Crowdsourcing-Driven Road Condition Marking and Traffic Diversion System

Ming-fei QIN¹, Jing-wei ZHANG¹, Qiu-yu CHEN¹, Cai-qing YAO², Qiong WANG² and Xiao-li HU¹,*

¹School of Computer and Information Security, Guilin University of Electronic Technology, Guilin, Guangxi, China
²School of Electronic Engineering and Automation, Guilin University of Electronic Technology, Guilin, Guangxi, China

*Corresponding author

Keywords: Crowdsourcing, Traffic diversion model, Road width assessment.

Abstract. Crowdsourcing technology has been widely used in navigation platforms, but the data judgement mechanism often causes data misjudgment, which is mainly reflected in two aspects: (1) the criterion is inflexible for judging data; (2) the lack of ability to integrate extreme data. This paper elaborates the dynamic programming data judgement algorithm driven by crowdsourcing mechanism for road condition information, and provides a novel method for eliminating the mechanization. The traffic diversion model and the road width assessment algorithm are further proposed, which enrich the function of the existing navigation systems. Based on the proposed algorithms and model, a road condition marking and traffic diversion system is realized. Theoretical analysis and system testing show that the algorithm can effectively judge the accuracy of the data and improve both the robustness and flexibility of the judgment criterion.

Introduction

Nowadays, road congestion is a common problem in our life. At the same time, smartphones are becoming popular, which provides an excellent opportunity for the crowdsourcing mechanism to be fully applied to the road navigation system.

[1] proposed that the public can be used to solve the problem of mechanized data processing for the mass-driven database systems. [2] mentioned that the main research fields of crowdsourcing focus on database, machine learning , etc. It is very important for the navigation platforms to obtain real-time road condition information, so they used the public resources for data acquisition to obtain a large amount of data at a lower cost.

However, how to effectively filter and check crowdsourcing data has become an urgent problem. This paper designs a judgment criterion based on the dynamic programming, which covers user's adoption value and data voting mechanism to let each person indirectly participate in the development of the judgment criterion.

[3] pointed out that it has become a trend in the future to use big data analysis for intelligent traffic management. Confronted with road congestion, the traditional solution is to maintain order through the manual command of traffic police, but this is inefficient and more passive. Based on the segment point in the route planned by navigation platform, a traffic diversion model and the road width assessment algorithm are proposed, helping users avoid the road segment reasonably where the congestion has occurred, and make up for the deficiency of the existing navigation platform's function.

Crowdsourcing Data Judgment Criterion Based on Dynamic Programming

Quantify User's Credibility

Different from the activeness-based crowdsourcing worker reputation model proposed in [4], which improved the average reputation model and quantifies the crowdsourcing worker's reputation by
using the concepts of active factor and historical factor, this paper puts forward the concepts of the adoption value, the contribution value and the base value, as well as the reverse-order gradient increment and subtraction mechanism of user's contribution value, which can quantify the credibility of crowdsourcing workers more comprehensively and objectively, and thus provide guarantee for the accuracy of crowdsourcing data.

The Base Value. The initial base value of the new user is 60 points. The base value of the old user is given on the first day of the corresponding month by the average value of the user's adoption value in the previous month. The base value of the second day is given by the adoption value of the first day. Based on the above rule, a user’s base value can be obtained by an iterative process.

The Contribution Value. The contribution value will be cleared at the beginning of each month. During one month, the user's contribution value will be increased in the corresponding score according to the level of the adoption value if the uploaded data is adopted. On the contrary, the user's contribution value will be cut back in the way if the uploaded data is judged to be false or beguiling, which is detailed in the section 2.2.

The Adoption Value. The adoption value is the sum of the base value and the contribution value, which is used as the basis for the user's credibility and determines the authenticity of the data.

Operational Method. The adoption value is used as the judgment of the user's credibility. The adoption value is composed of base value and contribution value. The base value is determined by the average value of the user adoption values last month. The contribution value is cleared and recalculated at the beginning of each month. Data fluctuates dynamically in a certain range to ensure that the extremely high or low credibility of users will not trigger the abnormal fluctuation of the base value. The reward and punishment mechanism is introduced for elevating the authenticity of the uploaded data.

The Reverse-Order Gradient Addition and Subtraction Mechanism of Contribution Value

As mentioned in [5], crowdsourcing workers have selfish characteristics and are committed to maximizing their own interests. In order to ensure that crowdsourcing tasks are accomplished with high quality and efficiency, there must be a reward and punishment mechanism in every link of the actual use of crowdsourcing mechanism, which is to motivate excellent crowdsourcing workers while punishing the malicious. This paper proposes a reverse-order gradient addition and subtraction mechanism of user's contribution value, which not only achieves the effect of rewards and penalties, but also is not easy to trigger extreme phenomena.

Supposing that the full score of a user's adoption value is 100 points, which is divided into five score segments, namely 0 to 60, 60 to 70, 70 to 80, 80 to 90 and 90 to 100. Corresponding to the segment of user's adoption value, the user's contribution value is increased in sequence by 2 points, 1 point, 0.5 points, 0.2 points and 0.1 points, and is also reduced respectively by 0.1 points, 0.2 points, 0.5 points, 1 points and 2 points. The reverse-order gradient addition and subtraction mechanism reduces the increment from low to high regularly and increases the decrement from low to high regularly to make the final scores more tally with the standard normal distribution. At the same time, the reverse-order gradient addition and subtraction mechanism is put into application for maintaining the maximize of the confidence interval of the variance concerning adoption values effectively.

Based on the hypothesis that population accord with $X \sim n(\mu, \sigma^2)$, in which $\mu$ and $\sigma^2$ are unknown, $X_n$ is a sample taken from the population $X$. Assuming that the CI of $\sigma^2$ is $1-\alpha$, and the unbiased estimate of $\sigma^2$ is $S^2$, the confidence interval (CI) is defined in Eq. 1.

$$\frac{(n-1)S^2}{\chi^2_{\alpha/2}(n-1)} < \sigma^2 < \frac{(n-1)S^2}{\chi^2_{1-\alpha/2}(n-1)}$$

(1)
The Data Voting Mechanism

**Problem Modeling and Definition.** The crowdsourcing task of road condition remaking is assigned to crowdsourcing workers who have quantified credibility attributes. The uploaded traffic information has the basic attributes of accuracy and error as well. The data voting mechanism binds the attributes of crowdsourcing workers to those of crowdsourcing data. [6,7,8,9] also conveyed that the combination of worker's reputation value and the completion quality of crowdsourcing task can drive the dynamic change of the credibility of crowdsourcing workers and the accuracy criterion of crowdsourcing data. Before discussing how to obtain high-quality crowdsourcing remarking results, the following definitions and assumptions are given.

**Definition 1 (Actual Effective Voting Rate).** In one time unit, which can be one day, one month and one year, the actual effective voting rate is the ratio of the correct information actually accounted for all the uploaded traffic information.

**Definition 2 (Reference Effective Voting Rate).** According to the actual effective voting rate, the reference effective voting rate is defined as the effective voting rate calculated by the data voting mechanism on one day. Therefore, the reference effective voting rate can be multiplied by the number of the uploaded road condition to obtain the reference effective voting number criterion. When the number of people who upload some kind of traffic information of a certain location on the day reaches or exceeds the calculated reference effective voting number criterion, the road condition information of this location is judged to be true.

**Assumption 1.** In the first month of the system operation, the average actual effective voting rate \( L_1 \) is set to be the number of those reported information by all users in this month. By analogy, the average actual effective voting rate in the next month will be \( L_2, L_3, L_4, \ldots, L_n \) in proper sequence. Among them, \( L_n \) is the average actual effective voting rate on the \( n \)-th month.

**Assumption 2.** The reference effective voting rate \( R_1 \) is set to be the number of the reported information on the first day of next month, which is calculated by the average actual effective voting rate value \( L_i \) from last month. By analogy, the reference effective voting rate calculated in the next day will be \( R_2, R_3, R_4, \ldots, R_n \) in proper sequence. Among them, \( R_n \) is the reference effective voting rate on the \( n \)-th day.

**Assumption 3.** The actual effective voting rate \( E_1 \) is set to be the number of the reported information by the user on the first day of each month. By analogy, the actual effective voting rates for each day thereafter will be \( E_2, E_3, E_4, \ldots, E_n \) in proper sequence. Among them, \( E_n \) is the proportion of correct information in all information reported on the \( n \)-th day.

**Problem Solving Algorithm.** When the system is operated in the first month, the majority of the reported information are used to be the identification for judging the accuracy of road condition information. The most uploaded traffic information at a specific location are the most accurate.

However, after obtaining the initial data when the system is put into use, the actual effective voting rate and the reference effective voting rate of every day for each subsequent month will be dynamically programmed according to the past actual effective voting rate and the reference effective voting rate, which are no longer judged as true by the majority. The average \( L_1 \) of the actual effective voting rate of the first month is calculated, and \( L_1 \) is used to be the reference effective voting rate \( R_1 \) of the first day of the second month. The information is adopted in the reported information by \( R_1 \). If it is equal to or exceeds the number of adoptions calculated by \( R_1 \), the information is judged as true. At the end of the first day, the actual effective voting rate value \( E_1 \) is calculated, then the reference effective voting rate value is calculated as \( R_2=(E_1+R_1)/2 \) for the second day, and so on. At the beginning of the third month, the average \( L_2 \) of the actual effective voting rate in all the reported information for the second month is calculated, and \( L_2 \) is used to be the reference effective voting rate \( R_1 \) on the first day of the third month. The values of every month after that can be calculated in the same way.

Thus, we can get the reference effective voting rate every day in the \( n \)-th month. If there are \( m \) days in the \( n \)-th month, there will be calculated as follows,

\[
L_{n,1}=(R_1+R_2+R_3+\ldots+R_m)/m;
\]
\[ R_1 = L_{n-1}; \]
\[ R_2 = \frac{(E_1 + R_1)}{2}; \]
\[ R_3 = \frac{(E_2 + R_2)}{2}; \]
\[ \vdots \]
\[ R_n = \frac{(E_{n-1} + R_{n-1})}{2}; \]
\[ L_n = \frac{(R_1 + R_2 + R_3 + \ldots + R_n)}{m}. \]

Simple Illustration of the Proposed Algorithm

Figure 1. The illustration of the proposed algorithm.

(1) \( L_{10} \) was calculated as the average actual effective voting rate of the previous month (October).

As shown in Fig. 1, the day was supposed to be on October 1, a total of 10 people jointly marked a certain point, and 8 people's credibility reached the criterion. The actual effective voting rate today was \( \frac{8}{10} = 0.8 \). By analogy, No.2, No.3... No.31, finally, the average actual effective voting rate for the whole month of October was calculated as \( \frac{0.8+0.5+\ldots+0.6}{31} = 0.7 \).

(2) \( E_1 \) was calculated as the actual effective voting rate, and \( R_1 \) was calculated as the reference effective voting rate on the 1st of this month (November).

The reference effective voting rate \( R_1 \) on the 1st of this month was the average of the actual effective voting rate of the last month, \( R_1 = L_{10} = 0.7 \). For example, on November 1st, if 10 people marked a certain abnormal point, the number of users should be equal to or more than 7, who reached the credibility value. In such case, this abnormal point information was true. If there may be 8 people actually, the actual effective voting rate was \( \frac{8}{10} = 0.8 \), therefore, \( E_1 = 0.8 \).

(3) \( E_2 \) was calculated from the actual effective voting rate, and \( R_2 \) was calculated as reference effective voting rate on the 2nd of this month (November).

The reference effective voting rate \( R_2 \) on November 2 was the average of the reference voting rate \( R_1 \) and the actual effective voting rate \( E_1 \) on November 1. And the actual voting rate was calculated in the same way.

Algorithm Verification for Applications

Firstly, the number of uploads and adoptions in October was initialized, and then the ratio of adopters to uploaders was equal to the actual effective voting rate in October every day. Several extreme numerical values were set, such as the actual effective voting rate on Oct. 9 was as high as 1.0, while the actual effective voting rate on Oct. 17 was as low as 0.23. The line chart of the actual effective voting rate in October is shown in Fig. 2_October, and the average actual effective voting rate \( L_{10} \) in October can also be calculated by the above algorithm.

Then, the number of uploads and adoptions in November were initialized, and extreme values were set too. For example, the actual effective voting rate on Nov. 3 was as high as 0.88, while the actual effective voting rate on Nov. 2 was as low as 0.26. According to the related algorithms of the above dynamic programming criterion, the actual reference effective voting rate in November was calculated and the reference effective voting rate in November was dynamically programmed. The
line chart of the actual effective voting rate and the reference effective voting rate in November is shown in Fig. 2_November.

Finally, iterative calculation was performed using the same method as in November, and the line chart of the actual effective voting rate and the reference effective voting rate in December is shown in Fig. 2_December.

Figure 2. The line chart of initialized actual effective voting rate data in October and the actual and reference effective voting rate data test in November and December.

We can see that the amplitude of the reference effective voting data line chart in October fluctuates greatly due to the uneven reputation of crowdsourcing workers. If we used the fixed crowdsourcing data judgment criterion, obviously it cannot meet the needs of effectively screening crowdsourcing data.

Then, comparing with the trend and amplitude of the line between the actual effective voting rate and the reference effective voting rate in Fig. 2_November, although the actual effective voting rate in November still was set to be an extreme value, the amplitude of the break line of the reference effective voting rate data in November still fluctuated. The reference effective voting rate in November was not fixed, but conformed to the dynamic changes of previous data rules. The factors causing the changes were both the complex and changeable road conditions and the crowdsourcing groups with uneven pros and cons. We can clearly see that the fluctuation range of reference effective voting rate in November was much smaller than the fluctuation range of actual effective voting rate. Therefore, the dynamic programming criterion algorithm can effectively avoid criterion polarization.

Figure 3. Experimental result data analysis.

Finally, because the data judgment criterion in November were based on the dynamic programming of the past actual data, it drove the adoption value of crowdsourcing workers to follow the dynamic change and float within a reasonable range. It has promoted the credibility of crowdsourcing workers, and in turn positively promoted the reasonable floating of the actual effective voting rate and gradually eliminated and integrated extreme data. The actual effective voting rate and the reference effective voting rate in November and December were compared, and the specific comparison results were shown in Fig. 3. We can see that the maximum value in two months showed a downward trend, and the minimum value showed an upward trend. So the dynamic programming data judgment criterion algorithm can effectively alleviate the excessive or
low fluctuation of data, which has the ability to synthesize extreme data, and has strong flexibility. Further, after making a difference between the maximum value and the minimum value, we can find that the acceleration of data change was controlled. From Fig. 3, it presents that both the maximum downward acceleration and the minimum upward acceleration are showing a downward trend, which shows that the dynamic programming data judgment criterion algorithm has strong robustness and stability.

The Traffic Diversion Model

Based on a large amount of road condition information obtained by crowdsourcing, the traffic diversion model was proposed. [10,11] proposed the road condition information acquisition in no real-time situations, which proposed that statistical data should be used to circumvent the abnormal road segments and guide some traffic flows that should have passed the abnormal road conditions to the surrounding roads, so as to avoid aggravating traffic congestion and other road conditions, and at the same time to alleviate or avoid the occurrence of secondary congestion.

The function of our proposed traffic diversion model is based on the crowdsourcing data after the dynamic programming criterion was reviewed. When calculating the route, it should avoid the congestion point and generate an early warning route to achieve the effect of early diversion and to prevent congestion.

As shown in Fig.4, the address information of the two segment points at the beginning and the end of the abnormal route segment are extracted, then they are changed iteratively with the pattern of superposition based on their latitudes and longitudes.

After the latitude and longitude of the new point were obtained, the starting point is set to be the newly planned point, and the request is sent to navigation platform servers. Servers will return a path or multiple paths. Comparing the new path with the original paths, a judgment is that whether the start of the route abnormal point and the connection data of the previous point are consistent with the original data. If the current route and original route are superimposed, it can not meet the requirements, which should remove the current route data and program the route again.

Finally, the address information of the two segment points of the beginning and the end of the abnormal route segment are used as cardinal number, and a brand-new point is selected to construct the new path.

The Road Width Assessment Algorithm

Problem Introduction

At present, sensors and UAVs (Unmanned Aerial Vehicle) are mostly used for road detection and measurement of road-related attributes. As mentioned in [12], sensor measurements or UAV’s vision are used to provide data for road modeling and detection. Nowadays, the major navigation platforms have not yet provided road width data services, which can be obtained through
crowdsourcing mechanism. A road width judgment algorithm is proposed to assess road width attributes, which can provide more supported data for traffic diversion.

The data obtained through crowdsourcing has two parts. One part is the uploaded information in the system through users with strong subjective initiative. The other part is obtained through hardware facilities with strong objectivity, such as the GPS module, which uploads some coordinate points on the users’ driving track. These coordinate points consist of latitude and longitude with high accuracy. Therefore, an algorithm is needed to utilize these discrete coordinate points to assess the uneven road width in real time.

**Problem Modeling**

As we all known, even on a relatively straight road, the vehicle may not always go in a straight line during the actual driving. Thus, there will appear a lot of turning point on users’ driving tracks. In the case of normal driving, these turning points float around two sides of road within a range defined by two actual road boundaries. As a result, it is impossible to cross the boundary of the road, which can be used to represent the actual width of the road to some extent. This section designs a specific algorithm to assess the approximate width of the road based on these turning point data sets.

In real life, the road cannot be idealized and completely straight. Basically, they are all curved. Therefore, in order to facilitate the processing easily and to make the same kind of algorithm can be applied to most types of road conditions. We need to turn curves roads into straight roads and to use the idea of limit to divide a long and non-curved road into several straight lines, which allows the same kind of algorithm to be applied to different roads.

**Problem Solving Algorithm**

As shown in Fig.5, for a straight line route obtained after dividing by the limitation method, it might as well be set to be four driving turning points in each similar road section. Supposing that the starting point is S, the middle two points are $C_1$ and $C_2$, and the ending point is E.

$$S(x_1, y_1):$$ the starting point in the turning point list.

$$E(x_2, y_2):$$ the end point in the turning point list.
$O(x_3,y_3)$: one point on the semicircular data set.

Semicircle_set: the semicircular data set.

Rectangular_set: the rectangle data set.

Turning_point_list: the turning point list.

Build_Semicircle_set($x_1,y_1,x_2,y_2$): the semicircular data set is generated based on two points.

Get_point(set): a point is taken from the data set.

Build_Rectangular_set($x_1,y_1,x_2,y_2,x_3,y_3$): a rectangular area data set is generated based on three points.

Judge_isowned_set(list,set): It judges whether all points are in a list belong to one data set.

Judge_num_setBound(list,set): It judges the number of points in a list that are on the boundary of the data set.

Calc_length($x_1,y_1,x_2,y_2$): It calculates the distance between two points.

The road width assessment algorithm Pseudo-code are as follows:

1. Begin
2. Initialize:$S(x_1,y_1),E(x_2,y_2),Turning\_point\_list$;
3. Semicircle_set= Build_Semicircle_set($x_1,y_1,x_2,y_2$);
4. do{$O(x_3,y_3)$= Get_point(Semicircle_set);
5. Rectangular_set=Build_Rectangular_set($x_1,y_1,x_2,y_2, x_3,y_3$)}
6. While(Judge_num_setBound(Turning_point_list,Rectangular_set)==0||Judge_isowned_set(Turning_point_list,Rectangular_set)==0))
7. Calc_length($x_2,y_2, x_3,y_3$);
8. End

Algorithm Optimization

The road width assessment algorithm needs to traverse all points on the circle and then to combine with the start and end points in the list of turning points to draw a rectangular area. Time complexity is high, and most of the time are spent in comparisons. We introduce a binary search algorithm based on trend to improve it. Considering that the data for assessing the road width are not ordered, the planned rectangle can contain a trend of increasing or decreasing the number of turning points. Therefore, this trend replaces the condition that the sequence must be ordered in the original binary search algorithm.

![Figure 6](image)

As shown in Fig. 6, one point is selected on the half of the semicircle to be the intermediate point $O_1$, and the point $E$ and the point $S$ are combined to construct a rectangle for determining whether meeting the two conditions that are described in section 4.3. If there is no match, two points are taken on the upper and lower 1/4 semi-circle named with $O_2, O_3$, a new rectangle is constructed, then judgement and calculation are carried out again. The number of turning points included in $O_2$ and $O_3$ are compared. If the number is larger at point $O_2$, the semicircle is reduced to 1/4 circle, the upper half of the 1/4 circle $EO_1$ is output. Otherwise, the semicircle range is reduced to the lower half of the 1/4 circle. Next, the original semicircular data set is replaced with the 1/4 circle data set, and the same method is carried out iteratively until the rectangular interval can contain all the turning points.
Conclusions

This paper contributes a complete description of the algorithms involving in the process of judging, using and enriching crowdsourcing data. An algorithm is proposed for dynamically programming and judging the accuracy criterion of crowdsourcing data, the dynamic programming algorithm is composed of adoption value and the data voting mechanism. After experimentalizing and data testing, the stability and flexibility of the dynamic programming criterion algorithm was proved. Then, based on the crowdsourcing data after the dynamic criterion judging, the traffic diversion model is proposed, and the algorithm for avoiding the congestion point is used to generate the early warning route to achieve the effect of diversion and congestion prevention. Finally, in order to increase the objectivity and diversity of the crowdsourcing data source, based on the user's latitude and longitude coordinates uploaded by the GPS module in real time, the road width assessment algorithm is proposed, which provides a way to enrich the function of the existing navigation platform. An optimization scheme based on a binary search algorithm is also provided to improve algorithm calculation and the speed of searching.

We will continue to enhance the dynamic programming criterion algorithm and consummate the use of the traffic diversion model on the implemented system. At the same time, it is necessary to examine the extraordinary and extreme conditions in the road width assessment algorithm in detail, and give solutions.

Acknowledgement

This research was supported by the National Natural Science Foundation of China (No.61862013, U1501252, U1711263, 61662015), the Guangxi Natural Science Foundation Project (No.2017GXNSFAA198035, 2018GXNSFAA281199), and the National Collegiate Innovation and Entrepreneurship Training Program.

References

[1] Li Guoliang, Chai Chengliang, Fan Ju, et al. CDB: A Crowd-Powered Database System. In Proceedings of the 2017 ACM International Conference on Management of Data(SIGMOD '17), pp.1463-1478.
[2] Feng Jian-Hong, Li Guo-Liang, Feng Jian-Hua. A Survey on Crowdsourcing. Chinese Journal of Computers, 2015, 38( 9) : 1713 - 1726.
[3] Zhang Xi. Overview on the Development of Intelligent Transportation Systems under the Big Data. China Computer & Communication, 2019(01):17-19.
[4] Yan Jun, Ku Shaoping, Yu Chu. Reputation model of crowdsourcing workers based on active degree. Journal of Computer Applications, 2017, 37(07):2039-2043.
[5] Rui Lanlan, Zang Pan, Huang Haoqiu, Qiu Xuesong. Reputation-based Incentive Mechanisms in Crowdsourcing.Journal of Electronics & Information Technology, 2016, 38(07):1808-1815.
[6] Zhong Qiuyan, Liu Zhijuan. EM Evaluation Method of Crowdsourcing Quality Considering the Reputation of Workers. Science and Technology Management Research, 2018, 38(21):70-76.
[7] Ruan Shanshan, Wang Xiaoping, Xue Xiaoping. Crowdsourcing quality control based on reputation model of Dempster-Shafer theory. Journal of Computer Applications, 2015, 35(08):2380-2385.
[8] Jeroen B.P. Vuurens, Arjen P.de Vries. Obtaining High-Quality Relevance Judgments Using Crowdsourcing. IEEE internet computing, 2012, 16(5):20-27.
[9] Peter Welinder, Steve Branson, Serge Belongie, et al. The Multidimensional Wisdom of Crowds. In Proceedings of the 23rd International Conference on Neural Information Processing Systems - Volume 2 (NIPS'10), pp.2424-2432.

[10] He Hanhui, Li Jia, Zhu Jianhe, et al. An Application Study of Computer Simulation in Traffic Diversion Plans. Central South Highway Engineering, 2007, 32(2):99-102.

[11] Chen Yan-yan, Ling Ying, Du Hua-bing, Liu Xiao-ming. Model of Optimum Route Selection in Vehicle Automatic Guidance System Based on Unblocked Reliability Analyses. Journal of Beijing Polytechnic University, 2003(01):39-42.

[12] Hou Yangyang, Wang Huan. Visual Road Detection from UAV Images Based on Stroke Width Transform. Computer & Digital Engineering, 2018, 46(1):182-186.