Urban Public Traffic Service Model

Benting Wan
Software institute, Jiangxi University of finance and economics, Nanchang, Jiangxi, China
ren_btw@163.com

Abstract

Mobile information services will play an important role in our future work and private life. Enabling mobility in urban and populous areas needs novel techniques for individual traffic planning. However, though there already are WAP-based and WEB-based traffic systems featuring route planning, their usability is not often universal. We presents a SMS-based urban public traffic query service model(SMS-UPSM) offering exact transfer query service and supporting most mobile devices, it can be integrated into WAP-based and WEB-based traffic systems. The SMS-UPSM is composed of bus station library model and least transfer path search algorithm. The least transfer path search algorithm is consisted of direct reach path search algorithm and least transfer converse search algorithm(CS-LT). Bus station library model improves veracity of Chinese parse word algorithm and flexibility of traffic information system, and direct reach path search algorithm can immediately get paths whose bus stations are on the same line, and CS-LT algorithm is based on point to point least transfer algorithm(P2P-LT) which makes use of adjacent and reachable matrixes, it greatly improves transfer algorithm performance. And the implementation results show that the performance of P2P-LS algorithm is better than Dijkstra algorithm and the SMS-UPTS has good efficiency.

1. Introduction

Mobile device has much excellent characters: it is very convenient for bringing and able to gain information anytime and anywhere, and abundant users make use of short message service of mobile phone to communication [1][2][3][4], so it is very useful to supply much more effective and SMS-based services for mobile users, the SMS-UPSM which is presented in the paper is one of the applications about SMS-based services. Currently, there are many public traffic query service information systems[5][6], they are classified into basing on WEB, WAP and SMS and Mix traffic service systems. WEB-based system can not provide services that users can get transfer information anytime and anywhere; and to WAP-based system, the communication cost is higher than SMS-based system, and many types of mobile phone device do not support WAP function; however SMS-based systems can provide cheaper service for users and satisfy most of users communication requirement, but the search result is not exact and performance needs to be improved. Comparing with WEB-based and WAP-based public traffic system, the SMS-UPSM needs to extract source and destination bus stations from short message sent by users[7], in order to gain exact bus stations from short message, some SMS-based systems make use of specify input format, though this method is very exactly for exacting bus station, it can not satisfy users random input requirement; in order to improve short message service practicality, some systems take parse word algorithm to exact bus station from short message[8], however, there are so many strange words to Chinese parse word algorithm that it can not extract exactly bus station from short message, in order to improve parse word algorithm affectively, the bus station library is constructed, which make the SMS-UPSM much more usability.

Many literatures present public traffic search algorithm[9], they improve Dijkstra algorithm performance, however, there are so many bus stations in the public traffic system that the algorithm performance is very low. The direct reach path search and CS-LT algorithm are described in this paper, and the CS-LT algorithm regards a bus line as a point in public traffic system that reduces the search space of algorithm.

Our paper is organized as follows: we focus on the bus station library and least transfer path search algorithm which are consisted of direct reach path search algorithm and CS-LT algorithm in section 2. Section 3 presents an extensive case study and architecture about system. We conclude with a short summary and outlook.

2. SMS-UPSM Service Model

WEB-based and WAP-based public traffic query system includes mainly three steps. (1) Getting source and destination stations from request information; (2) Searching public traffic information; (3) Response search result to user. However, SMS-UPSM includes mainly six steps: (1) Accepting short message sent by mobile user; (2) Filtering SMS format message; (3) Extracting source and destination bus stations; (4) Searching public traffic information; (5) Encapsulating search result; (6) Sending SMS message to mobile user. Compared with WEB-based and WAP-based public traffic query system, the SMS-UPTS needs to parse the short message and extract exact source and destination bus stations from short message. The parse word algorithm makes use of bus station library, which is described in the subsection 2.1. To SMS-UPTS system, the performance of step (4) is crucial, we make use of least transfer path search algorithm which is consisted of direct reach path search and CS-LT algorithm and is described in the subsection 2.2.

2.1. Bus station library model

Currently, many literatures have presented parse word algorithms whose performance have been improved on convergence and veracity [11][12], however, if there are many strange words the parse algorithm is very lack, such as the Chinese message is “三眼井到六眼井的公交”, the parsing results are: “三”, “眼”, “井”, “六”, “眼”, “井”, and “公交”, because the “三眼井” and “六眼井” are strange Chinese words for parse algorithm, so the expectation values are not: “三眼井”, “六眼井”, and “公交”. To Chinese parse word algorithm, on the one hand there exits many strange words about bus stations that leads to parse results error, on the other hand there exits many bus stations which are constructed by some words that leads to parse results mistaken. In order to avoid these error and mistaken, the bus station directory are constructed, which makes parse algorithm quickly and exactly extract source and destination bus station from message.

To a bus station, there exists many alias name besides a formal name. For example, the “江西财经大学西门” is the formal name of bus station, however, the user often use “财大西门” and “江财大西门” etc, in order to make sure of the parse algorithm be able to get the exactly bus station, all bus station names are organized into directory, but when a bus station is deleted and added and updated, and especially finding formal name corresponding alias name, the directory method is very difficult. So we organizes the bus stations into a tree which includes three layers, the root is entry, the second layer are formal bus station name, to each node of second layer, it includes many leaf nodes which are the alias name of bus stations, Fig.1 describes the structure.

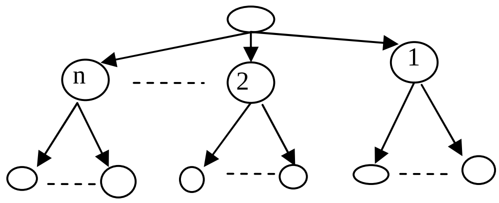


Figure 1. bus stations library model

The model is very convenient for adding and deleting and updating bus stations, such as, if the formal bus station is deleted, its leaves are deleted. And the model is very convenient for creating bus station directory used by parse algorithm. The creating directory algorithm includes two steps: (1) extract all bus station names from library, and the alias names are labeled number of formal name; (2) order the bus stations according to directory order.

The forward max matching parse word algorithm is used to extract bus station from short message, when the alias name of bus station is extracted, then according to the number labeled on alias, the formal bus station name can be immediately gotten, it is regarded as parameter of least transfer path search algorithm.

2.2. Least transfer path search algorithm

If source and destination bus station are the same bus line, the transfer number is regarded as 0, otherwise, the transfer number is greater or equal 1, In order to immediately get the transfer line, the first step is to solve whether or not the source and destination bus stations are on the same bus line, if the result is true, transfer line is direct reach, otherwise we needs to continue search the transfer lines. They are described in the 2.2.1 and 2.2.2 subsections.

Assumption: there are *m* bus stations and *n* bus lines in public traffic system, the set of bus station is $Z=\{z_i \mid z_i \text{ is bus station}\}$, and the bus line set is: $L=\{l_i \mid l_i \text{ is bus line}\}$, to each bus line, the sequence of bus stations is: $l_i=\{z_{li1}, z_{li2}, z_{li3}, \cdots, z_{lik}\}, z_{lik} \in Z$.

2.2.1 Direct reach path search algorithm

It is obvious, to any two bus stations, if they are in a bus line they are direct reach path. In order to solve whether or not the two bus stations are direct reach path, the direct matrix is constructed. The direct matrix is defined by the following formula (1), it can be described by formula (2).

$$t_{ij}=\begin{cases}1 & z_i \in l_k \\ 0 & z_i \notin l_k\end{cases} \quad (1)$$

$$T = \begin{array}{c|cccc} & l_1 & l_2 & \dots & l_n \\ z_1 & 0 & 1 & \dots & 1 \\ z_2 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ z_m & 1 & 0 & \cdots & 1 \end{array} \quad (2)$$

To bus station z_i , we can get bus line set Lz_i which passes through z_i, according to direct matrix definition, if $t_{ij}=1$ and $t_{ij} \in T$, the $l_{ij} \in Lz_i$, So we can immediately get Lz_i and Lz_j which are bus line sets of source bus station z_i and destination bus station z_j, the formula (3) and formula (4) are Lz_i and Lz_j respectively.

$$Lz_i = \{ l_{i1}, \dots, l_{ik1} \} \quad (3)$$

$$Lz_j = \{ l_{j1}, \dots, l_{jk2} \} \quad (4)$$

According to formula (5), we can judge whether z_i and z_j are direct reach or not, if the across set Lz of Lz_i and Lz_j is not null, the across set is the direct bus lines, otherwise, we needs to use transfer algorithm to get the transfer bus lines, which are presented in the 2.2.2.

$$Lz_i \cap Lz_j = Lz \quad (5)$$

2.2.2 Least transfer converse search algorithm

In this paper, the n bus lines are regarded as n nodes of graph. If any two bus lines have the same bus stations the two nodes in graph are direct connected. For example, the fig.2 has 7 points which present 7 bus lines. It is obvious that the node number of bus line graph is much fewer than bus station graph, so the SMS-UPTS has fewer search space that improves the algorithm performance.

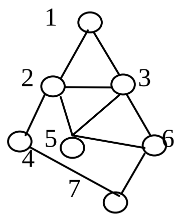


Figure 2. Bus line graph

To bus line graph, the corresponding adjacent matrix is presented by A, and the A^1, A^2, …, A^k are regarded as reachable matrix. According to bus line graph definition, if the element of A is not zero, the transfer number of two nodes is 1, obviously.

Theorem 1: To the reachable matrix A^1, A^2, …, A^k(k is diameter of graph), if the element a^s_{ij} of reachable matrix A^s is zero, the corresponding element a^1_{ij} , a^2_{ij} , …., a^{s-1}_{ij} of the reachable matrix A^1, A^2, …, A^{s-1} are zero.

Proof: Assume there exist an reachable matrix $A^l(1 \le l \le k)$, the a^l_{ij} is not zero, that is to say the $a^l_{ij} \neq 0$, then the value of a^{l+1}_{ij} can be gotten according to formula(6).

$$a^{l+1}_{ij} = \sum_{m=1}^{n} a^l_{im} \times a_{mj} \quad (6)$$

According to reachable theorem $A^R=A^{R-1}*A$ and diameter definition of graph, and the range of l is $1 \le l \le k$. To undirected graph G, the $a_{ii}=a_{jj}=1$, if $a^l{}_{ij} \ne 0$, the $a^l{}_{ij} \times a_{jj} \ne 0$, so we can know that the $a^{l+1}{}_{ij}$ is not zero, what is more, the element $a^{l+1}{}_{ij}$, $a^{l+2}{}_{ij}$, …, and $a^s{}_{ij}$ of A^{l+1}, A^{l+2},…, A^s are not zero respectively, this is contradiction whit assumption, so the assumption is not true and the theorem is true.

According to theorem 1, we can conclude that if the element of A^{R-1} is zero and corresponding position the element of A^R is not zero, the transfer number of two bus station is $R+1$. It is obvious that the element $a^{r-1}{}_{ij}$ of A^{R-1} is zero, According to theorem 1, the elements of $a^1{}_{ij}$, ..., $a^{r-1}{}_{ij}$ are zero, however the $a^r{}_{ij}$ is not zero, so the transfer number is $R+1$ from bus station i to j. To the urban public traffic system, the R is litter than 5 at common condition. For example, the maximal transfer number is 3 to nanchang city of jiangxi province public traffic system.

Given any two nodes i and j in graph, then the minimal rank value l of reachable matrix can be resolved, it satisfies the condition that the $a^1{}_{ij}$ of A^l is not zero and $a^s{}_{ij}=0$ ($0<s<l$). Then we makes use of reachable matrix A, A^2, …, A^l, from back to forward, iterating step by step, the least transfer path are solved from the node i to j of graph. The P2P-LT algorithm is described by the following steps, the algorithm is presented by fig.3.

```
Tree  ShortPath(Matrix A[][][],int l, Station si, Station sj){
Tree  PathTree=null;
PathTree.insert(0,1,(si,sj));
If(l==1){
   return PathTree;
 }
int laycount=1, number=0, count=1;
For(int m=l-1,m>=0;m--){                    // the number of tree layer
   number=laycount;
    laycount=0;
    For(int n=0; n<number; n++){        //the number nodes of each layer
    count=1;
    sj=PathTree.Getvalue(l-m-1,n+1);
    For( int k=0;k<n;k++)
     if(a[m][si][k]&a[0][k][sj]!=0){      //the number sub-nodes are included to each node
      laycount++;
      PathTree.insert(l-m,count,k);
      count++;
     }
   }
 }
}
```

Figure 3. P2P-LT algorithm

(1) If l is 1 and root is null, the least transfer path is (i, j); if l is 1 and root is not null, the source node is inserted into Tree and algorithm returns.

(2) If l>1, the *vals* is calculated by formula (7).

$$vals = a^r{}_{is_1} \,\&\, a^1{}_{s_1 j} \quad (s_1=1,\ 2,\ \ldots,\ n) \quad (7)$$

If *vals* is 1, there exits a path form *i* to s_1 and s_1 to *j*, so the node s_1 is regarded as a child of node *j*. The *r* is *l-1* in the formula(7), if the *vals* is 1, the s_1 is regarded as a child inserting into a Tree.

(3) Replacing *l* with *l-1*, and the destination node *j* is replaced with node s_1, and the root of tree is replaced with child s_1. if *l*=1, then go to step(1), otherwise go to step(2).

From fig.3, the P2P-LT algorithm can get a tree which has *l* layers, and any path from root to leaf node of tree is least transfer path. And The P2P-LT algorithm can be used to solve the least transfer path about any two points in common graph besides bus line graph.

According to P2P-LT algorithm, there exits *h* nodes needing to iterate *n* times, so the time complex of algorithm is O(*hn*). While *h* can be calculated by formula (8).

$$h = Tnodes - \ln odes - snodes - 1 \quad (8)$$

In the fornumla(8), the *Tnodes* presents node number of tree and the *lnodes* presents leaf number of tree and the *snodes* present node number of *l-1* layer of tree. It is obvious that $h<n<m$. so the performance of P2P-LT algorithm is better than Dijkstra algorithm.

If *Lz* is null in formula(5) that presents that the bus station z_i and z_j are not direct reach. In order to find the least transfer bus line from z_i to z_j, the CS-LT algorithm is described as following steps and fig.4.

(1) Getting Descartes production *Lz of* Lz_i and Lz_j, it is described by formula (9).

$$Lz = Lz_i \times Lz_j = \{(l_{i1}, l_{j1}), \ldots, (l_{ik1}, l_{jk2})\} \quad (9)$$

(2)Getting the subset *Z* of *Lz*, to each element of *Z* satisfies the condition: the rank of reachable matrix A^l （$0<l<=k$） is minimal and equal.

(3) Getting the least transfer sequence set.

To each element of *Z*, the least transfer path can be gotten making use of P2P-LT algorithm, all least transfer path are inserted into least transfer sequence set.

(4) Getting the shortest path from least transfer sequence set. If there exits multi-shortest-paths, the number of shortest paths which specified by user are selected.

```
Path TrafficSearch（Station i, Sation j, Matrix T, Matrix A[][]）{
Lzi=findroute(i, T);            // the bus line set through station i
Lzj=findroute(j,T);             // the bus line set through station j
Path=Jion(Lzi, Lzj);            // the intersection of  Lzi  and Lzj
If(Path!=null)
  Return Path;
Else{
  Lz=Dikaer(Lzi,Lzj);                // Cartesian  product of  bus line
  Finditem(Lz, A, D);                // find the element set A from reachable matrixes D
  Path=RecursiveSearch(A, D, Lz);    // recursive least transfer bus lines
   Path=Transfer(Path);              // find the shortest bus lines
  Return Path;
 }
}
```

Figure 4. CS-LT algorithm

If the size of Z is s, the time complex of CS-LT algorithm is O(shn), the s is much litter than n, it obvious the performance of CS-LT algorithm is better than Bellman-Ford algorithm which time complex is O(n^3).

3. Implement

Nanchang city of jiangxi province public traffic query system are implemented, there are 120 bus lines and 1256 bus stations in the public traffic system. Bus stations library are constructed using of XML, the bus line graph is presented by adjacent matrix, and direct matrix. When the bus line graph are changed, the direct matrix, adjacent matrix, reachable matrix are calculated by the system. The implementation results are described by fig.5.

(a) (b)

Figure 5. Implementation results

In Fig.5(a), the public traffic search theme is bus line from economic and finance university to train station of Nanchang Jiangxi, and the Fig.5(b) is the search result. From the implementation result, the system is very useful to mobile users.

The least transfer executing time are compared about one source node reaching each node in bus line graph. In implementation, executing time is average of algorithm running 100 times, the AdjTree presents least transfer path algorithm to all nodes, and the AdjOne presents least transfer path algorithm to one node.

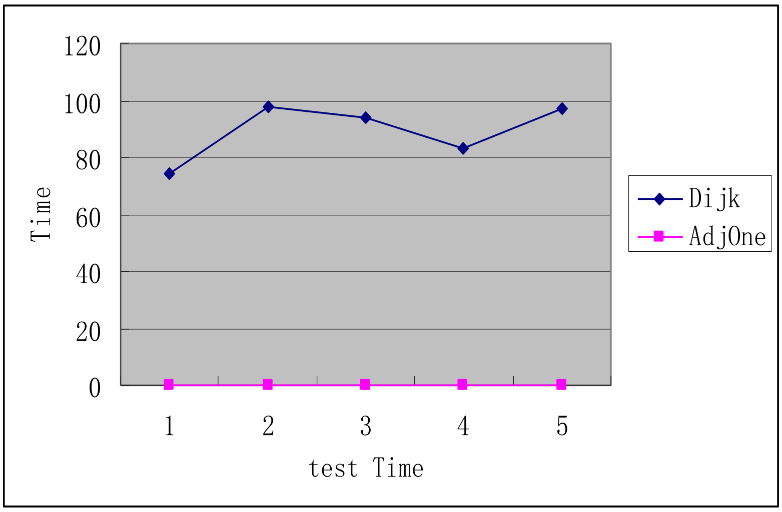


Figure 6. Executing time is compared by Dijkstra and Adjone

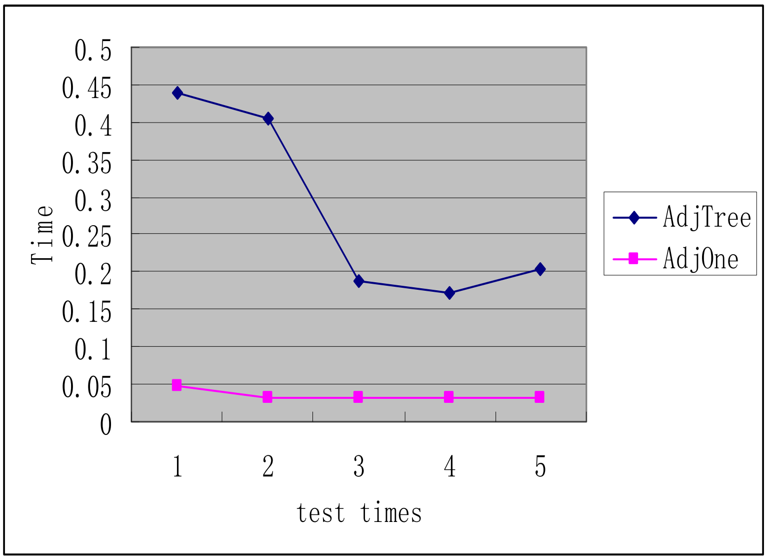


Figure 7. Executing time is compared by AdjTree and Adjone

From fig.6, the executing time of Dijkstra algorithm is much higher than AdjOne, the performance of AdjOne is better than Dijkstra algorithm. From fig.7, the executing time of AdjTree and AdjOne are very few, they are not high 0.5 seconds. The Dijkstra and P2P-LT are implemented, the fig.8 describes the results. In fig.7, the *Adjone* presents that one least transfer path is solved by P2P-LT algorithm, and *AdjTree* presents that all least transfer paths are solved by P2P-LT algorithm, and *Dijk* presents that all least transfer paths are solved by Dijkstra algorithm. From fig.8, the performance of P2P-LT is better than Dijkstra algorithm.

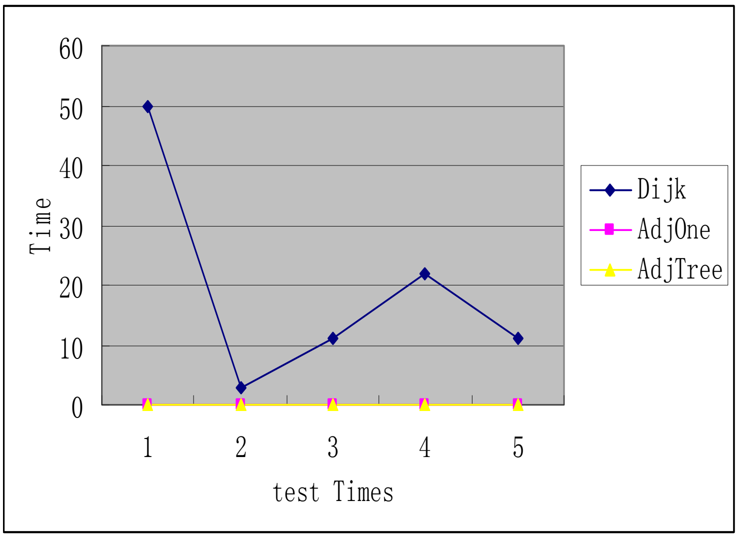


Figure 8. Comparing P2P-LT with Dijkstra algorithm

The average searching time is given in fig.9, the transacting time of AdjTree is not over 800 millisecond when 500 searching services are submission to the SMS-UPSM system.

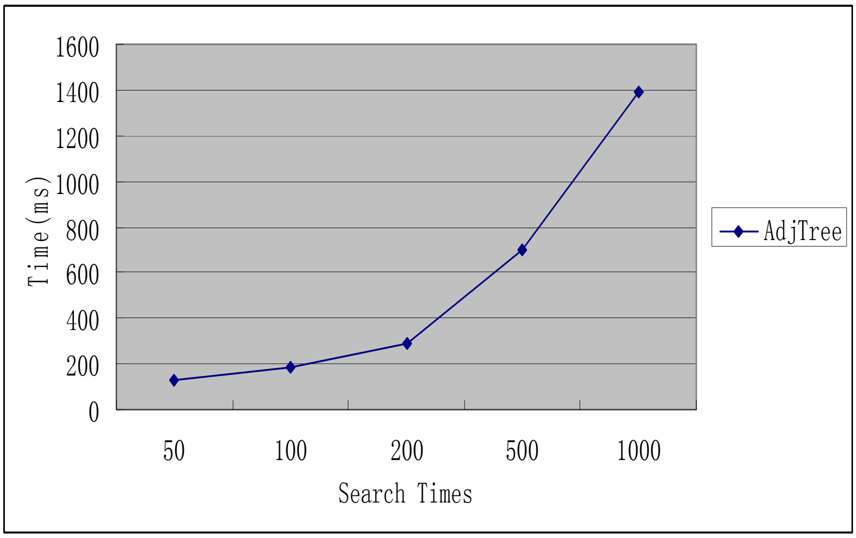


Figure 9. Average searching time

4. Conclusion

The SMS-UPSM supplies short message search service of urban public traffic transfer, it is very convenient to make traffic planning for mobile phone user. The SMS-UPSM makes use of bus stations library to improve extracting bus stations veracity and takes CS-LT algorithm which based on P2P-LT algorithm. And P2P-LT algorithm makes use of adjacent and reachable matrix, they make SMS-based public traffic query system be able to quickly and exactly gain bus line transfer result, so it has good market value.

Acknowledgement

This work is supported by Educational Office of JiangXi Province(No.O7YB191 and GJJ09564). The authors are grateful for the anonymous reviewers who made constructive comments.

References

[1] Xing Xie, Gengxin Miao1,Ruihua Song, Ji-Rong Wen, Wei-Ying Ma. Efficient Browsing of Web Search Results on Mobile Devices Based on Block Importance Model. Proceedings of the Third IEEE International Conference on Pervasive Computing and Communications. pp.17-26, IEEE Computer Society Washington, DC, USA, 2005.

[2] Karen Church, Barry Smyth and Mark. T. Keane. Evaluating Interfaces for Intelligent Mobile Search. W4A at WWW2006 23-26 May 2006, Edinburgh, UK. vol. 134. pp.69-78, ACM Press, New York, NY. 2006.

[3] Rudy Schusteritsch, Shailendra Rao, Kerry Rodden. Mobile Search with Text Messages: Designing the User Experience for Google SMS. CHI 2005, April 2–7, 2005, Portland, Oregon, pp. 1777 – 1780, USA. ACM Press New York, NY. 2005.

[4] Ruihua Song, Haifeng Liu, Ji-Rong Wen, Wei-Ying Ma. Learning Block Importance Models for Web Pages. WWW 2004, May 17-22, 2004, New York, NY USA. pp.203-211, ACM Press, New York, NY. 2004.

[5] W.-T. Balke, W. Kießling, C. Unbehend. A Situation-aware Mobile Traffic Information System. Proceedings of the 36th Annual Hawaii International Conference on System Sciences (HICSS'03),vol.9, pp.292.2, IEEE Computer Society Washington, DC, USA, 2003.

[6] Dimitris Kotzinos, Poulicos Prastacos. Use of a Web-Based GIS for Real-Time Traffic Information Fusion and Presentation over the Internet. 7th AGILE Conference on Geographic Informaiton Scienc 29 April-1May 2004, Heraklion, Greece Poster Session, pp.819-824, 2004.

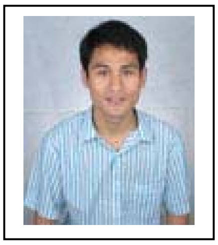
[7] Rudy Schusteritsch, Shailendra Rao, Kerry Rodden. Mobile Search with Text Messages: Designing the User Experience for Google SMS. CHI 2005, April 2–7, 2005, Portland, Oregon, pp. 1777 – 1780, USA. ACM Press New York, NY. 2005.

[8] Gondy Leroy, Hsinchun Chen, and Jesse D. Martinez, A shallow parser based on closed-class words to capture relations in biomedical text. Journal of Biomedical Informatics 36:145–158,2003.

[9] A.V.Goldberg and C.Harrelson. Computing the Shortest Path: A* Search Meets Graph Theory. In proc, 16th ACM-SIAM Symposiym on Discrete Algorithms, pp.156-165, 2005.

[10] LIAO Chujiang, CAI Zhongliang, DU Qi ngyun, WANG Changyao, Modern Real ization of Publ ic Traff ic Optimal Path Based on Least Transfers, Vol. 31 No. 10 pp.904-907, Geomatics and Information Science of Wuhan University, Oct . 2006.

Authors



Benting wan is an instructor in the software institute, Jiangxi university of finance and economics. He received his Ph.D. degree in computer application technology at the china university of petroleum (BeiJing) in 2006. His current research interests include distributed and parallel computing and mobile computing.