

A Summary of the Application of Cellular Automata in Urban Spatial Structure and Development

Keyang Cao ¹, Shengquan Wang ^{1, a}, Yihan Jiang ¹, Ang Li ^{1, 2, b}

¹Nanjing University of Science and Technology & ZiJin College, Nanjing City, China

²College of Telecommunications & Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing City, China

^awangshenghquan@hotmail.com, ^bliang878@njjust.edu.cn

Abstract

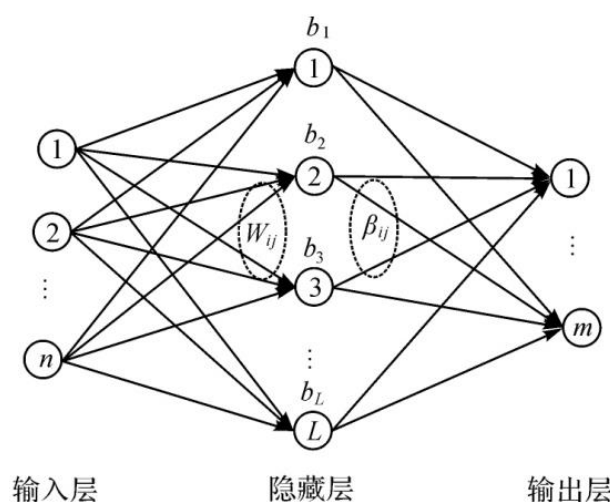
With the development of GIS technology, it provides a basis for the simulation of urban spatial structure and its expansion. Combining in-depth learning with GIS technology can better realize the simulation of urban land use and population. Cellular automata model (ELM-CA) has better simulation accuracy than logistic regression model, and ELM does not need iteration. Therefore, it is more suitable for this kind of research. This paper mainly introduces ELM-CA urban expansion model and some details of GIS in python.

Keywords

Extreme Learning Machine Deep Learning GIS Cellular Automata.

1. Introduction to Extreme Learning Machine (elm)

Extreme Learning Machinery (ELM) is an algorithm for forward-propagating neural networks. It has the following advantages: the connection weights in the input layer and the hidden layer and the paranoid values of the hidden layer neurons are randomly assigned and are not adjusted after the setting is completed. Therefore, it is only necessary to request the connection weight between the hidden layer and the output layer. Compared with logistic regression, the speed of operation is accelerated.



As shown on the left, W_{ij} The weight of the connection between the i -th hidden layer neuron and the j th input layer neuron, β_{ij} Is the connection weight of the i -th hidden layer neuron and the j th output layer neuron. b_i Is the paranoid value of the i -th hidden layer neurons.

First, a single hidden layer neural network is used to illustrate that there are n samples (X_i, Y_i) among them $X_i = [X_1, X_2, \dots, X_n]$, $t_i = [t_1, t_2, \dots, t_3]$. For a single hidden layer neural network with l hidden layer nodes, it can be represented as follows $\sum_{i=1}^L \beta_i g(W_i \bullet X_j + b_i) = o_j$, $j=1, 2, \dots, N$, where $g(x)$ is the activation function, W_i Enter a weight value β_i The weight value for the output, b_i Is a paranoid value. The above formula is expressed by a matrix as:

$$H\beta = Y$$

$$H = \begin{bmatrix} g(W_1 \cdot X_1 + b_1) \cdots g(W_L \cdot X_1 + b_L) \\ \vdots \\ g(W_1 \cdot X_N + b_1) \cdots g(W_L \cdot X_N + b_L) \end{bmatrix} \quad \beta = \begin{bmatrix} \beta_1 \\ \vdots \\ \beta_L \end{bmatrix} \quad Y = \begin{bmatrix} Y_1 \\ \vdots \\ Y_N \end{bmatrix}$$

In the extreme learning machine, the weight value w and the paranoid value are randomly assigned and cannot be modified, so it is only required to solve the above equation, ie $\hat{\beta} = H^+ \cdot Y$ (H^+ Yes H Generalized inverse matrix). Compared with logistic regression, it eliminates the iterative process and is therefore more efficient, and overcomes the drawbacks of the traditional gradient descent method that can only find the local optimal solution.

The following is a comparison of Wang He's training time, test accuracy, and auc indicator in extreme learning machine, neural network, and logistic regression.

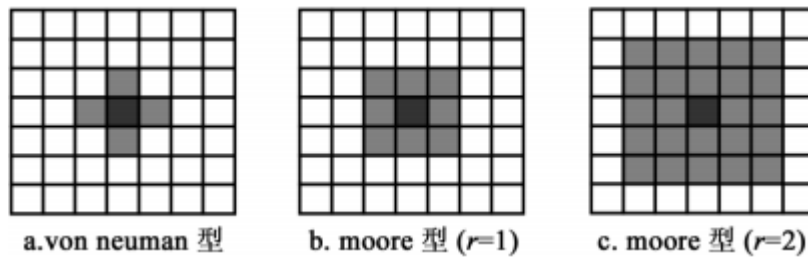
参数	极限学习机	神经网络	逻辑回归
训练耗时/s	376.8	1 168.4	254.5
总体精度	0.820 5	0.792 4	0.733 9
AUC	0.903 4	0.853 0	0.816 4

1.1 Cellular automaton (ca)

The cellular automaton, also known as the cellular automaton, can be simply understood as a rule that uses a square as a cell to establish a relationship with its critical cell. The life game proposed by the early John Horton Conway was a cellular automaton, and later by Stephen Wolfram at A NEM KIND OF SCIENCE

It is described in detail in a book. There is no mathematical formula used to prove the theorem in this book. The process is all done by drawing. And Wolfram himself also stated in the book: "Completely breaking the existing The academic system understands nature in terms of completely different principles." This is also the highlight of the cellular automaton. Although it is now more controversial. However, the cellular automaton has played a role in solving the problem of complexity science. The role of light and heavy, this is recognized by everyone.

The cellular automaton has the following characteristics: 1 cell is distributed in the discrete cell space divided according to certain rules; 2 the evolution of the system is performed step by step at equal intervals, and the time variable takes the same time point; 3 each cell There is a clear state, and the state of the cell can only take a finite number of discrete values; the state value of the next evolution of the cell is determined by the determined conversion rules; the conversion rules of each cell are only in the local domain. The state of the cell is determined. It is described by mathematical language as $A=(L,d,S,N,f)$, A represents a cellular automaton system, L represents the cell space, and d represents the dimension of the cell space in the cellular automaton. A positive integer. S is a finite, discrete set of states of the cell. N represents a collection of all cells in a neighborhood. f represents a local map or a local rule.



The cell space can be understood as a square, each square has its cell state, which can be expressed as {0, 1}, {raw, dead}, {burned tree, unburned tree, open space} or {a,b,c,d}

The field now mainly introduces the field of two-dimensional cellular automata. In general, the two-dimensional field mainly has the following three types: 1.von neuman type 2.moore type 3.margolus type.Among them, the von Neuman type has only four fields of up, down, left and right, and the moore type has a field in which the Moore type is surrounded by the second circle or even the third circle.The margulous type is that it treats 2*2 cell blocks uniformly each time, while in the other two domain types, each cell block is processed separately.Of course, there are also one-dimensional fields or higher-dimensional fields. Here we only discuss the aspects of urban spatial structure and its expansion, so we only need to discuss two-dimensional fields.

The update rule of the state of the cell is like the physical rules of the world in which we have a world of cells.In fact, in the embryonic period of the cellular automata, Neumann wanted to study the self-replicating machine. However, in his time, the life sciences had not fully developed. Later, Conway introduced the cellular automata to Neumann and asked him to make The rule is to create "life" for the world and to study self-replication.So the concept of life games was raised at that time.Then the state update rule refers to a transfer function that determines the state of the cell at the next moment according to the current state of the cell and the state of the cell in the domain, and can be written as

$$f : S_i^{t+1} = f(S_i^t, S_N^t)$$

among them S_N^t The combination of domain states for time t is called local mapping or local rules of cellular automata.

1.2 Elm-ca model

The introduction of the elm-ca model is illustrated by the actual conversion of urban land used by Wang He and Zeng Yongnian (hereafter referred to as Wang).

In Wang's model, the state of the cell is {urban land, cultivated land, woodland, waters, bare land}, and the conversion rules are that rivers, lakes and other waters do not change, and urban land does not convert to non-urban land.The probability of non-urban land conversion to urban land is p, p is determined by the following three factors: 1. Urban land potential P_{ELM} , 2. The influence of the cellular domain $P_{neighbor}$, 3. Random factors P_{random} .

$$P = P_{ELM} \bullet P_{neighbor} \bullet P_{random}$$

among them P_{ELM} It can be expressed by an extreme learning machine, thus transforming the potential of non-urban land (cultivated land, woodland, bare land) into urban land.Each cell unit has n driving factors, which are determined by spatial distance variables (distance from the city) and natural attribute variables.Corresponding to the n neurons of the input layer, the output layer of the limit learning corresponds to the conversion potential.

$$P_{ELM} = \sum_{i=1}^L \beta_i g(W_i \bullet X + b_i)$$

Domain influence factor

$$P_{neighbour} = \frac{\sum_{5 \times 5} con(S_{i,j} = Urban)}{5 \times 5 - 1}$$

among them $S_{i,j}$ Represents the state of the domain coordinates (i, j), and con is the conditional function.

The random factor is a random variable with a function value between 0 and 1.

2. Partial details of GIS in python

2.1 Tile map.

In order to obtain city data, obtaining remote sensing images is an indispensable step. Today, all major map websites store remote sensing images in the form of tiles. A tile map is a multi-resolution map, each tile is composed of 256*256 png format images, and the general tile map is composed of 22 levels. The tile maps are transformed into a planar coordinate system by a conventional earth-like coordinate (WGS-84) and then changed into a tile coordinate system. The formula for changing from WGS-84 to the tile coordinate system is given directly here.

$$X = \left[\frac{lon + 180}{360} \times 2^z \right] \quad Y = \left[1 - \frac{\ln(\tan(lat) + \sec(lat))}{\pi} \times 2^{z-1} \right]$$

The tile coordinate system is determined by (x, y, z), x and y represent the plane coordinates on the map, and z represents the level. Lon stands for precision and lat stands for latitude.

2.2 Google Maps url resolution address

Here are six Google map modes, road maps (mmaps, m), satellite maps (satellite, s), tagged satellite maps (satellite+labels, y), topographic maps (terrain, t), labeled terrain Figure (terrain+labels, p), label layer (labels, h). There are four key parameters (x, y, z, s) in the domain name address of Google Maps. (x, y) is the coordinate, z is the zoom level, and s is the map type.

序号	URL 解析地址	图层标识	地图种类
1	http://mt0.googleapis.com/vt? lyrs=m&x=48&y=96&z=8	lyrs=m	roadmap
2	http://mt0.googleapis.com/vt? lyrs=t&x=48&y=96&z=8	lyrs=t	terrain
3	http://mt0.googleapis.com/vt? lyrs=p&x=48&y=96&z=8	lyrs=p	Terrain + labels
4	http://mt0.googleapis.com/vt? lyrs=s&x=48&y=96&z=8	lyrs=s	satellite
5	http://mt0.googleapis.com/vt? lyrs=y&x=48&y=96&z=8	lyrs=y	satellite + labels
6	http://mt0.googleapis.com/vt? lyrs=h&x=48&y=96&z=8	lyrs=h	labels

2.3 GDAL library in python

The GDAL library is a commonly used tool for splicing tile maps in Python, and is also a commonly used library in GIS. But when we download the tile map, we will encounter a lot of map images, which also causes troubles for splicing tile maps. So when downloading, we can mark the coordinates of the tiles (x, y) and use the name to Named, this will facilitate future stitching. The following part of the GDAL stitching code.

```
Gdal stitching part of the code
try:
from osgeo import gdal
except ImportError:
import gdal
gdal. All Register()
driver = gdal. GetDriverByName( format)
dst_ds = driver. Create( out_file, xtotsize, ytotsize,
```

```

3, gdal.GDT_Byte) #新建大图
Src_ds = gdal.Open (tilename, GA_ReadOnly) # traversal read
Take each tile
data = src_ds.  Read R aster( s_xoff, s_yoff, s_xsize, s_
ysize, t_xoff, t_yoff, GDT_Byte)
dst_ds . Write R
aster( t_xoff, t_yoff, t_xsize, t_ysize, data, t_xsize, t_ysize, GDT_Byte)
src_ds = None
dst_ds = None
Where: t_xoff, t_yoff are the offsets of the tile in the large image.
For each tile, s_xoff = s_yoff = 0, s_xsize = s_ysize = t_xsize = t_ysize.

```

3. Summary

The cellular automaton was only studied in the 1980s, and its field system science was first proposed in the 1930s. Therefore, the cellular automaton is still a very young subject. Especially in the field of social sciences and geosciences, it has played a huge role. Combining the extreme learning machine with the cell state of the cell unit can speed up the analysis of the cell state. Because the author's level is limited, it can only introduce the current situation of the use of cellular automata in the GIS field, and it is not suitable for in-depth research. For further study, please refer to other literature.

References

- [1] Wang Danwei, Wang Furong, Huang Kai. Simulation model of high-speed railway train group tracking operation based on cellular automata [j]. Railway standard design, 2018, 62 (04): 87-91.
- [2] He Xin. Modeling and simulation of port traffic flow based on Agent and cellular automata [D]. Dalian Maritime University, 2017.
- [3] Li Yuanmei. Research and application of network public opinion evolution model [d]. Xinjiang University, 2017.
- [4] Zhao Yun. The mechanism and effect estimation of high-speed railway on regional knowledge spillover [d]. Beijing Jiaotong University, 2017.
- [5] Wang Xin. Analysis and prediction of land use change in group towns based on high resolution remote sensing images [d]. Beijing Tong University, 2017.
- [6] Wang Shuaifei. Research on the evolution and emergence mechanism of complex systems based on cellular automata [d]. University of Electronic Science and Technology, 2017.