International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 6s
DOI: https://doi.org/10.17762/ijritcc.v11i6s.6898
Article Received: 25 March 2023 Revised: 02 May 2023 Accepted: 26 May 2023

Traffic Path Recommendation Model based on a Weighted Sum of Extracted Parameter

Hitendra Shankarrao Khairnar[1*], B. A. Sonkamble[2]
[1]Research Scholar,Dept. of Computer Engineering, SCTR's PICT, Pune
Assistant Professor, MKSSS Cummins College of Engineering, Pune
Savitribai Phule Pune University (SPPU), Pune, India
hitendra.khairnar@cumminscollege.in
[2]Professor, Dept. of Computer Engineering, SCTR's Pune Institute of Computer Technology, Pune
Savitribai Phule Pune University (SPPU), Pune, India
Pune, India
basonkamble@pict.edu

Abstract— A path recommendation for vehicular traffic is important task of traffic analysis. It is a challenging problem for researchers to extract all paths and recommend the shortest path between Origin and Destination (OD) pairs. This paper comes up with a model which is established on the weighted sum of selected link references to recommend a path for OD pairs. First, to maintain spatial dependence between link references, a vehicular traffic network of roads is proposed as a rectangular coordinate system. The algorithm based on K-means and smoothing is introduced to select link references across OD pairs. A distance aggregation algorithm is proposed to evaluate all possible paths across an OD pair. Finally, out of overwhelming paths, the algorithm recommends the shortest distance path across an OD pair. Our proposed model effectively selects the link references and gets an overall shortest path recommendation. The proposed model analyzes the non-Euclidean distance of selected link references. Our experimental analysis shows that on an average, the first four link predictions lead to 77.37% distance coverage for the recommended path.

Keywords- Road Segment, Origin - destination pair, path recommendation, dynamic routing, K-means, Add-one smoothing.

I. INTRODUCTION

The Indian road network is one of the most heavily used vehicular traffic networks in the world, constituting 2% of all roads in India and running 40% of the total road traffic [1]. 72% of people in India are using google maps to know about their driving path. With thousands of new roads being built each day and cities changing traffic rules when necessary, Google maps will always be fighting a battle to stay accurate. A vehicle path recommendation system can never be sure about the shortest distance path without knowing all possible paths across OD pairs. Vehicular traffic analysis analyzes data about vehicular traffic flow, accident rate across road segments, etc. Vehicular traffic path recommendation is the fundamental operation of road traffic analysis. Currently, many travelers use various types of navigation techniques such as Electronic Route Guidance Systems, Global Positioning Systems, and Automatic Vehicle Location to know about the existence of multiple paths between the origin and destination of their travel. Such techniques extract possible paths among OD pairs based on macroscopic and microscopic classification of traffic parameters. Macroscopic classification considers the aggregate behavior of traffic flow referring to volume speed, density, etc. Microscopic models consider the interaction of individual vehicles to deal with parameters such as driver behavior during driving, vehicle locations at different time instances, distance headway, time headway, and the velocity and acceleration of individual vehicles. However, since vehicular traffic depends on multiple static and dynamic parameters traffic path recommendation is a challenging task [2]. Many business verticals like surveillance systems, traffic incident detection, etc. require time series-based traffic data in the form of traffic flow, journey time, speed, distance to be covered, and congestion across multiple paths for analysis point of view. In order to deal with the vertical and horizontal shift of time sequence traffic data, there exist moment and slope-based feature extraction techniques[3][4]. The conditional probability density functions are important for modeling time-varying variability and reliability related to extracted traffic parameters. Non-parametric kernel density estimation enhances the flexibility and accuracy of path recommendation under various complex traffic conditions [5]. Extracted traffic parameters of a road segment are affected by current and past values of traffic parameters at adjacent road segments, and further cascade to the rest of the road network [6]. Generally, traffic parameter sensing devices are located sparsely within the road network. Empirical mode decomposition techniques are suggested to bifurcate traffic parameters into several components with varying frequencies [7]. Different unknown situations like natural

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 6s
DOI: https://doi.org/10.17762/ijritcc.v11i6s.6898
***Article Received**: 25 March 2023 **Revised**: 02 May 2023 **Accepted**: 26 May 2023*

calamities, and congestion may affect travelers' plans about the path to be followed. Advanced information about traffic parameters also influences the performance of path recommendation systems. Hence, it is necessary to recommend multiple possible paths among OD pairs. Whereas according to dynamic rerouting behavior, travelers may change their routes dynamically based on distances to be traveled to reach to destination [8]. Grid-based layout representation of the road network helps to search all possible paths among OD pairs by reconfigure the recommendation about adjacent road segments at road intersection locations [9]. The map-matching process helps to extract multiple paths that exist across OD pairs. The probabilistic route prediction model for map matching is able to identify how much distance is traveled by a vehicle across a grid-based layout [10]. Apart from simple and empirical methods of graph representation, a method to learn optimized graphs through data-driven techniques to explores the relationship between road segments. Such a relationship in turn directs to find undiscovered paths among OD pairs [11].

Sections of the paper are organized as follows: Section 2 briefs about the related work. Section 3 explains the model of our path recommendation system which introduces the construction of a road network using a rectangular coordinate system followed by algorithms for link reference selection, distance aggregation, and origin to destination path recommendation. Section 4 explains the path recommendation results and evaluation through real-world link references examples, model parameters, result analysis, and model interpretation. Section 5 is about the discussion. Section 6 mentions the conclusion and future work.

II. RELATED WORK

Vehicle license plate recognition at sparse locations, is one of the ways to collect vehicular traffic data. Such a data-driven approach provides an alternative way for vehicular data collection traffic parameter estimation and path prediction [12]. A path based on extracted traffic parameters is trending to become privacy-aware and traveller centric as an optimal traffic path for transportation has emerged as a way to improve the performance of the transportation system.[13]. The process of path recommendation was examined on three fronts viz. scope, the process of modeling, and data quality [14]. To represent vehicular traffic networks using graphs, a deep learning framework is proposed to learn interactions between link references of traffic networks. Norms-based graph convolution function helps interpret ability of paths among OD pairs [15]. In order to determine relationships between traffic parameters, vector auto-regression models based on deep learning are suggested [16]. Accurate prediction of real-time traffic parameter prediction affects the efficiency of the overall path prediction system. Model-based on the integration of time series data of traffic parameters and unified two-directional memory networks for path prediction [17]. To overcome the challenges of SRIMA based traffic parameter prediction, a hybrid model consisting of auto-correlation and time series-based traffic parameters is proposed [18]. Common challenges related to traffic parameter estimation and related solutions with reference to deep learning, graph neural networks, and different traffic data sets are discussed [19]. To handle space and time-related characteristics of road traffic demand distribution, a convolution network of dynamic transition (DTCN) is proposed. The DTCN network captures varying traffic dynamics to search possible paths across OD pairs [20]. GPU-based graph processing framework and graph optimization techniques for path prediction related to intelligent vehicular networks along with issues and challenges are presented [21]. Spatio-temporal graph convolution network to extract traffic parameters and study related patterns in the time-space domain [22]. A problem of data sparsity with Markov chain-based models of taxi path recommendation is discussed. Ensemble learning, driven by data, combines support vector regression and deep learning at different routes [23]. The need for path optimization to improve city-wide ride-hailing system performance is highlighted [24]. To handle dynamic routing problems, proximity learning-based ant colony algorithms are proposed to predict the nodes of optimal routes [25]. In order to predict multiple paths across OD pairs, a Taxi-drivers route selection strategy and corresponding GPS-based log file management technique is proposed [26]. Researchers incorporated a belief-desire-intention modeling method in an agent-based artificial transportation system. Such a combination helps to study travelers' psychological characteristics and logical thinking about selections of paths across OD pairs [27]. In Literature, there exist route choice models based on the strategy of maximization of utilities across paths. User equilibrium and stochastic user equilibrium-based models are used to encourage the selection of a portfolio of routes across OD pair [28]. A Comparison of various traffic simulation tools to realize functionalities of traffic parameters and planning for road networks is discussed. Issues and challenges associated with the simulation of road conditions and heterogeneous road transportation networks are also presented [29]. All pairs' shortest paths [30] are used to find the best link reference sequence through which the shortest path can pass through. Our proposed model considers not only the effect of traffic parameters on adjacent link references but also link references up to the destination location, thereby leading to appropriate interpretation and path propagation toward the destination location.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 6s
DOI: https://doi.org/10.17762/ijritcc.v11i6s.6898
***Article Received**: 25 March 2023 **Revised**: 02 May 2023 **Accepted**: 26 May 2023*

III. MODEL OF PATH RECOMMENDATION SYSTEM

The model of the path recommendation system consists of the following phases as shown in Figure 1.

1. Formulation of the graph-based road network: The problem is formulated to design a road network model as a graph, where each edge represents a link that references connecting junctions, or major traffic data collection locations.
2. Link reference selection across OD pairs: Extending training set of link references based on the K-means algorithm.
3. Distance aggregation of selected link references: For computing the total length of all possible paths.
4. Source-to-destination path recommendation based on distance aggregation of link references.

A. *Formulation of Graph-based Road Network*

To model a road network using a rectangular coordinate system and handling distance among link references, the adjacency relationship is used. Assume that the path across origin to destination is described as a collection of link references L_{ix} perpendicular to the horizontal axis: a link

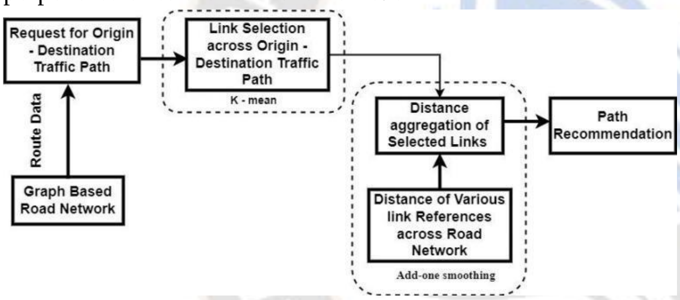


Figure 1. Model of Path recommendation system

reference endpoints L_{ix} is outlined by (X_{i1},Y_{i1}) and (X_{i1},Y_{i2}), where $Y_{i1} \neq Y_{i2}$ or a link reference L_{iy} perpendicular to the vertical axis. The link referenced endpoints L_{iy} are outlined by (X_{i1},Y_{i1}) and (X_{i2},Y_{i1}), where $X_{i1} \neq X_{i2}$. Various link references based on network graph design rules are as follows:

(i) Suppose there are n ($n \leq 5$) link references L_{m1}, L_{m2}, L_{m3}, L_{m4}, and L_{m5} intersecting at some location. Let L_{m1} be outlined as L_{m1x} perpendicular to the horizontal axis. Link reference L_{m2} and L_{m5} are adjacent to L_{m1} are represented by line segment L_{m2y} and L_{m5y} crossing with the L_{m1x} and perpendicular to the vertical axis. Link references L_{m3} and L_{m4} are separately defined by line segments L_{m3x} and L_{m4x} crossing with the line segment L_{miy} (L_{m2y} or L_{m5y}) and perpendicular to the X axis.

(ii) Let three link references L_i, L_j, and L_k intersect at some point. Let link reference L_i is outlined as L_{ix} perpendicular to a horizontal axis. Link reference L_j is outlined as L_{jy} and perpendicular to the vertical axis. Link reference L_k is split into two segments. One is the link segment L_{kx} crossing with line segment L_{ix} and perpendicular to the horizontal axis. Second L_{ky} crossing with line segment L_{jy} and perpendicular to the vertical axis.

The detailing of each path length is as follows: The length of segment L_{ix} (Dist $L_{ix} = |Y_{i2} - Y_{i1}|$) is computed as the amount of line segment perpendicular to the vertical axis between two endpoints of L_{ix}. The length of line segment L_{iy} (Dist $L_{iy} = |X_{i2} - X_{i1}|$) is computed as the amount of line segment perpendicular to the horizontal axis between two endpoints of L_{iy}. The length of all paths across pairs of origin and destination will be the sum of distances of line segments perpendicular to the horizontal axis and vertical axis. Finally, the recommended path will be the shortest path among all possible paths.

Algorithm 1, accords with the main link references of a map thereby describing the modeling of graph-based road networks for path recommendation.

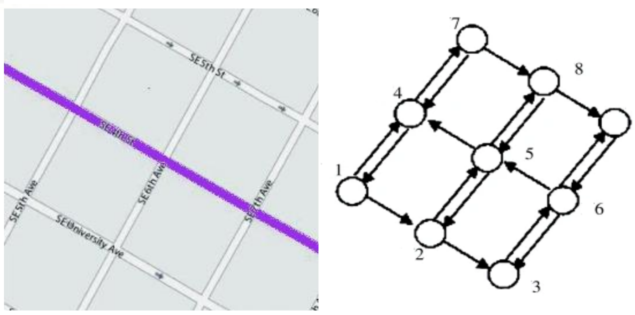
Figure 2 (a) shows a sample of a road network map consisting of nodes and link references to connect nodes. Algorithm 1 transforms the map of *Figure 2(a)* into a rectangular coordinate-based graph network as shown in *Figure 2(b)*.

B. *Link reference selection across OD pair using k-means*

Suppose X and Y as the origin and destination locations. A path composed of ‘n’ link reference L_1, L_2, .. L_n from X to y is formulated as a sequence of n - 1 link references.

Algorithm 1 Building graph based road network

Input: 1. Set of link references connecting road junctions, or major traffic data collection locations
2. Length of each link reference
Output: Graph representing a road model using a rectangular coordinate system and associated distance across link references
steps:
1: Accept origin and destination location to build a graph.
2: flag = TRUE /* (flag = FALSE at destination)*/
3: while (flag == TRUE) do
4: Starting from origin, $\forall$ link reference E_i, identify adjacent node j and edge cost = distance of link reference $L_i(d)$
5: if for $\forall$ subsequent identified nodes j $\neq$ destination location, repeat step 4 and construct Graph based representation of a road network model
6: **if** (j = destination location) **then**
7: flag = FALSE
8: **end if**
9: Return a graph of rectangular coordinate system graph and associated distance across link references.



(a) (b)

Figure 2. (a) A sample road network (b) Representation of a sample network using rectangular coordinate system

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 6s
DOI: https://doi.org/10.17762/ijritcc.v11i6s.6898
***Article Received**: 25 March 2023 **Revised**: 02 May 2023 **Accepted**: 26 May 2023*

$$X \rightarrow Y = L_{1x}\ (L_{1y}) \ + \ L_{2y}(L_{2x}) + .. + L_{(n-1)y}(L_{(n-1)x}) + L_{nx}(L_{ny}) \quad (1)$$

Where location X represents the coordinates of the line segment L_{1x} or $L_{1y.}$ Location Y represents the coordinates of line segment L_{nx} or $L_{ny.}$ In between selected link, references are the junction points of line segments $L_{(i-1)x}$ and R_{iy}.

It is necessary to train a recommendation system based on past history across different time instances of a day. Paths are generally regular to a certain extent. In addition, it is also observed that during certain time periods of the day, traffic reaches its peak, and to avoid congested roads, alternative routes recommendation will be helpful. During the off-peak times of the day, the system will predict all possible paths and recommend the shortest distance route across the origin-destination location.

It is not known how many links are associated with the point *p(x, y).*, *Algorithm 2* extends a new link in the training set using add-one smoothing.

Functioning of Algorithm 2 for link reference selection is as follows

1. Initialize *p* and *p'* to Φ.
2. Traverse an existing training set D of link references across origin (x_{i1},y_{i1}) to destination (x_{in},y_{in}) and insert these locations to *p*. Clean duplicate coordinate points of *p*, get *p'* to be repressed for various origin and destination locations.
3. Use the K-means algorithm to identify link references to be added in *p'*.
4. Traverse each of the link references sets to determine coordinate points associated with the link reference set C_i. If not, Insert link reference (c[i][k], c[j][l]) into NewD.

Algorithm 2 Link reference selection to extend training set

Input: 1. A training set D of link references
Output: Extension of training set to NewD with selection of link reference.
steps: p and p'are coordinate points set, O: Origin location, D: Destination location, path link reference set C.

```
p, p' = ϕ // Initially empty
Extending path across OD pair to set NewD = ϕ
for each path across OD pair (path t_i in D) do
    O = (x_i1,y_i1)
    D = (x_in,y_in)
end for
Insert O and D coordinate into set p;
p' = filter(p); // Filter repeated coordinate of p, which could get the set NewD composed of different starting and endpoints
C = K-means (p');
for ∀ i=1 to n do
    for ∀ j = i+1 to n do
        for ∀ k = 0 to k ≤ c[j].length do
            Compute c[j].length as the number of link references across i^th path of OD pair
            add (c[i][k], c[j][l]) into NewD /* c[i][k] is the k^th link reference in the i^th path across OD pair*/
        end for
    end for
end for
Return NewD
```

C. *Distance aggregation of selected link references using add-one smoothing*

In order to calculate the total length of each possible path across an OD pair, It is necessary to keep track of respective link references across routes. The add-one smoothing technique is used to add one by one link references from NewD set to get the sum of the total length for each possible path.

Working of *Algorithm 3* is given below:

1. Initialize Distance aggregation of selected Link reference = 0.
2. For a link reference access a starting point $A_i(x_{i1},y_{i1})$ and last point $B_i(x_{in},y_{in})$ from l_i. Access a starting point $A_j(x_{j1},y_{j1})$ and last point $B_j(x_{jn},y_{jn})$ from l_j of the link reference of the path. Compute vector a[i] and a[j].
3. Initialize Distance aggregation of selected Link reference = 0.
4. For a link reference access a starting point $A_i(x_{i1},y_{i1})$ and last point $B_i(x_{in},y_{in})$ from l_i. Access a starting point $A_j(x_{j1},y_{j1})$ and last point $B_j(x_{jn},y_{jn})$ from l_j of the link reference of the path. Compute vector a[i] and a[j].
5. Compute angle between vectors
$$\cos(a_i , a_j) = ((x_{in}, - x_{i1}).(y_{in} - y_{i1})) + (x_{jn}, - x_{j1}).\ (y_{jn} - y_{j1})) \quad (2)$$
6. If 0 ≤ cosine value ≤ 1, traverse coordinate points related to L_i and L_j. If -1 ≤ cosine value ≤ 1, L_j and L_i will be part of the opposite direction path.

Algorithm 3 Distance aggregation of selected Link reference

Input: A training set to NewD with selection of link reference
Output: Aggregated distance of selected link references across path.
steps: a: Vector, O: Origin location, D: Destination location

```
Distance sum of selected link reference = 0
for ∀ i=1 to m do /* Within Link reference sequence set NewD, there exist m number of paths across origin to destination location */
    Origin location O_i = (x_i1,y_i1);
    Destination location D = (x_in,y_in)
    Vector a[i] = (x_in − x_i1 , y_in − y_i1)
    for ∀j = i+1 to m do
        Origin location O_j = (x_j1,y_j1);
        Destination location D = (x_jn,y_jn)
        Vector a[i] = (x_in − x_i1 , y_in − y_i1)
        Vector a[j] = (x_jn − x_j1 , y_jn − y_j1)
        if (0 ≤ cos(a[i], a[j]) ≤ 1) then
            for ∀ O_k1 in l_i ∈ NewD do
                for ∀ O_k2 in l_j ∈ NewD do
                    check (O_k1 = O_k2)
                    sum = l_i + l_j /* l_i, l_j are link references length */
                end for
            end for
        end if
    end for
end for
return the aggregated distance of selected link references
```

7. After calculating all link references of set NewD, calculate the distance sum of selected link references to get the length of possible paths.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 6s
DOI: https://doi.org/10.17762/ijritcc.v11i6s.6898
***Article Received**: 25 March 2023 **Revised**: 02 May 2023 **Accepted**: 26 May 2023*

$$path\ length = {}^{n}\Sigma_{i=1}\, l_i$$
$$= length\ of\ selected\ Link\ reference \quad (3)$$

D. Source to destination path recommendation

Figure 3, shows the structure of the shortest path. NewD is processed to discover different link references through which the shortest path passes. For an OD pair, the shortest path passes through a varied number of link references. We define a path function f (O, D, r) as follows.
f (O, D, r) is the shortest path (if there exists one) from O to D that passes through at most r link references.

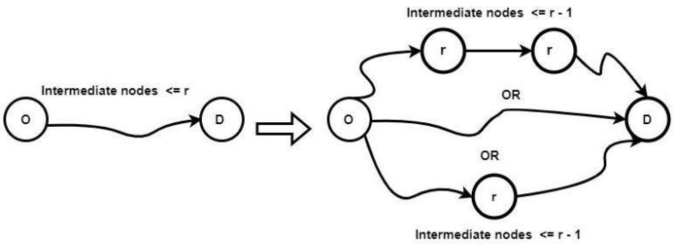


Figure 3. Structure of the recommended shortest path across OD pair

Functioning of *Algorithm 4* is as follows

1. If r = 0, the recommended path f(O, D, 0) must be the direct link reference from O to D.
2. For any integer r > 0, either f(O, D, r) passes through vertex r or it doesn't.

- If f(O, D, r) where r > 0 passes through vertex X, it consists of a subpath from O to X, followed by a subpath from X to D. Both of these subpaths pass through at most r-1 link references. So the two subpaths must be f(O, X, r-m) and f(X, D, r-1-m).
- If f(O, D, r) does not pass through r, then it passes through only link references numbered at most r -1 and it must be the shortest path to recommend. So in this case, f(O, D, r) = f(O, D, r-1)
- Let dist(O, D, r) denote the length of the path f(O, D, r). The recursive structure of f (O, D, r) implies the recurrence as follows

$$dist(O, D, r) = \quad dist\ (O \rightarrow D) \text{ — if } r = 0$$
$$OR$$
$$= \min\{dist\ (O, D, r\ \text{-}1),\ dist(O, X, r - 1) + dist(X, D, r\text{-}1\text{-}m)\} \text{ — Otherwise} \quad (4)$$

Algorithm 4 Shortest path recommendation
Input: Aggregated distance of selected link references across path
Output: Shortest path recommendation across origin destination location
steps: O: Origin location, D: Destination location, r: Intermediate location

```
O ∈ NewD
D ∈ NewD
dist(O, D) : Distance between Origin and Destination
for ∀ link references r starting from O do
    for ∀ link references reaching upto D do
        if dist(O, D, r) > dist(O, r) + dist(r, D) then
            dist(O, D, r) = dist(O, r) + dist(r, D)
        end if
    end for
end for
return dist(O,D,r ) as shortest distance of recommended path
```

IV. PATH RECOMMENDATION RESULTS

A. Experimental platform

Vehicle should be equipped with a device for route data collection. A data collector uses a mobile phone with google maps.

B. Data collection

For various time periods of the day, one-month data about traffic features is collected across 18 link references of possible routes between MKSSS's Cummins College of Engineering, Pune, and College of Engineering, Pune. Traffic features namely travel time, vehicle speed, traffic flow at different time instances, and distance across link references are suitable for traffic path recommendation. Figure 4, shows the map of an example road network along with link references.

Figure 4. An example of the road network

Table 1 describes the description of link references and the distances across the same.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 6s
DOI: https://doi.org/10.17762/ijritcc.v11i6s.6898
***Article Received**: 25 March 2023 **Revised**: 02 May 2023 **Accepted**: 26 May 2023*

Table 1 Link Description

Link Ref No.	Link Endpoints	Distance (KM)
L1	MKSSS Cummins College - Rajaram Bridge	1.4
L2	Rajaram Bridge - Mhatre Bridge	2.3
L3	Mhatre Bridge - Alka Chowk	1.9
L4	Alka Chowk - Kasba Peth Kshetriya Karyalay	3.0
L5	Kasba Peth Kshetriya Karyalay - College of Engineering, Pune	2.4
L6	MKSSS Cummins College - Karve Statue	2.1
L7	Karve Statue - Erandwane	2.7
L8	Erandwane - Omkareshwar temple	3.2
L9	Omkareshwar temple - Kasba Peth Kshetriya Karyalay	1.7
L10	Rajaram Bridge - Nursing home colony	2.8
L11	Nursing home colony - Vishrambaug wada	2.4
L12	Vishram baug wada - Kasba Peth Kshetriya Karyalay	1.6
L13	Erandwane - FC College	2.6
L14	FC College - Kasba Peth Kshetriya Karyalay	2.7
L15	Karve Statue - Athavale Chowk	2.6
L16	Athavale Chowk - Ratna Memorial Hospital	2.4
L17	Ratna Memorial Hospital - Agriculture college flyover	2.5
L18	Agriculture college flyover - College of Engineering, Pune	3.8

For specific time instances of a day, the recommended path will be the shortest path containing origin, destination locations and set of link references to traverse across the path.

Algorithm 1, transforms the above map into a rectangular coordinate-based graph network as in *Figure 5*. Algorithm 1, describes the steps of modeling graph-based road networks for path recommendation.

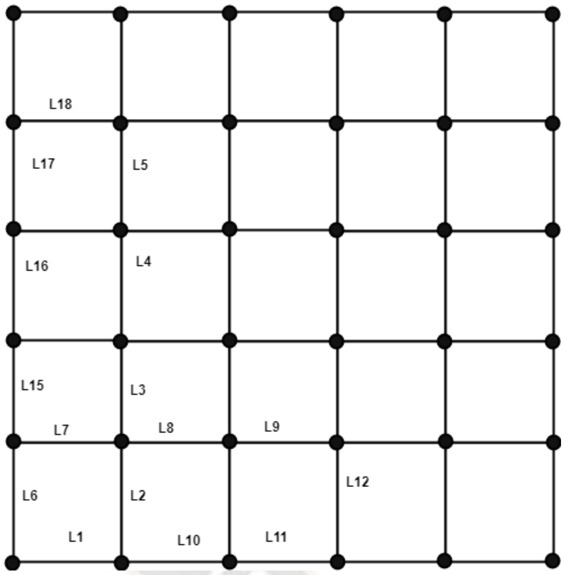


Figure 5. Graphical representation of Figure. 4 road network

C. Experimental metric

For evaluation of recommended path, traversed link references based prediction accuracy metric is used.

A vehicle passes over i^{th} link references, thereby predicting a possible route set $R = \{R_1, R_2, \ldots R_n\}$. So the definition of the route prediction accuracy is as follows

$$Pi = ({}^{n}\Sigma_{k=1}\, D(R_k, CR) \,/\, {}^{n}\Sigma_{i=1}\, Dist|R_t|\,) * 100\% \quad (5)$$

CR is the upcoming path, $D(R_k, CR)$ is the number of common link references between the path of set R - R_k and the entire upcoming path. Dist $|R_t|$ is the distance of link references across the path R_t.

An example road network is shown in *figure 4* and the corresponding link reference distance is in Table 1. For an upcoming path at the time period of the day, CR is from starting link reference L1 to end link reference L18. When the vehicle passes via the first link reference, the consideration sequence is denoted as L1 followed by at most ‘r’ link references, giving f(O, D, r) as a possible path from origin location O to destination location D. Duplication of link references across different time instances is $D(R_1, R_1) = 2$, as this training example has two starting link references. $D(R_2, R_1) = 2$, as two link references are getting duplicated across the upcoming paths, R_1 and R_2. Sum of distance for Number of link references across upcoming paths R_1 and R_2. as Dist $|R_1| = 5$ and Dist $|R_2| = 5$.

As the vehicle has traveled through the first link reference, the prediction accuracy:

$$\{D(R_1, R_1) + D(R_2, R_1)\}$$
$$P1 = . * 100\% \quad (6)$$

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 6s
DOI: https://doi.org/10.17762/ijritcc.v11i6s.6898
***Article Received**: 25 March 2023 **Revised**: 02 May 2023 **Accepted**: 26 May 2023*

$$\{Dist\ |R_1| + Dist\ |R_2|\}$$

$$= \{2+2\}/\{5+5\} * 100\% = 40\%$$

D. Experimental results

1) Training and test data

In the experiment, all of the collected link references are from google map and each of the details is included in a . XLS file. As per the description of the Graph-based representation of a road network model in figure 5, link references are represented by coordinate points. K-means clustering-based extended training examples are also considered. It has been observed that the distance between each coordinate point of divided link references and the corresponding clustering center is 0.280km and on average farthest distance is 0.760 km. To explore alternative paths the all-pair shortest path algorithm is used to extract all possible paths across origin and destination locations. The aggregate distance of selected link references across paths is the overall distance of the path.

Test Examples (i). It includes computed paths in terms of link references that have not appeared in the training examples. Simulate real-world path data to evaluate the prediction accuracy.

Test Examples (ii). It will be possible to estimate the prediction accuracy with respect to *Test Examples (i)* by justifying the large number of paths in the training examples.

For example, figure 6 demonstrates three different feasible paths between MKSSS Cummins college of Engineering (Origin location) and College of Engineering, Pune (Destination location). The highlighted blue path is the recommended shortest path. The path has passed through 5 selected link references including L_1, L_{10}, L_{11}, L_{12}, and L_5 as described in table 2.

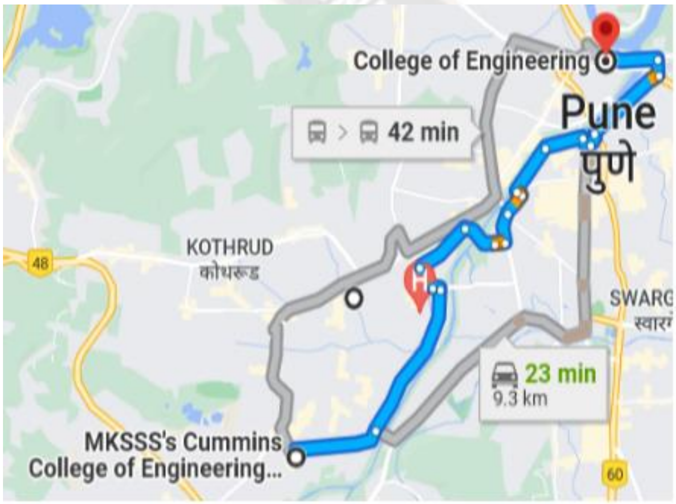


Figure 6. Pictorial representation of possible paths between OD pair

2) Prediction accuracy

Table 2, shows the various paths across origin-destination pairs for test example by link references traveled and corresponding distance values.

Table 2 Various paths and related distances traveled from MKSSS Cummins College, Pune to College of Engineering, Pune as an OD pair

Path across OD pair	Link references traveled	Distance (KM)
Path 1	L1 - L2 - L3 - L4 - L5	11.0
Path 2	L1 - L10 - L11 - L12 - L5	10.6
Path 3	L6 - L15 - L16 - L17 - L18	13.4
Path 4	L6 - L7 - L13 - L14 - L5	12.5
Path 5	L6 - L7 - L8 - L9 - L5	12.1

Figure 7, shows various paths starting from the origin: MKSSS Cummins College, Pune to the destination: College of Engineering, Pune as an OD pair. The shortest highlighted path passes through link references L1, L10, L11, L5 and finally, ends at the destination location.

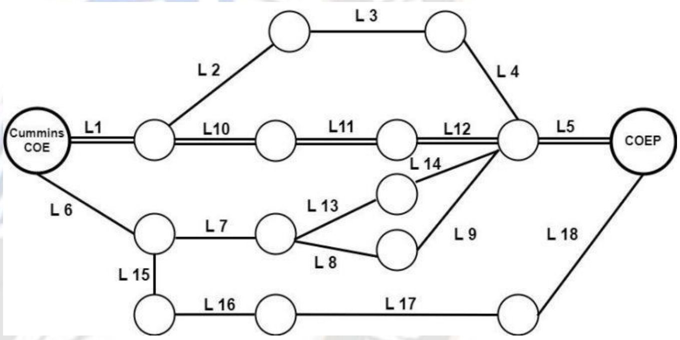


Figure 7. Shortest and other Paths between OD pair

Path prediction accuracy based on i^{th} link reference prediction is a perceptive decision criterion which is handy for examining the algorithm performance.Without knowing all possible paths across OD pairs, a recommendation system is not assured about the shortest distance path. The aim of algorithm 4 is the prediction of possible shortest link references from the current link reference, that is out of all possible link references selection of shortest one.

For the test example of table 2, at the origin location, little information is known about its path toward the destination location. After traversal of the first link reference prediction accuracy of our algorithm increases to 40% as out of 5 possible paths, first, the link reference is part of 2 possible paths. After the traversal of the second link reference, the performance of our algorithm further increases by 20% as out of 5 possible link predictions, the second link reference will lead to coverage of 39.62% of the overall shortest distance. Accurate prediction of third link reference will lead to coverage of 62.26 % of the overall shortest distance. Fourth link prediction will lead to

77.35% of overall shortest distance coverage and thereby ensuring that the shortest path recommendation is among all possible paths.

V. DISCUSSION

Shortest path computation is still a highly researched topic in the research field. This paper is one such attempt to solve location-oriented problems of traffic path recommendation and a better understanding of where the vehicle can move in the surrounding link references. In addition, it is also observed that during certain time periods of the day, traffic reaches its peak, and to avoid congested roads, alternative routes recommendation will be helpful. During the off-peak times of a day, the system will predict all possible paths and recommend a shortest-distance route across the origin-destination location. At the origin location, the correct link prediction rate is low as vehicles could select any one of the possible link references. Furthermore, on average, the prediction accuracy of link reference for repeated trips across OD pairs is higher than for unknown trips. Results of the experiment show that an accuracy of 77.73% is achieved by our model for shortest path recommendation.

VI. CONCLUSION AND FUTURE WORK

This paper presents a model for extracting all paths and recommendation of a shortest path for OD pair. We have considered only distance parameters for path recommendation and have not used the Euclidean approach as it is based on more than one parameter. We feel that future work can be carried out in two major directions. Firstly, to investigate traffic parameters such as vehicle speed, traffic flow, travel time, and congestion across different paths for route recommendation. Secondly, to analyze the application of different statistical methods to work with unknown coordinate points in the road network.

REFERENCES

[1] Mukherjee S. Statistical analysis of the road network of India. Pramana - J Phys 2012; 79(1): 483–491.

[2] P. Campigotto, C. Rudloff, M. Leodolter and D. Bauer. Personalized and Situation-Aware Multimodal Route Recommendations: The FAVOUR Algorithm. IEEE Transactions on ITS. 2017; 18(1): 92 - 102.

[3] Durga Toshniwal. Feature extraction from time series data. Journal of Computational Methods in Science and Engineering, 2009; (9): 99-110.

[4] Sreevani, C.A. Murthy. Bridging Feature Selection and Extraction: Compound Feature Generation. IEEE Transactions on Knowledge and Data Engineering, 2017; 29(4): 757-770.

[5] Jeng-Min Chiou, Han- Tsung Liou and Wan-Hui Chen. Modeling Time-Varying Variability and Reliability of Freeway Travel Time Using Functional Principal Component Analysis. IEEE Transactions on ITS. 2021; 22(1): 257-266.

[6] Yongjin Lee, Hyunjeong Jeon and Keemin Sohn. Predicting Short-Term Traffic Speed Using a Deep Neural Network to Accommodate Citywide Spatio-Temporal Correlations. IEEE Transactions on ITS. 2021; 22(3): 1435-1448.

[7] Zhongda Tian. Approach for Short-term Traffic Flow Prediction Based on Empirical Mode Decomposition and Combination Model Fusion. IEEE Transactions on ITSystems. 2021; 22(9): 5566-5576.

[8] Xiaomei Zhao, C. Wan, H. Sun, D. Xie and Z. Gao. Dynamic Rerouting Behaviour and its impact on Dynamic Traffic Patterns. IEEE Transactions on ITS, 2017;18(10):. 2763-2779.

[9] Maram Bani Younes, Azzedine Boukerche and Graciela Roman-Alonso. An intelligent path recommendation protocol (ICOD) for VANETs. Elsevier Computer Network Journal, 2014; 64(5):225 - 242.

[10] Shun Taguchi, Satoshi Koide and Takayoshi. Online Map Matching With Route Prediction. IEEE Transactions on ITS, 2019; 20(1): 338-347.

[11] Kan Guo, Yongli Hu, Zhen Qian, Hao Liu, Ke Zhang, Yanfeng Sun, Junbin Gao and Baocai Yin. Optimized Graph Convolution Recurrent Recurrent Neural Network for Traffic Prediction. IEEE Transactions on ITS. 2021; 22(2): 1138-1149.

[12] Pu Wang, Jiyu Lai, Zhiren Huang, Qian Tan, Tao Lin. Estimating Traffic Flow in a Large Road Network Based on Multi-Source Traffic Data. IEEE Transactions on ITS. 2021; 22(9): 5672-5683.

[13] Junping Zhang; Fei-Yue Wang; Kunfeng Wang; Wei-Hua Lin; Xin Xu; Cheng Chen. Data-Driven Intelligent Transportation Systems: A Survey. IEEE Transaction on ITS, 2011; 12(4): 1624-1639.

[14] V Eleni I. Vlahogianni,John C. Golias &Matthew G. Karlaftis. Short-term traffic forecasting: overview of objectives and methods. Transportation Reviews, 2004; 24 (5): 533–557.

[15] Jiexia Ye, Juanjuan Zhao, Kejiang Ye, Chengzhong Xu. How to build a graph-based Deep Learning Architecture in Traffic Domain : A Survey. IEEE Transactions on ITS.2022;23(5): 3904-3924.

[16] Zeyang Cheng, Jian Lu, Huajian Zhou, Yibin Zhang, Lin Zhang. Short-term Traffic flow Prediction: An Integrated Method of Econometrics and Hybrid Deep Learning. IEEE Transactions on ITS.2022; 23(6): 5231-5244.

[17] Changxi Ma, Guowen Dai, Jibiao Zhou. Short-term Traffic flow prediction for urban road sections based on time series analysis and LSTM BILSTM Method. IEEE Transactions on ITS.2022; 23(6): 5615-5624.

[18] Xun Zhou; Changle Li; Zhe Liu; Tom H. Luan; Zhifang Miao; Lina Zhu; Lei Xiong. See the near future: A short-term predictive methodology to traffic load in ITS. IEEE International Conference on Communications (ICC). 2017: 1-6.

[19] Zhiyong Cui, Kristian Henrickson, Ruimin Ke, and Yinhai Wang. Traffic Graph Convolutional Recurrent Neural Network: A Deep Learning Framework for Network-Scale Traffic Learning and Forecasting. IEEE Transactions on ITS. 2020; 21(11): 4883-4894.

[20] .Bowen Du, Xiao Hu, Leilei Sun, Junming Liu, Yanan Qiao and Weifeng Lv. Traffic Demand Prediction Based on Dynamic Transition Convolutional Neural Network. IEEE Transactions on ITS. 2021; 22(2): 1237-1247.

[21] Zhigao Zheng, Ali Kashif Bashir. Graph-Enabled Intelligent

Vehicular Network Data Processing. IEEE Transactions on ITS.2022;23(5): 4726-4735.

[22] Bing Yu, Haoteng Yin, Zhanxing Zhu. Spatio-Temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting," Proceeding of the Twenty-Seventh joint conference on AI (IJCAI) 2018:. 3634 – 3640.

[23] Xiaocai Zhang , Zhixun Zhao, Yi Zheng, Jinyan Li. Prediction of Taxi Destinations Using a Novel Data Embedding Method and Ensemble Learning. IEEE Transactions on ITS. 2020; 21(1): 68-78.

[24] Yuhang Xu, Wanyuan wang, Guangwei Xiong, Xiang Liu, Weiwei Wu. Network - flow based Efficient Vehicle Dispatch for City-Scale Ride Hailing Systems. IEEE Transactions on ITS.2022;23(6):5526-5538.

[25] Xiaoshu Xiang, Ye Tian, Xingyi Zhang, Jianhua Xiao, Yaochu Jin. A Pairwise Proximity Learning based Ant Colony Algorithm for dynamic Vehicle Routing Problems. IEEE Transactions on ITS. 2022;23(6): 5275-5286.

[26] Yaobin He, Fan Zhang , Ye Li, Jun Huang, Ling Yin, and Chengzhong Xu. Multiple Routes Recommendation System on Massive Taxi Trajectories. TSINGHUA SCI AND TECH.2016;21(5): 510-520.

[27] Xi Chen, Lei Peng, Minghong Zhang, and Wei Li. A Public Traffic Demand Forecast Method Based on Computational Experiments. IEEE Transactions on ITS. 2017; 18(4): 984-995.

[28] David Levinson, Shanjiang Zhu. A portfolio theory of route choice. Elsevier Transportation Research Part C. 2013; 35: 232 - 243.

[29] AndreasPella, Andreas Meingasta, OliverSchauera. Trends in Real-time Traffic Simulation. Transportation Research Procedia. 2017; (25) : 1477-1484

[30] Haosheng Huang , Dominik Bucher , Julian Kissling, Robert Weibel , and Martin Raubal. Multimodal Route Planning With Public Transport and Carpooling. IEEE Transactions on ITS. 2019; 20(9): 3513-3525.