Neural network-based skyline evacuation route planning algorithm for different amounts of rainfall

Ruei-Ping Wang, Yu-Ren Chen, Yi-Chung Chen, Tsu-Chiang Lei, and Hsin-Ping Wang

***Abstract*--Skyline path planning algorithms are popular location-based services because they can help users find optimal travel routes in a road network based on multiple conditions. At present, many researchers are incorporating these algorithms into evacuation route planning. However, the issue with previous methods was that they could only query the static dimensions of roads, such as road section length or the safety score resulting from a particular amount of rainfall. They could not process dynamic dimensions such as the safety scores of the same route after different amounts of rainfall, which can vary significantly. This study thus proposed a neural network-based skyline evacuation route planning algorithm that can calculate the safety scores of each road section resulting from different rainfalls and identify reasonable evacuation routes. Experiments demonstrate the validity of the proposed approach.**

I. INTRODUCTION

Skyline path planning algorithms have become one of the most popular types of location-based services [1][2][3][4] because they can help users find optimal travel routes in a road network based on multiple conditions. Given a directed graph in which any node s is the origin and node t is the destination, each path p connecting s and t has d dimensional values $[w_1,w_2,\ldots,w_d]$. If the d dimensional values of path p_i are not worse than those of path p_j and p_i has at least one dimensional value that is better than that of p_j, then $p_i \neq p_j$. Furthermore, we say that p_i dominates p_j. The objective of a skyline path planning algorithm is to identify all paths not dominated by any other paths.

Existing skyline path planning algorithms mostly focus on processing static information such as road section length and driving costs but often overlook important dynamic information that users may take into consideration when choosing a route, such as whether the road will be flooded and what its current safety score is. For instance, most users will choose to avoid routes in low-lying areas during heavy rains even if they offer the shortest or fastest route because these routes are likely to be flooded. Clearly, this is a condition that existing skyline path planning algorithms cannot consider, so new methods are needed to take a dynamic safety score into account.

Considering a dynamic safety score in skyline path planning algorithms is difficult because the conditions of each road section must be considered separately. The index must be recalculated as the conditions change, so the amount of computation required can be staggering. Second, the dynamic safety score of a route is not calculated the way a conventional dimension is. For instance, to calculate the total distance of a route, we need only add the distances of all of the road sections in the route together. However, the total risk of a route equals the product of the safety scores of all the road sections. Thus, it does not seem reasonable to use a conventional skyline path planning algorithm to process dynamic values such a safety score.

This work was supported in part by the Ministry of Science and Technology of Taiwan, R.O.C., under Contracts MOST 106-2119-M-224 -003 and MOST 105-2627-M-035-004.

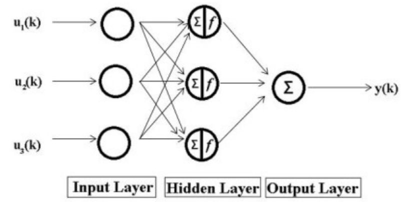

Fig. 1. NN structure used in this work

This study proposes a skyline path planning algorithm based on neural networks (NN) to overcome the aforementioned shortcoming. First, an NN was used to construct the safety score model of each road section. For each road section, different safety factors serve as the input of the NN, and the output is the safety score of the road section. With this model, we can quickly calculate the safety score of each route that a car can take, as NNs generally offer high calculation speeds (time complexity O(n)). With experiments, we demonstrated that the proposed approach can provide users with the optimal evacuation routes in the shortest time.

II. ALGORITHMS

A. Use of NN to construct index model for each type of safety score of all road sections in road network

To use the algorithm, users must (1) choose all of the safety factors that may influence the safety scores of target road sections, (2) construct an NN based on the safety factors chosen in (1), and (3) derive the training algorithm for the model. As Step (1) is performed by the user, we only introduce Steps (2) and (3) here.

The NN used in this study is as shown in Fig. 1. First, the number of neurons in the input layer are set based on the number of key factors chosen by the user. The hidden layer only contains one layer of nodes, the number of which is identical to the number of nodes in the input layer. Of particular note is that the tangent sigmoid is expected to serve as the nonlinear function of this layer. Thus, there will be scenarios in which both positive and negative values symmetric to 0 exist. Finally, the output layer only contains one node, which outputs a value ranging from 1 to -1. A value closer to 1 indicates that a road section is safer, whereas a value closer to -1 means poorer safety. Due to space limitations, refer to [5] for an NN framework formula similar to that of Fig. 1.

In this paper, we used a back-propagation algorithm to train the target NN. The algorithm uses an error to correct the parameters in each layer of the NN. Below is the target equation used to correct the parameters in the NN:

$$Error(\mathbf{w},i) = \tfrac{1}{2}(y_d(i) - y(i))^2 = \tfrac{1}{2}error(i)^2, \qquad (1)$$

where $error(i)$ denotes the error value that can be used to correct the errors in each parameter w. Next, assuming that ξ indicates the magnitude of each parameter correction, then

$$w(i) = w(i-1) - \xi(\partial Error / \partial w). \qquad (2)$$

B. Calculation of safety scores of routes between two points from small road sections

Below, we use the undirected graph in Fig. 2 to explain how the safety scores of the routes between two points are calculated from small road sections. The graph contains three points A, B, and C, so the existing road sections include AB, BC, and AC. The percentage next to each road section is the safety score of each road section, calculated using the NN. For example, the 76% next to Road Section AB indicates there is a 76% chance that a car will safely travel this road section (i.e., with no incidents). To get from A to C, there are two possible routes: Route ABC and Route AC. The likelihood of traveling Route ABC safely is 76%×84%=63.84%, while Route AC only contains one road section, namely Road Section AC, so the likelihood of traveling Route AC safely is equal to the safety score of Road Section AC, which is 96%.

C. Calculation method of skyline paths

The safety score of a route is defined as the product of the safety scores of every road section in a route, which is different from calculating a sum. Thus, previous calculation methods used for skyline paths may not be applicable. To identify the skyline paths between origins and destinations in this paper, we first found the routes with the maximum dimension values and maximum dimension value totals between the origin and destination in the road network (note: If the safety score were a smaller-the-better property, then we would take the reciprocal of the safety score). These routes were added to skyline set **V**, which contains k+1 routes, where k is the current dimension. Next, we used a greedy algorithm to expand the routes, one road section at a time. A dominance test is conducted on the expanded route and the skyline paths in **V**. If the route is dominated, then the expansion of this route ceases. This approach greatly reduces the number of searches required and thereby accelerates queries.

III. SIMULATIONS

For our experiment simulations, we used the road network of Shalu District in Taichung City, Taiwan, which contains 4,268 edges and 3,303 nodes. There was no safety factor data available, so we generated the data ourselves. A total of six dimensions of data were produced. Thus, even with the same

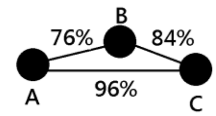

Fig. 2. Example of safety score calculation

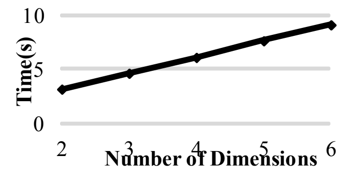

Fig. 3 Time required to predict safety scores of all road sections in Shalu District using NNs

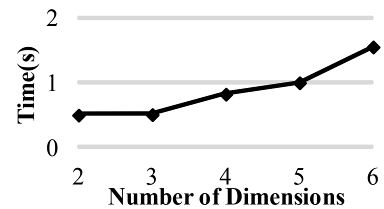

Fig. 4 Skyline path algorithm time graph, origin: 14454, destination: 15176, rainfall: 50

rainfall, six different safety scores must be considered for each road. For each dimension of each edge, we used 100 NNs for training and then selected the best NN.

Figure 3 shows the amount of time taken to predict each safety score of each edge using NNs. In other words, if the entire road network contains n edges and we consider d safety scores for each edge, then the predictions for the entire network require $n\times d$ NNs. In Fig. 3, we can clearly see a linear relationship between prediction time and the number of dimensions, which supports our previous inferences.

Figure 4 shows the time needed to calculate all of the skyline paths from Point 14454 (arbitrary house) to Point 15176 (an evacuation center in Shalu District) with 50 mm of rainfall and different numbers of dimensions. As can be seen, skyline query time increases exponentially with the number of dimensions, thereby supporting skyline path query theories [3].

IV. CONCLUSIONS

The NN-based prediction approach developed in this study can quickly calculate the safety scores of all the roads in a road network and plan skyline paths according to said indexes, which is capable of increasing user safety in the event of disasters. There is more than one kind of safety factor; in future studies we hope to provide users with more routes to choose from based on a wider array of factors.

REFERENCE

[1] X. Huang and C. Jensen, "In-route skyline querying for location-based services," *Proceeding on W2GIS*, pp. 120-135, 2004.

[2] H. P. Kriegel, M. Renz, and M. Schubert."Route skyline queries: a multi-preference path planning approach" *Proceeding on ICDE*, pp.261-272, 2010.

[3] Y. Tian, Ken C. K. Lee, and W. C. Lee, "Finding Skyline Paths in Road Networks," *Proceeding on SIGSPATIAL GIS*, pp.444-447, 2009.

[4] Y. C. Chen and C. Lee, "Skyline Path Queries with Aggregate Attributes," *IEEE Access,* vol. 4, pp. 4690-4706, 2016.

[5] Y. C. Chen and C. Lee, "A Neural Skyline Filter for Accelerating the Skyline Search Algorithms," *Expert Systems,* vol. 32, no. 1, 2015.