimension representing detected features from observable data. The connection of RBM's units is restricted to different layers and there are no connections within a layer (see Figure 2).

The weight matrix W_{ij}(size: $m \times n$) encode a statistical relationship by the conditional distribution $p(h|v)$ and $p(v|h)$ between the visible and the hidden layer, which can be mathematically described as equations (1) and (2):

$$p(h|v) = \frac{p(h,v)}{p(v)} \quad (1)$$

$$p(v|h) = \frac{p(v,h)}{p(h)} \quad (2)$$

where v is the visible vector and h is the hidden vector. And the joint distribution $p(h,v)$ and $p(v,h)$ is calculated by energy function as equation (3):

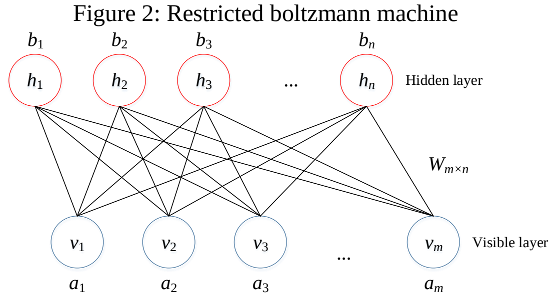
$$p(h,v) = p(v,h) = \frac{e^{-E(v,h)}}{\sum_{v,h} e^{-E(v,h)}} \quad (3)$$

The energy function $E(v,h)$ of certain configuration can be defined as equation (4):

$$E(v,h) - \sum_{i=1}^{n_v} a_i v_i \sum_{j=1}^{n_h} b_j h_j \sum_{i=1}^{n_v} \sum_{j=1}^{n_h} h_i W_{j,i} v_i \quad (4)$$

where a_i, b_j are the offsets of the visible and hidden layer respectively).

Figure 2: Restricted boltzmann machine



The training process of DBN which can be divided into two main stages is as follow (Qin et al, 2017):

- **Unsupervised pre-train stage**

Step 1: Initialize weights W with normal distribution. For each input data $x_t, t \in [1, z], v = x_t$.
Step 2: Compute the probability of hidden units $p(h|v)$.
Step 3: Compute the probability of reconstructed visible units $p(v|h)$.
Step 4: Obtain the reconstruction error ΔW.
Step 5: Update the weights W, and calculate the energy function $E(v,h)$.
Step 6: Repeat *Step 2*~*Step 5* until the energy function $E(v,h)$ decreases to be a convergent state.

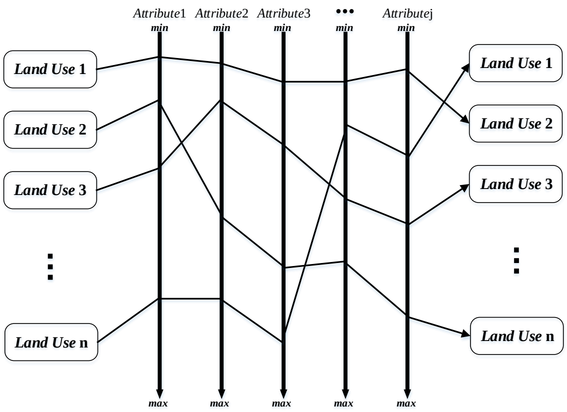
- **Supervised fine-tuning stage**

After an unsupervised pre-train stage, all the parameters are required to be slightly adjusted in supervised manner until the loss function of DBN reaches its minimum (Wang et al, 2016). In this paper, back-propagation (BP) is periodically works in the top-level RBM during the supervised fine-tuning stage.

2.2 DBN-based CA model for simulating multiple land use changes

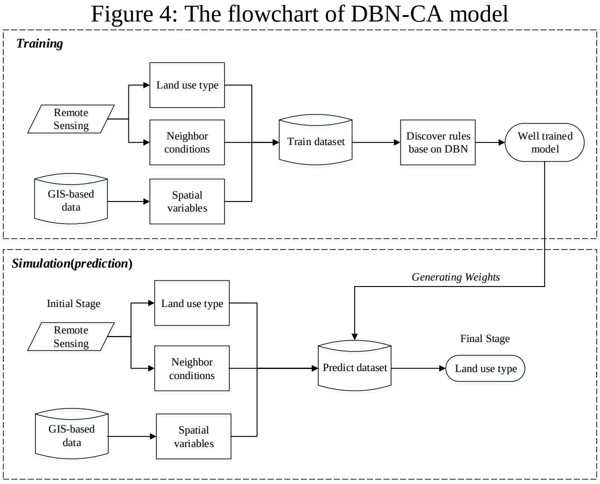
When using CA model to simulating multiple land use changes, the transition rule that determines what land use type of cell will be in the predict time-point can be highly complicated (see Figure 3).

Figure 3: Complex relationships of land use conversion



It consists of two parts (see Figure 4): discovery of transition rules based on DBN from train data and simulation of multiple land use changes based on the well-trained model.

Figure 4: The flowchart of DBN-CA model



The first part of the DBN-based CA model is to train DBN for discovering transition rules from land use data and GIS-based spatial variables. The simulation is cell-based and each cell is composed of a set of attributes as the inputs to the DBN after covert them into the range of [0,1] (Gong, 1996). The attributes during discover transition rules of multiple land use change are shown in Table 1. And They can be expressed by:

$$X = [x_1, x_2, x_3, \ldots, x_n\,]^T \qquad (5)$$

where x_i is the i th attribute and T is transposition. Accordingly, each cell has its own status which is land use types that can be expressed by:

$$Y = [y_1, y_2, y_3, \ldots, y_n\,]^T \qquad (6)$$

where y_i is the ith status and T is transposition. Then use inputs and outputs train the DBN, and save it as the well-trained model when the literation is done.

The second part is to simulate (or predict) the process of multiple land use change by applying the well-trained model. The inputs are land use data and GIS-based spatial variables in the initial stage, and the outputs are the land use type of each cell in final stage.

3 Implementation and results

3.1 Study area

The BTH region (36°05′ N~42°37′ N and 113°11′ E~119°45′ E) is located in the northern North China Plain and includes two centrally directly-controlled municipalities (Beijing and Tianjin) and Hebei Province. The total terrestrial area of the region is about 0.22 million km^2 (see Figure 5). It is the biggest urbanized region in Northern China. With a rapid urbanization and economic development, its land use structure has experienced, and will continue to experience dramatic changes.

3.2 Data preprocessing

The spatial data selected for this simulation consisted of land use datasets and GIS-based data (Table 1). TM satellite images are used to investigate the actual land use structure and its neighbor conditions. The cell size of 250×250 is adopted for implementation. A series of spatial variables include various distance-based variables and physical properties (White and Engelen, 1993) were chosen for the simulation from GIS-based data.

3.3 Experimental design

The major purpose of this implementation is to evaluate the performance of DBN-CA model in simulating multiple land use changes. In this paper, the parameters in Table 2 are set to run the DBN. Cells converted into cropland, woodland, grassland, water body, build up and others are marked as 1, 2, 3, 4, 5, 6 respectively.

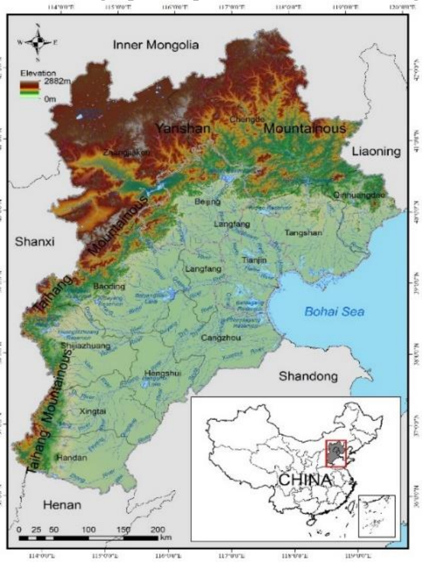
The spatial data in 2005 and 2010 are treated as the training dataset, which are used to train the model for deriving the transition rules, and the spatial data in 2015 are used to confirm the predictability of the well-trained DBN-CA model.

3.4 Model validation and comparison

The simulated results for years 2015 are validated by comparing them with the actual land use structure on cell-by-cell which is derived from the TM data. The visual comparison indicates that the simulated structures are in good accordance with the actual ones (see Figure 6).

It is further validated by comparing with Logistic Regression based CA (LR-CA) in the same dataset. The simulation results of these different scenarios are shown in Figure 7. The confusion matrix, the overall accuracy and the kappa coefficients of the cell-by-cell comparison (Congalton 1991) with the simulation results of DBN-CA and LR-CA are given in Table 3 and Table 4. As shown in Tables 2 and 3, the overall accuracy of the DBN-CA model is 1.48% higher than that of the LR-CA model, and the kappa coefficient is 1.9% higher than that of the LR-CA model. The comparison with LR-CA model indicates that the DBN-CA model is more accurate in the simulation of multiple land use changes.

Figure 5: Geographical position of the BTH region



4 Conclusion

CA models have been widely used in simulating geographical processes. With a large set of spatial variables and complex relationship, the traditional methods are insufficient for calibrating their parameters and interpret their meaning for a large complicate region. This paper presents a novel neural network model, deep belief network, for discovering the nonlinear relationship between spatial variables and geographical processes.

The proposed DBN-CA model is applied to simulating multiple land use changes in BTH region, China. Simulation results have been quantitatively evaluated by confusion matrix, overall accuracy, and kappa coefficient. And it also achieved a considerably higher overall accuracy and kappa coefficient when compared with LR-CA. The results indicate that the CA model based on DBN is a suitable tool for simulating multiple land use changes.

5 References

Almeida, C., Gleriani, J., Castejon, E. and Soares‐ Filho, B. (2008). Using neural networks and cellular automata for modelling intra‐ urban land‐ use dynamics. *International Journal of Geographical Information Science*, 22(9), pp.943-963.

Batty, M. (1997). Cellular Automata and Urban Form: A Primer. *Journal of the American Planning Association*, 63(2), pp.266-274.

Clarke, K., Hoppen, S. and Gaydos, L. (1997). A self-modifying cellular automaton model of historical urbanization in the San Francisco Bay area. *Environment and Planning B: Planning and Design*, 24(2), pp.247-261.

Congalton, R. (1991). A review of assessing the accuracy of classifications of remotely sensed data. *Remote Sensing of Environment*, 37(1), pp.35-46.

He, C., Zhao, Y., Tian, J. and Shi, P. (2013). Modeling the urban landscape dynamics in a megalopolitan cluster area by incorporating a gravitational field model with cellular automata. *Landscape and Urban Planning*, 113, pp.78-89.

Hinton, G. (2006). Reducing the Dimensionality of Data with Neural Networks. *Science*, 313(5786), pp.504-507.

Jenerette, G. D., & Wu, J. (2001). Analysis and simulation of land-use change in the central arizona – phoenix region, usa. *Landscape Ecology*,16(7), 611-626.

Ji, N., Zhang, J. and Zhang, C. (2014). A sparse-response deep belief network based on rate distortion theory. *Pattern Recognition*, 47(9), pp.3179-3191.

Larochelle, H., Erhan, D., Courville, A., Bergstra, J., & Bengio, Y. (2007). An empirical evaluation of deep architectures on problems with many factors of variation. (Vol.227, pp.473-480). ACM.

Li, X. and Gar-On Yeh, A. (2004). Data mining of cellular automata's transition rules. *International Journal of Geographical Information Science*, 18(8), pp.723-744.

Li, X. and Yeh, A. (2002). Neural-network-based cellular automata for simulating multiple land use changes using GIS. *International Journal of Geographical Information Science*, 16(4), pp.323-343.

Liu, X., Li, X., Liu, L., He, J. and Ai, B. (2008). A bottom-up approach to discover transition rules of cellular automata using ant intelligence. *International Journal of Geographical Information Science*, 22(11-12), pp.1247-1269.

Liu, X., Li, X., Shi, X., Wu, S. and Liu, T. (2008). Simulating complex urban development using kernel-based non-linear cellular automata. *Ecological Modelling*, 211(1-2), pp.169-181.

Qin, M., Li, Z. and Du, Z. (2017). Red tide time series forecasting by combining ARIMA and deep belief network. *Knowledge-Based Systems*, 125, pp.39-52.

Santé, I., García, A., Miranda, D. and Crecente, R. (2010). Cellular automata models for the simulation of real-world urban processes: A review and analysis. *Landscape and Urban Planning*, 96(2), pp.108-122.

Shen, F., Chao, J. and Zhao, J. (2015). Forecasting exchange rate using deep belief networks and conjugate gradient method. *Neurocomputing*, 167, pp.243-253.

Silva, E. and Clarke, K. (2002). Calibration of the SLEUTH urban growth model for Lisbon and Porto, Portugal. *Computers, Environment and Urban Systems*, 26(6), pp.525-552.

Torrens, P. and O'Sullivan, D. (2001). Cellular Automata and Urban Simulation: Where Do We Go from Here?. *Environment and Planning B: Planning and Design*, 28(2), pp.163-168.

Verburg, P., de Nijs, T., Ritsema van Eck, J., Visser, H. and de Jong, K. (2004). A method to analyse neighbourhood characteristics of land use patterns. *Computers, Environment and Urban Systems*, 28(6), pp.667-690.

Wang, H., Wang, G., Li, G., Peng, J. and Liu, Y. (2016). Deep belief network based deterministic and probabilistic wind speed forecasting approach. *Applied Energy*, 182, pp.80-93.

White, R. and Engelen, G. (1993). Cellular Automata and Fractal Urban Form: A Cellular Modelling Approach to the Evolution of Urban Land-Use Patterns. *Environment and Planning A*, 25(8), pp.1175-1199.

Wolfram, S. (1984). Cellular automata as models of complexity. *Nature*, 311(5985), pp.419-424.

Wu, F. (1998). Simulating urban encroachment on rural land with fuzzy-logic-controlled cellular automata in a geographical information system. *Journal of Environmental Management*, 53(4), pp.293-308.

Wu, F. (1998). SimLand: a prototype to simulate land conversion through the integrated GIS and CA with AHP-derived transition rules. *International Journal of Geographical Information Science*, 12(1), pp.63-82.

Yang, Q., Li, X. and Shi, X. (2008). Cellular automata for simulating land use changes based on support vector machines. *Computers & Geosciences*, 34(6), pp.592-602.

Yang, J., Tang, G., Cao, M. and Zhu, R. (2013). An intelligent method to discover transition rules for cellular automata using bee colony optimisation. *International Journal of Geographical Information Science*, 27(10), pp.1849-1864.

Table 1: Input and output of DBN

X(inputs)		Y(output)	
x_1	Land use type of the cell	y_1	Cropland
x_2	Cell number of cropland in neighborhood		
x_3	Cell number of woodland in neighborhood	y_2	Woodland
x_4	Cell number of grassland in neighborhood		
x_5	Cell number of water body in neighborhood	y_3	Grassland
x_6	Cell number of build-up in neighborhood		
x_7	Cell number of unused land in neighborhood	y_4	Water body
x_8	Distance to city centre		
x_9	Distance to main road	y_5	Build up
x_{10}	Distance to rivers		
x_{11}	Distance to towns	y_6	Others
x_{12}	Distance to railways		
x_{13}	Slope		

Table 2: Parameters used in the DBN

Description	Symbol	Quantity
Dimension of the input of the DBN	*n_ins*	7
Dimension of the output of the DBN	*n_outs*	[10, 8, 6]
Intermediate layer size, must contain at least one value	*hidden_layers_sizes*	6
Number of epoch to do pretraining	*pretraining_epochs*	0.1
Maximal number of iterations of run the optimizer	*training_epochs*	0.01
Learning rate to be used during pre-training	*pretrain_lr*	10
Learning rate used in the finetune stage	*finetune_lr*	100

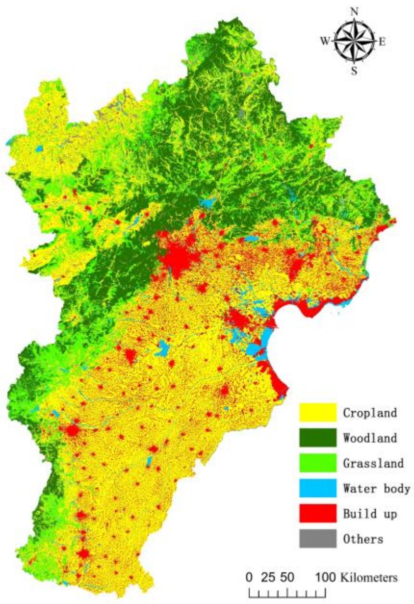
Table 3: Confusion matrix between actual and simulated land use structure based on DBN-CA

	Actual cropland	Actual woodland	Actual grassland	Actual water body	Actual build up	Actual others
Simulation cropland	**89.18%**	0.88%	0.71%	0.49%	8.69%	0.05%
Simulation woodland	1.34%	**94.94%**	2.27%	0.13%	1.30%	0.02%
Simulation grassland	3.48%	3.49%	**89.40%**	0.27%	2.69%	0.66%
Simulation water body	15.48%	2.05%	1.18%	**67.62%**	12.38%	1.29%
Simulation build up	14.31%	0.29%	0.39%	3.23%	**81.70%**	0.08%
Simulation others	29.60%	2.74%	16.31%	2.49%	4.80%	**44.05%**
Overall accuracy			**88.32%**			
Kappa coefficient			**0.829**			

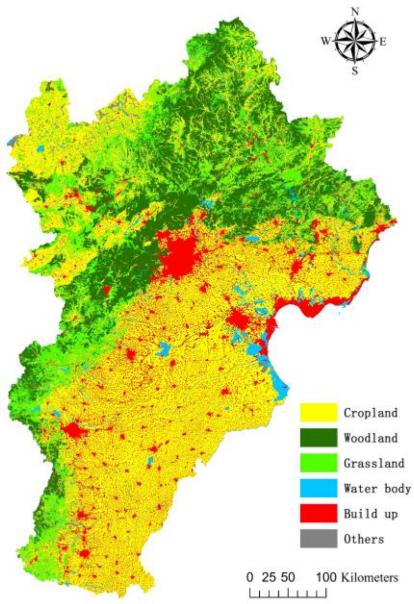
Table 4: Confusion matrix between actual and simulated land use structure based on LR-CA

	Actual cropland	Actual woodland	Actual grassland	Actual water body	Actual build up	Actual others
Simulation cropland	**88.41%**	0.55%	1.14%	0.93%	8.40%	0.56%
Simulation woodland	2.07%	**94.38%**	2.73%	0.28%	0.41%	0.12%
Simulation grassland	2.24%	2.95%	**93.02%**	0.22%	0.61%	0.97%
Simulation water body	8.85%	1.01%	1.67%	**75.82%**	11.77%	0.88%
Simulation build up	27.81%	1.91%	2.75%	2.93%	**64.27%**	0.33%
Simulation others	3.91%	0.61%	18.07%	6.87%	1.89%	**68.65%**
Overall accuracy			**86.84%**			
Kappa coefficient			**0.810**			

Figure 6: Simulated and actual multiple land use changes of BTH region in 2015.

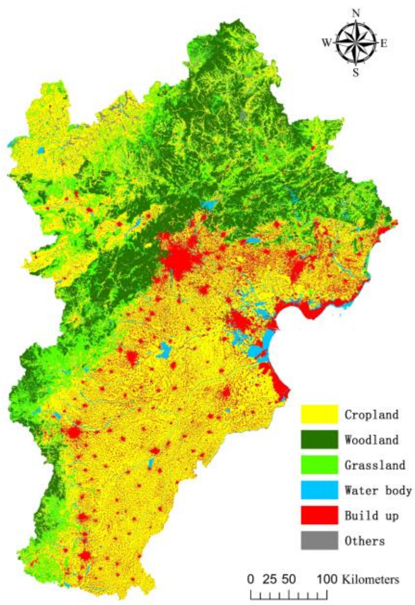


(a) simulation

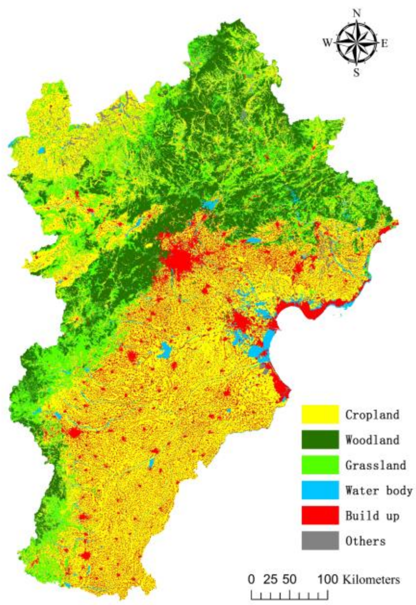


(b) actual

Figure 7: Comparison between DBN-CA and LR-CA



(a) DBN-CA



(b) LR-CA