International Journal of Smart Home
Vol. 9, No. 3 (2015), pp. 253-262
http://dx.doi.org/10.14257/ijsh.2015.9.3.23

An Intelligent Analysis and Mining System for Urban Lighting Information

Guanlin Chen[1,2], Erpeng Wang[1,2], Xinxin Sun[3 *] and Yanqiang Tang[1]

[1]*School of Computer and Computing Science, Zhejiang University City College, Hangzhou, 310015, P.R. China*
[2]*College of Computer Science, Zhejiang University, Hangzhou, 310027, P.R. China*
[3]*Department of Computer Science and Information, Zhejiang University of Water Conservancy and Electric Power, Hangzhou, 310018, P.R. China*
**Corresponding author email: sunxx@zjweu.edu.cn*

Abstract

Urban lighting information plays an important role in digital urban management. In this paper, an IAMSULI (Intelligent Analysis and Mining System for Urban Lighting Information) is proposed. Using data analysis technology, case-based reasoning technology and data mining technology comprehensively, the system aims at providing better decision support for lighting management. IAMSULI is composed of system management, geographical information management, lighting information management, lighting data analysis, comprehensive classification statistics and event management. The system can help us quickly find lighting problems, provide early warning alarm of the problems, and also supply constructive information for the urban development.

Keywords: *Lighting Information, Data Mining, Digital Urban Management, J2EE, CBR*

1. Introduction

In the construction of digital urban management, large amounts of data have been accumulated [1]. How to analyze and mine valuable information effectively in the big data is of great significance for the decision support of urban management [2].

Recently there are some interesting researches on data analysis and data mining for digital urban management. 2010, Yaowu Wang, *et al.*, proposed an urban planning spatial decision support system (UPSDSS) based on exploratory spatial data analysis technology support [3]. 2011, Tsu-Chiang Lei, *et al.*, proposed a spatial information analysis system to investigate the debris flow occurrence [4]. 2012, Changzheng Zhang, *et al.*, proposed a data mining algorithm of association rules and selects the traffic accidents data in a given city in China [5]. 2013, Longfei Wang, *et al.*, proposed a quantitative method for analyzing regional traffic state evolution [6]. In the same year, Kun Hao Tang, *et al.*, proposed a PCCA algorithm which combines PCA and CLARA algorithms for a set of real-time environmental data [7]. 2014, Guanlin Chen, *et al.*, proposed an intelligent flood control decision support system (IFCDSS) using statistical analysis and data mining technology for digital urban management [8].

However, after analyzing these methods, we found that there is little research of data analysis and mining system for urban lighting information in the field of city management.

Urban lighting is the common infrastructure of a city. Lighting in the city plays an important role in traffic safety, social order, people's life and urban landscape, which is not

ISSN: 1975-4094 IJSH

only directly related to energy conservation, environment protection and the life of the masses, but also reflects the management level of the city and the cultural quality.

The main content of this paper is to design and implement an Intelligent Analysis and Mining System for Urban Lighting Information (IAMSULI). To achieve the scientific supervision of city lighting, we have collected a large amount of lighting information and done much analysis, statistics and data mining using these data.

The system is based on the mainstream of the J2EE platform [9] and the MVC pattern. We use Struts2+Hibernate framework, integrate HighCharts graphics plug-in and Baidu Maps API plug-in technology, and take ID3 algorithm and Case-Based Reasoning (CBR) [10] to build a decision tree, which provides the better decision support for lighting management.

2. Overall System Design and Analysis

2.1. Overall System Design

The system is implemented based on the lighting data of the city, taking advantage of statistical techniques, case-based reasoning technology and data mining technology. Our goal is to find lighting problems quickly, provide early warning function of the problems, and offer comprehensive help to effective management of city lighting.

IAMSULI is composed of six modules, and the basic function framework is shown in Figure 1 below.

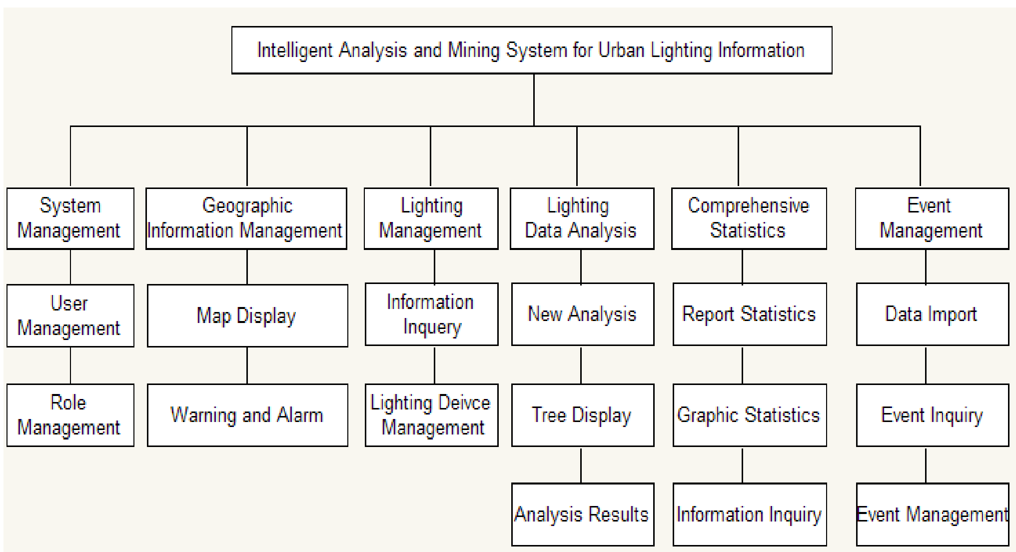

Figure 1. The Basic Framework of IAMSULI

- System management: including user management and role management, which is used to set up user and role permissions;
- Geographic information management: including map display and warning alarm, which is used for the geographic display of lighting information and warning information;
- Lighting information management: including information query and lighting device management , which is used for the CRUD function of lighting information;
- Lighting data analysis: including new analysis, tree display and analysis results, which shows the implementation of data mining functions of lighting data;

- Comprehensive classification statistics: including report statistics, graphical statistics and information query, which implements the diversification of statistical information on the display and information query function after the query selection;
- Event management: including data import, event query and event management. Data import module works for the input of event data in batches. The event management module provides CRUD functions for event information.

2.2. Database Design

The system uses SQL Server 2005 database and there are nine chief tables designed. These tables include lighting (tbl_light), lighting type (tbl_light_type), lighting state (tbl_ligth_state), user (tbl_user), role (tbl_role), unit (tbl_unit), event (tbl_event), event state (tbl_event_state) and region (tbl-region).

The description of the relationships of these tables is as follows: the user in table tbl_user has a role, and the role permission strings are different. The user in table tbl_user is referenced by the table tbl_event as the receiver of an event. The lighting state table tbl_lighting_type and the lighting type table tbl_lighting_type are referenced by the light table tbl_lighting. The lighting table tbl_lighting, the event state table tbl_event_state, the unit table tbl_unit and the user table tbl_user are referenced by the event table tbl_event. The region table tbl_region is referenced by the unit table tbl_unit.

3. Detailed System Design and Implementation

IAMSULI is designed based on the mainstream of the J2EE platform and the MVC pattern. At the same time, it comprehensively utilizes many kinds of intelligent technology for lighting information and event information management, analysis and mining. For example, the system integrates with Baidu Maps API technology to realize the combination of light information and geographic location, and use data mining technology which forms a decision tree to find matching analysis through the analysis of the case library data in the database. The statistical results are displayed in the pie chart and other graphics charts using HighCharts API with pure JavaScript, which show diversified characteristics of data display.

In the following, we will give detailed descriptions to some key modules such as geographic information management module, lighting data analysis module and comprehensive classification statistics module.

3.1. Design and Implementation of Geographic Information Management Module

The geographic information management module is designed to realize the GIS function in IAMSULI, that is to say, display lighting information and early warning information on the map. This function module consists of two functional sub-modules, map display and warning alarm. The map display is implemented with the Baidu Maps API, which is open to all developers.

The map display module accomplishes the display of the geographical position and information of the lighting devices in the database, including the exact latitude and longitude. The Baidu Maps API used in IAMSULI is in the version 1.4.

The JavaScript code of using the Baidu Maps API is as follows.

```
<script src="http://api.map.baidu.com/api?v=1.4" type="text/javascript"></script>
```

The JavaScript code of adding tags in the map is as follows.

```
function addMarker(point,markerInfo){ // latitude, longitude and lighting information
  var marker = new BMap.Marker(point);
  map.addOverlay(marker);
```



```
if(markerInfo.lightStateName == "event processing"){
    marker.setAnimation(BMAP_ANIMATION_BOUNCE); // beating the animation
}
marker.addEventListener("click", function(e) {
    ……// click the add event listener, and the corresponding information is added to the display box
}
```

The system provides functions such as zoom and drag of the map, intelligent address searching, a small map displaying, obtaining longitude and latitude, locating longitude and latitude, lighting searching and satellite map displaying. The concrete realization code is as follows:

```
var map = new BMap.Map("container"); //creating map instance
var point = new BMap.Point(120.2, 30.3); // create initial point coordinate - Hangzhou
map.centerAndZoom(point, 13); // initialize the map, set the center coordinates and map level
map.addControl(new BMap.OverviewMapControl());//add the default thumbnail map control
map.addControl(new BMap.OverviewMapControl({isOpen:true, anchor: BMAP_ANCHOR_TOP_RIGHT}));// open the thumbnail map in the upper right corner
map.addControl(new BMap.NavigationControl({anchor: BMAP_ANCHOR_BOTTOM_LEFT, type: BMAP_NAVIGATION_CONTROL_SMALL})); // Add the lower left corner zoom controls, contains only the pan and zoom button
map.addControl(new BMap.MapTypeControl({mapTypes: [BMAP_NORMAL_MAP,BMAP_HYBRID_MAP]})); //2D map, satellite map
map.addControl(new BMap.MapTypeControl({anchor: BMAP_ANCHOR_TOP_LEFT}));/the default map control in the upper left corner
map.enableScrollWheelZoom(); // enable wheel zoom, disabled by default
map.enableContinuousZoom(); // enable continuous zoom, disabled by default
map.addEventListener("click", showInfo);// click the add event listener to obtain the latitude and longitude
function showInfo(e){
    if(key == 1){
      $("#longitude").val(e.point.lng);
      $("#latitude").val(e.point.lat);
    }
}
```

When initializing the map, we set the longitude and latitude in Hangzhou city and set the map zoom level to 13 in order to make the initial showing range to the city level. The red labels on the initial map mean the lighting information. By clicking the labels, we will see the corresponding lighting information, among which the beating label implies the lighting is being processed or in some troubles. Meanwhile, we can see the expiring lamps on the map, view the remaining time and remind the replacement of corresponding equipment in advance.

The Baidu Maps interface integrated in IAMSULI is shown in Figure 2.

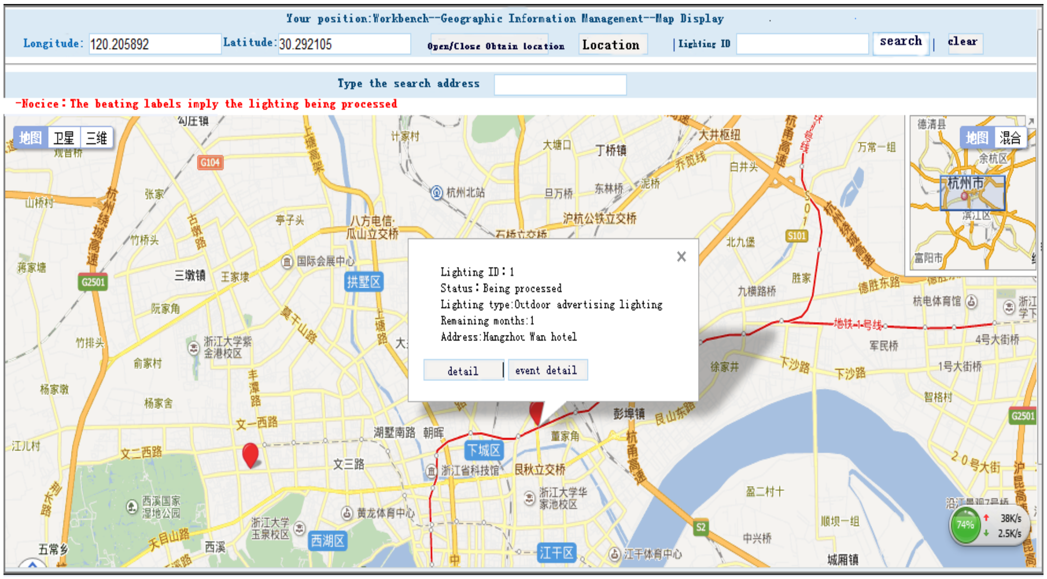

Figure 2. The User Interface Using the Baidu Maps API

3.2. Design and Implementation of Lighting Data Analysis Module

The module of lighting data analysis provides data analysis and data mining functions, using CBR and ID3 algorithm with an inductive indexing method. That is to say, the system builds a decision tree using cases in the case storehouse to mine the valuable information. The system can predict the results according to the analysis of conditions inputted by the user, and provide an important basis for the real-time and accuracy of decision-making on events.

If we want to use CBR in the system, firstly, we must finish some operations such as preprocessing, feature extraction and other operations on the source cases, the specific process is as follows:

(1) Determine the Target of Mining: We use event records of road lighting covering the entire city of Hangzhou, which come from the digital urban management system in Hangzhou. The chief record format is shown in Table 1.

Table 1. The Chief Format of Lighting Records

Name	EventID	EventStateID	UnitID	UserID	LightingID	AddressDes
StandAdd	ReportTime	Problems	CheckType	LightingTime	RepairCost	ProblemSrc

(2) Data Feature Extraction: Data feature extraction is also known as the data reduction or the data dimensionality reduction. Its purpose is to minimize the complexity of the data based on the understanding of tasks and data. Without changing the original data, data feature extraction is able to find the attributes the target requires to make the data more compact and reduce the computational complexity. The attributes used in the system include ‘lighting type’, ‘position type’, ‘region type’, ‘month’, ‘problems’, and so on. Also, we establish the object model of the attributes in the background.

(3) Data Transformation: Data transformation is to establish a suitable mining algorithm of analysis model. For example, the standard addresses in the database have no fixed format, whose number is also uncountable. But we must take discrete values to fit the model, so we convert them to discrete numbers with limited values. In our model, we classify those addresses into city traffic location, business office, leisure and entertainment venue facilities, living facilities and shopping center according to the 'location description' field in an address. As a result, we should add 'position description' field to the event Table tbl_event. Similarly, we divide all the problems into three types: the lack of light, lack of bright hyphenation and other problems. The 'question type' field is added to the event Table tbl_event, whose values need artificial identification and input.

The system takes the data in the event Table tbl_event as the historical case library. Following the steps above, we use the ID3 algorithm to generate a decision tree. The ID3 algorithm needs the calculation of the information entropy.

Entropy is a measure of the disorder. If the probability division of an event A is (A_1, A_2,..., $_n$), and the occurrence probability of each part is (P_1, P_2,..., P_n), then the entropy formula is defined in equation 1:

$$entropy(p_1, p_2, ..., p_n) = -\log_2 p_1 - p_2 \log_2 p_2 - ... - p_n \log_2 p_n \qquad (1)$$

The implementation code of calculating the information entropy is as follows:

```
//Info(T)=(i=1...k)pi*log(2)pi the calculation of entropy attributes
   public static double sigma(int x, int total) {
      if (x == 0) {
         return 0;
      }
      double x_pi = getPi(x, total);
      return -(x_pi * logYBase2(x_pi));
   }
   //log2y
   public static double logYBase2(double y) {
      return Math.log(y) / Math.log(2);
   }
   // PI is the probability of the attributes appearing (= number / total number)
   public static double getPi(int x, int total) {
      return x * Double.parseDouble("1.0") / total;
   }
```

Information gain is a concept based on the information entropy. Entropy is a measure of the chaos of a system, so bigger entropy means more confusion. The branching scheme of ID3 decision tree algorithm is based on the calculation of the information gain of each attribute. We branch the tree by the attribute with the highest gain and repeat above steps to the remaining attributes, eventually forming a decision tree. Information gain is the difference value between two kinds of information, one of which is the information entropy not given the test attribute, and another has been set to a value of the information entropy. The information gain formula is shown as in equation 2.

$$Gain(S, A) = \text{Entropy}(S) - \sum_{v \in V(A)} \frac{|S_v|}{S} Entropy(S_v) \qquad (2)$$

In IAMSULI, we use five kinds of attributes: lighting type, location type, region type, month and problems, among which the first four attributes are taken as the test attributes and 'problems' as the target attribute. We calculate the information gain of these four kinds of attributes and take the attribute with the highest gain to branch the tree recursively until we get a decision tree. The value of test attributes is based on the historic case library, which is not a fixed value.

After we submit a new analysis with some selected conditions, the system will generate a decision tree and show the result in a web page. The tree structure shown here uses the jQuery plug-in of jOrgChart. After matching based on inputted conditions, the system will display the results. In order to compare the results and improve the trustworthiness, the system also looks for the similar cases from the case library.

The decision tree generated in the system is shown in Figure 3 below.

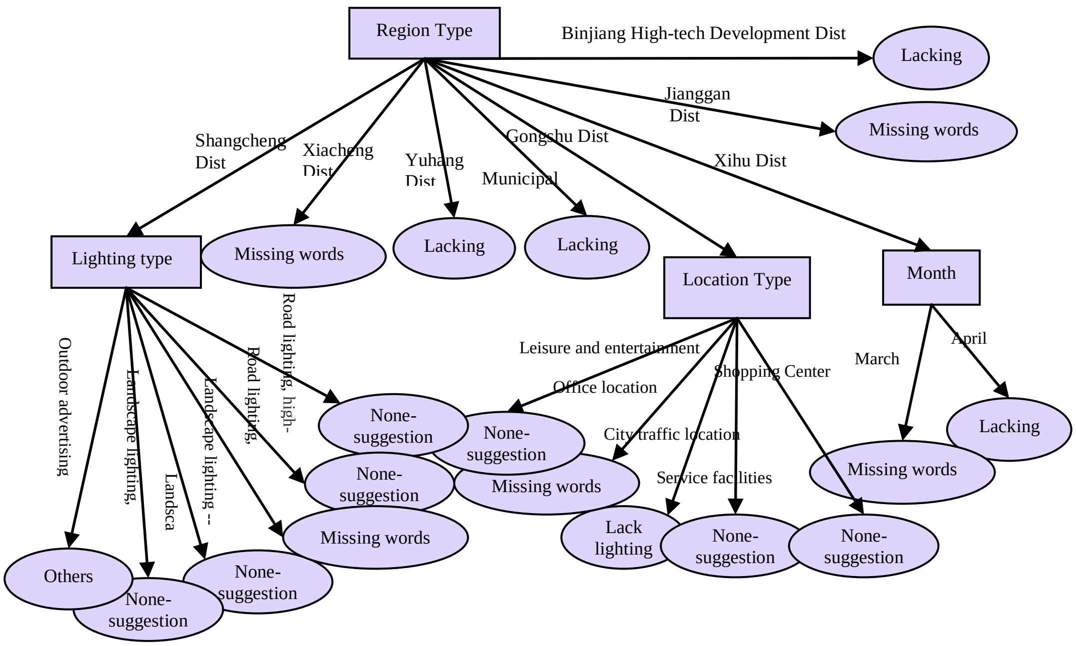

Figure 3. The Decision Tree Generated in the System

We can dig out the following information from the decision tree in the case library:

(1) Lighting type is the primary factor in lighting failure problems. For example, failure of missing words generally occurs in line lamp. But this is not the case in lacking light fault.

(2) There is a great influence on the relationship between underground lamp lighting failure and location type, such as lacking light failures always occur in service facilities, while other failures often occur in entertainment facilities.

(3) Lighting failure with the same lighting type and location varies with time. For example, lighting failure in underground lamps in city traffic location is often lacking light failure in April, May and June, which are not the case in other months.

3.3. Design and Implementation of Comprehensive Classification Statistic Module

The system realizes the classification and statistics functions of lighting and event information in the comprehensive classification statistic module, including statistical statements, information inquiry, statistical charts and other functions. Statistical report information can be exported in the form of Excel. We can query different statistical information through different conditions by the information query function. Chart statistics function can show the statistical information in a pie chart or a histogram form.

Here we focus on the design of chart statistics management module. We use HighCharts API and realize the comprehensive statistics of lighting and event information in this module. The statistical information is displayed in pie charts and histograms, and we can derive the statistical charts in picture format individually.

The user can choose different statistics by classification. According to the event information, we can get statistics by unit or type. According to the lighting information, we can get statistics by type or state. The statistical data is encapsulated into a string of the form JSON in the background. The foreground extracts data from the background with AJAX, and converts it into HighCharts format and displays it.

The following is the JS code of a pie chart statistic:

```
function typeLightPieChart(){
  $.ajax({
    type: "POST",
    url: "lightAction!lightChartCount?condition=type",
    data: null,
    success: function(data){
     var jsonStr = eval("("+data+")");
     var str="[";
     var clmnCate = "[";
     var clmnData = "[";
     $.each(jsonStr.chart,function(idx,item){
            if(idx!=(jsonStr.chart.length-1)){
              str = str + "['"+item.typeName+"',"+item.lightPercentOfType+"],";
              clmnCate = clmnCate +"'"+item.typeName+"',";
              clmnData = clmnData + item.lightCountOfType + ",";
            }else{
              str = str + "['"+item.typeName+"',"+item.lightPercentOfType+"]]";
              clmnCate = clmnCate +"'"+item.typeName+"']";
              clmnData = clmnData + item.lightCountOfType + "]";
            }

     });
     loadPieChat(eval("("+str+")"));
     loadColumnChart(eval("("+clmnCate+")"),eval("("+clmnData+")"));
    }
  });
}
```

The code above resolves the statistical data in JSON format firstly, then converts it to the format HighCharts graph supports, and finally loads it into the map object, so as to realize the dynamic display of graphics statistics.

The interface of chart statistics management is shown in Figure 4.

4. Conclusions

We implement an intelligent urban lighting information mining system (IAMSULI) as shown in the paper, which is a complex system using data analysis and mining methods. We focus on management, analysis and mining of the lighting and event data. Meanwhile, we can show the results diversely in the system, which supports Excel format, map information, tables and charts. In the future, we will continue to expand data sources and introduce the big data mining algorithm in the system, enhancing the applied value of lighting information in the digital urban management.

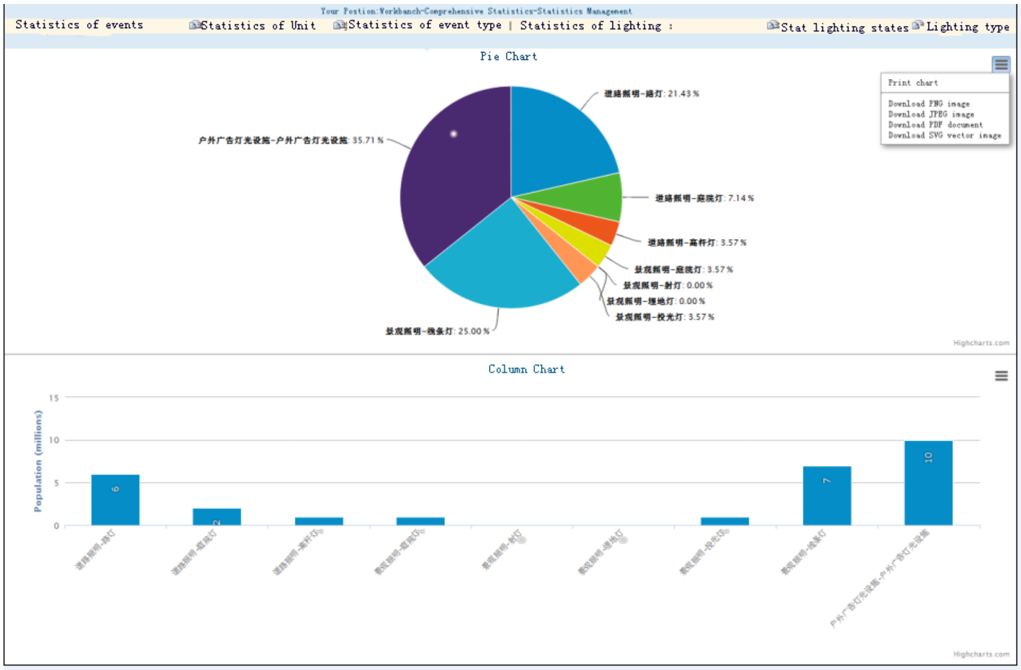

Figure 4. The User Interface of Chart Statistics Management

Acknowledgements

This work is supported by the 2013 Construction Research Project of Zhejiang Provence, China (No. 18) and the Public Welfare Technology Project of Zhejiang Province, China (No. 2014C33060).

References

[1] S. Feng, “Digital urban management of the business system discussion”, Computer Knowledge and Technology, vol. 5, no. 24, **(2009).**
[2] D. Yu and S. Zheng, “Towards adaptive decision support systems: a serviceoriented approach”, Advances in Information Sciences and Service Sciences, vol. 3, no. 7, **(2011).**
[3] Y. Wang and Z. Zou, “Spatial decision support system for urban planning: Case study of Harbin City in China”, Journal of Urban Planning and Development, vol. 136, no. 2, **(2010).**
[4] T.-C. Lei, S. Wan, T.-Y. Chou and H.-C. Pai, “The knowledge expression on debris flow potential analysis through PCA + LDA and rough sets theory: A case study of Chen-Yu-Lan watershed, Nantou, Taiwan”, Environmental Earth Sciences, vol. 63, no. 5, **(2011).**
[5] C. Zhang and S. Wang, “Application of data mining in urban traffic accidents governance based on association rules”, Advances in Information Sciences and Service Sciences, vol. 4, no. 19, (**2012**).
[6] L. Wang, Y. Li, S. Zhang and Q. Zhang, “Urban regional traffic state analysis software system emphasizing pattern transition”, Journal of Software, vol. 8, no. 6, (**2013**).
[7] K. H. Tang, L. Zhong, L. Li and G. Yang, “Monitoring data cleaning of urban tunnels by fusing PCA and CLARA algorithms”, Sensors and Transducers, vol. 159, no. 11, (**2013**).
[8] G. Chen, X. Sun, S. Li, J. He and J. Zhang, “An intelligent flood control decision support system for digital urban management”, International Journal on Smart Sensing and Intelligent Systems, vol. 7, no. 1, (**2014**).
[9] A. S. Boranbayev, “Defining methodologies for developing J2EE web based information systems”, Nonlinear Analysis: Theory, Methods & Applications, vol. 71, no. 12, (**2009**).
[10] B. Lian, Z. Wang and X. Yi, “The application of CBR in digitized urban management based on data mining”, Office Automation, vol. 12, (**2009**).

Authors

Guanlin Chen, Born in 1978, Ph.D., associate professor, chenguanlin@zucc.edu.cn. His main research interests include computer networks, E-government and information security.

Erpeng Wang, Born in 1990, postgraduate, wangerpeng1990zju@163.com. His main research interests include data mining, big data processing and smart city.

Xinxin Sun, Born in 1973, M.E, associate professor, sunxx@zjweu.edu.cn. Her main research interests include computer networks, software engineering.

