Local Travellers Route Pattern-Based Application for an Accurate Route Recommendation

Sudha Chaturvedi,
Lingayas University
Department of Computer Science and Information Technology,

Tapsi Nagpal,
Lingayas University
Department of Computer Science and Information Technology,

Vishnu Shankar Tiwari
PES University
Department of Computer Science & Engineering (AI & ML),

Praveen Kumar S M
PES University
Department of Computer Science & Engineering (AI & ML),

Abstract

Route recommendation systems usage has grown significantly in recent days. It is used by individuals and travelers for their day-to-day travel planning activities. Apart from this, many other areas have applications of route recommendation systems, like route prediction, vehicle routing problem, grocery and food delivery applications, infrastructure planning, traffic congestion estimation, VANETS, and Grid computing for resource estimation. Traditional implementations of route recommendation systems make use of static road attributes to recommend routes for users. Static attributes are the shortest distance from source to destination, the time of the day when traveling is planned, etc. There are other attributes that are dynamic in nature, like road condition, lighting on the road, temporary diversions, and road blocks, which also impact travel experience. If these attributes are not incorporated as a factor for route planning, then it can lead to conditions like a planned route having bad traffic conditions, a dead end on the suggested route, and security issues, etc. But capturing these attributes is very hard in nature. It is observed that residents in the area know these conditions, and they don't necessarily follow the shortest path, but they follow the best path, which has good road conditions, including security issues, dead ends, etc. With the proliferation of advanced devices, such as GPS, mobile phones, PDAs, etc., user historical travel data is available in abundance. This article presents a route recommendation system that leverages the historical user travel pattern data combined with dynamic attributes to recommend the best route to users for better planning.
Keywords Route, Recommendation, Prediction, Open Street Map, Travel, Pattern

Introduction

Route recommendation systems usage by travelers, taxi drivers, food and grocery delivery, and online retail product delivery has seen a huge increase in recent times. Many other applications used for infrastructure planning, traffic estimation, resource estimation in ad-hoc computing like VANETS, route prediction, etc., have high usage of route recommendation applications. Users like individual travelers and taxi drivers try to optimize their commute based on the shortest path and minimum commute time, but at the same time, try to avoid unpleasant experiences like bad road conditions, poor lighting conditions, security issues, etc. Multiple times, it is observed that the route recommendation and route prediction systems suggest routes that have conditions that are not expected to be experienced in the recommended route. A sample route suggested by the automated route recommendation system, as shown in Figure 1a, has poor road conditions, as shown in Figure 1b, but local users follow a better alternative route for the same source and destination shown in Figure 1c.

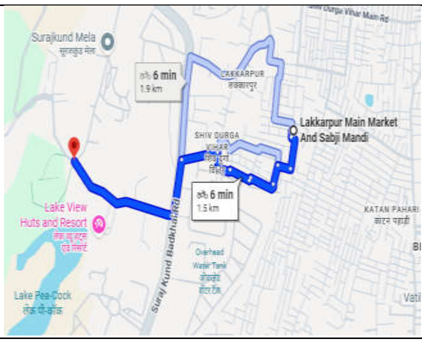

Figure 1a: Route suggested by application

Figure 1b: Suggested route has poor road conditions

Figure 1c: Local users follow a better alternative route

Another example of route recommendation suggested a route that has a dead end. This is often reported in the news media and newspapers, where travelers face issues, and then they travel back and take an alternative path. The route suggested by the route recommendation system is shown in Figure 2a, which has a dead end, and is shown in Figure 2b. The same destination route followed by a local traveler is shown in Figure 2c.

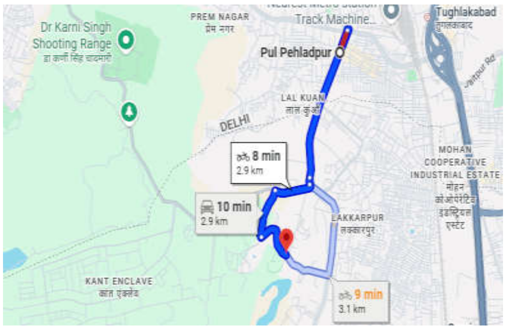

Figure 2a: Route suggested by application

Figure 2b: Suggested route has a dead end

Figure 2c: Local users follow a better alternative route

The route suggested by the route recommendation system, as in Figure 3a, has security concerns, and nobody is traveling on that road, as shown in Figure 3b. For the same destination, a better route followed by local travelers is shown in Figure 3c.

Figure 3a: Route suggested by application

Figure 3b: Suggested route has poor security conditions

Figure 3c: Local users follow a better alternative route

It is observed that local users, like residents and taxi drivers, are more aware of the road conditions and follow a much better route with better planning, but avoid road conditions like poor road quality, security, and poor lighting on the road (Lakshmi, A.V. et. al 2024). These users' travel patterns can be captured using location-capturing devices like GPS, cell phone PDAs. Over a period, a large volume of location data is accumulated over the server, which contains the user travel pattern behavior (Auwal et. al. 2023; Yinup et. al. 2024). This travel pattern behavior can be captured and fed into a prediction model and can be used to suggest better routes.

Background

Route recommendation is a very long researched area and is the process of suggesting the route to the end user based on optimization of criteria like shortest path, good road conditions, road with better security, no roadblocks, etc., for better user experience (Wu et. al. 2025). The criteria for optimization can be of two types: static and dynamic conditions. Examples of static conditions are – shortest path, good road conditions, better security, and examples of dynamic conditions are temporary roadblocks, excavation of roads for maintenance etc. The research ranges from simple graph-based algorithms (e.g., Dijkstra and Tree-based searches) to modern Artificial Intelligence and Machine Learning based algorithms. Many existing route recommendation systems are only based on static road conditions and don't leverage dynamic road conditions, resulting in poor road conditions (Jung et. al. 2019). Additionally, they depend on only current parameters like the shortest path and don't include the historical travel pattern of users. But it is noticed that local users know the conditions better, both static and dynamic conditions, and follow a better path for the same source and destination, which is better than suggested by route recommendation systems.

Traditional algorithms model the road network as a graph model, which consists of edges that model road segments and vertices that model important places and road intersections. Weights are assigned to edges of the road network, and search algorithms are run on the graph to compute the route (Cheng, X. 2021). These algorithms have limitations as they don't scale well, and they don't consider dynamic route conditions. Modern heuristic search algorithms (e.g., A*, LPA*) (Ashrafi et. al. 2025; Turno F. & Yatskiv 2023; Tiwari et. al. 2018) overcome the shortcomings of traditional models but still don't utilize historical travel patterns of users and are based only on static conditions. Route prediction was proposed by (Froehlich et. al. 2008), which is based on historical user travel patterns. The GPS coordinate logs are broken into smaller units called trips. Trips are then used to cluster them based on similarity score, and then the clustered trips are used to predict the route. These trips in this work are raw GPS coordinates and don't use a mapping to the road network, which leads to inaccuracies and clustering and affects route prediction. Historical travel pattern-based models, Prediction by partial match (PPM) (Tiwari et. al. 2018), reduce-based scalable Lempel-Ziv (LZ) (Arya et. al. 2024), Distributed Context Tree Weighting (CTW) (Tiwari et. al. 2018), and probabilistic generalized suffix tree (PGST) (Tiwari et. al. 2017) predict the route followed based on historical GPS data. Continuous logs for GPS traces are decomposed into smaller units called trips. These trips, composed of GPS coordinates, are mapped and matched to the road network to convert trips to a sequence of road network edges. Then these trips, composed of road network edges, are used to develop prediction models which can then be used for route prediction and route recommendation. These models are purely based on static road attributes from the historical location traces corpus and don't use dynamic road conditions like temporary roadblocks, temporary digging of roads, etc., which leads to the recommendation of infeasible routes. The proposed work uses both static and dynamic road conditions for route recommendation, which can predict better routes. The proposed system utilizes the historical travel pattern of the user to recommend a better path.

Route Recommendation Model Construction from Historical Travel Data

The route recommendation system proposed in this work makes use of two sets of data - digitized road network and GPS location traces. Historical location traces are mapped to digitized road networks, and then a prediction model is built that captures the travel pattern of users. This section deals with model building for prediction and the process to utilize the model for route recommendation.

Historical Travel Location Traces

Location traces are a continuous sequence of latitude/longitude coordinates captured at discrete intervals represented by$(x_{0,}y_{0,}t^0), (x_{1,}y_{1,}t^1), (x_{2,}y_{2,}t^2) \dots\dots\dots (x_{n,}y_{n,}t^n)$. Location data in the form of $(x_{i,}y_{i,}t^i)$, where x_i represents latitude and y_i represents longitude at time t^i**.** The continuous location traces are decomposed into smaller logical units that express the user's travel pattern. For example, a user travels from home to office and commutes back to home, which consists of two trips, whereas the raw GPS traces this whole traversed path will be one long sequence of location traces.

Decomposed trips are then mapped to digitized road network edges. The sequence of road network edges resultant from trip segmentation is done using a process known as map matching (Gupta, Ajay K & Shanker, Udai. 2022; Jinsoo et. al. 2022;). A prediction model is then developed using the trips as a sequence of road edges. In this work, the location traces used are from Microsoft Research's Geolife project (Schroedl et. al. 2004; Zheng Y. et. al.2009). The project captured data from various parts of the world over a long period for hundreds of users to capture their travel pattern behavior. A sample of raw GPS data is shown in Figure 4. At this stage, the trip is still a sequence of raw location traces, but in a shorter and logical form.

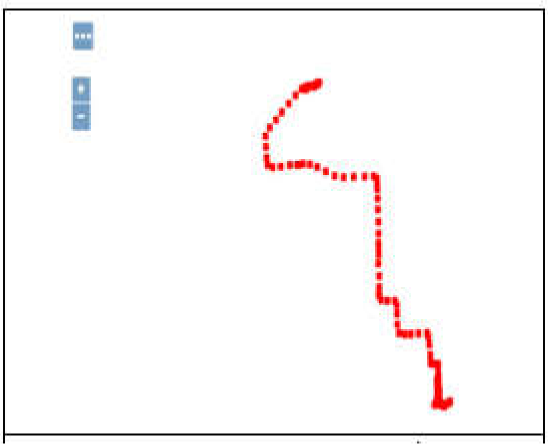

Figure 4. User trip in GPS coordinates

Digital Road Networks

A road graph G= (V_G, E_G) is digitally vectored form of actual road network where V_G is set of vertices representing road intersections, important places, landmarks etc and $E_G$is set of edges of road network which is an association between pair of vertices and representing actual structure of road network and is association between vertices. If it's possible to reach directly from one vertex to another, then there is an edge between them. It may be possible that there is no edge directly between the vertices, but through transitivity of edges, another vertex can be reachable, which is known as a path. In this work, the road network data is obtained from Open Street Map (OSM). OSM is an open-source platform that hosts various kinds of geographical data like road networks, water bodies, and domestic and international boundaries of the world (Hosseini et. al. 2025; R. Thottolil and U. Kumar 2021; Grinberger et. al. 2022).

Each road segment has certain attributes associated with it, which can be categorized as spatial and non-spatial attributes. Examples of non-spatial attributes are the name of the road, the length of the road segment, names of places that fall on the road segment, etc. Spatial attributes are a geometrical representation of a road segment. In minimal form, it's a vector of location represented by a latitude and longitude pair. The geometry of road segments can be multi-line, which means a sequential arrangement of lines representing the actual structure of the road. Traditional relational databases support required data types like text, numbers, etc. for the storage of non-spatial attributes, but don't support geometrical data types. For the storage of geometries like a point (latitude/longitude pair) representing the vertex of the graph, or a multiline representing the structure of the road network. Storage of spatial attributes (geometries) requires spatial data type-enabled databases. In this work, a spatial-enabled database, PostGIS, is used, which is an additional layer on top of the Postgres relational database to support geometrical data types. AS described in previous sections, the road network is captured from Open Street Map (OSM) and is processed and stored in a PostGIS database. Data rendering tools like GeoServer can connect to spatial databases and render the road network. An example of road network data captured from OSM is shown in Figure 5. Road network data captured from OSM and stored in spatial database is as shown in Figure 6.

Figure 5. Road network data from OSM

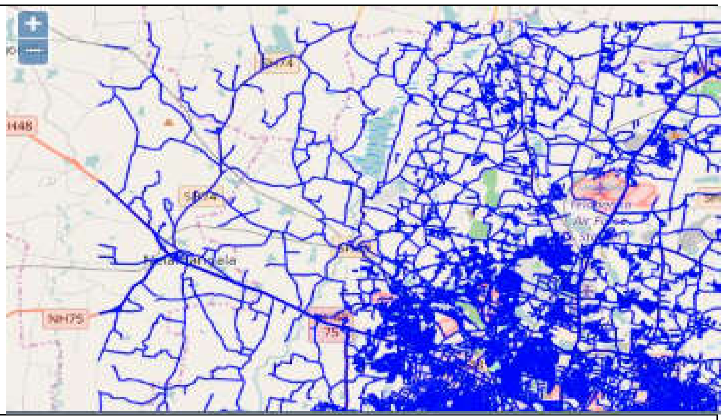

Figure 6. Road network data stored in PostGIS

Location Traces Mapping to Road Networks

For processing the travel pattern, the two datasets, road network and location traces data, should be mapped to create an association. The process of mapping location traces to road network data is known as map-matching (Huang et. al. 2024). Trip as a sequence of location triplet data $(x_{i,}y_{i,}t^{i})$ is mapped to road network edges. A trip as a sequence of location traces data map matched to road network edges is as shown in Figure 7.

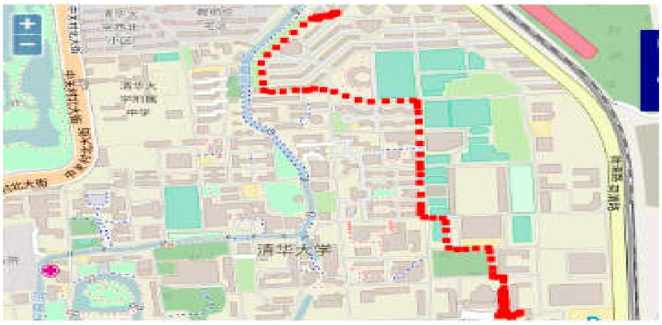

Figure 7. User trip mapped to road network

Recommendation Model Construction

Trips as a sequence of location traces converted to a sequence of road network edges using map matching is as described in the previous section.

$$T((x_{0,}y_{0,}t^{0}), (x_{1,}y_{1,}t^{1}), (x_{2,}y_{2,}t^{2}) \dots \dots \dots . (x_{n,}y_{n,}t^{n})) \rightarrow T(e_0, e_1, e_2, \dots \dots . e_n)$$

Where $(x_{i,}y_{i,}t^{i})$, the location is the coordinate of the user, and $(e_0, e_1, e_2, \dots \dots . e_n)$ are edges of the road network to which location coordinates are mapped (Jung et. al. 2019). Trips as a sequence of road network edges are used to construct the prediction model. Once the graph is stored in the database, the routing algorithm is applied to find the paths between the requested vertices representing the place names. For each trip source and destination, the path traversed is recorded. Given a set of edges $\sum = \{e_0, e_1, e_2, \dots \dots . e_k\}$ trip (T) is a contiguous sequence of edges $T = e_0, e_1, e_2, \dots \dots . e_k \ni \sum$. The paths from all the nodes to all the nodes are captured in the model, and edges are labeled with the frequency of the edges traveled in the past. There can also be scenarios where, in the past, the route was traversed by users, but due to some dynamic conditions like a roadblock, users are not traversing the path in the current scenario. Then the edge travel frequency is labelled with a negative weight. The process for model construction is described in Algorithm 1. An example graph-based model is shown in Figure 8.

Each vertex is labeled with vertex number $v_x \ni V_{G,}$ and edges are labeled with$e_y \ni E_G$. $f(e)$ is the frequency (thousands) of the edge e traveled in the historical corpus, and $d(e)$ is the distance in kilometers representing the length of the road segment.

Algorithm 1: *Route recommendation model construction*
Input: *User trips as a sequence of road network edges*
Output: *Graph-based route recommendation model*
Algorithm:

1. *Instantiate an empty graph* $\boldsymbol{G}\ (\boldsymbol{V}, \boldsymbol{E})$
2. *Initialize each edge* $\boldsymbol{e_i} \in \boldsymbol{E}$ *with weight* $-\infty$
3. *For each* $\boldsymbol{Trip_i} \in \boldsymbol{Trip_1}, \boldsymbol{Trip_2} \ldots.\boldsymbol{Trip_n}$
4. *For each sub-path* $\boldsymbol{s_j} \in \boldsymbol{Trip_i}$
5. *If edge weight* $\boldsymbol{f}(\boldsymbol{e_i}) = -\infty$ *then*
6. *for each edge* $\boldsymbol{e_i} \in \boldsymbol{s_j}$ *set weight of **edge*** $= \mathbf{1}$
7. ***else***
8. *for each edge* $\boldsymbol{e_i} \in \boldsymbol{s_j}$ *increase weight of **edge*** $\boldsymbol{f}(\boldsymbol{e_i}) = \boldsymbol{f}(\boldsymbol{e_i}) + \mathbf{1}$
9. *For each edge* $\boldsymbol{e_i} \in \boldsymbol{E}$ *where E is set of edges in graph* $\boldsymbol{G}$
10. *If edge* $\boldsymbol{e_i}$ *has a blocker, then set edge weight* $\boldsymbol{f}(\boldsymbol{e_i}) = -\infty$

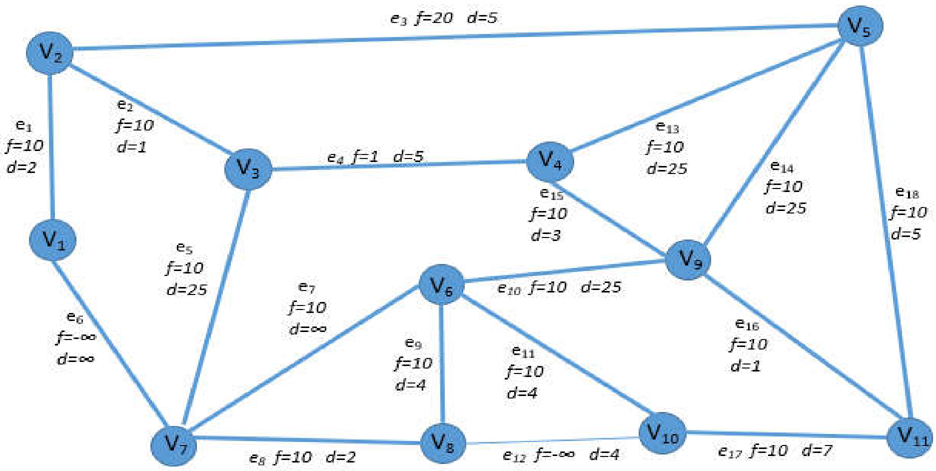

Figure 8: Graph network with edges labeled with historical travel frequency

Route Recommendation Application

Road conditions can be categorized into two kinds, namely static and dynamic. Scenarios shown in Figure 1, road has bad condition, and Figure 2, road has a dead end, are known as static conditions and don't change frequently. Other conditions can lead to bad experiences that change very frequently and for short periods. For example, road digging by municipal authorities, underground cables,clearing underground sewer lines, etc. The prediction model contains static and dynamic road conditions, which are then used to recommend the route for the user, given a source and destination pair. This model is based on user travel patterns and the current road segment condition, which appears in the recommended route and avoids roads with bad conditions. In this work, Floyd Warshall's algorithm is used as an all-pairs shortest path. The change applied in the traditional algorithm is used to find the shortest path based on distance. In this work, instead of finding the shortest path, an algorithm is applied to find the best path, which uses the weights on the edges as the frequency of the users who have traveled the path historically. The modified algorithm implemented is as represented in Algorithm 2.

Algorithm 2: All-pairs best path route selection
Input: Road network graph with frequency of edges traversed in the corpus
Output: Graph-based route recommendation model
Algorithm:

11. $def\ best\_path\_selection(vertices\_list)$:
12. $V\ =\ len(dist)$
13. $for\ k\ in\ range(V)$:
14. $for\ i\ in\ range(V)$:
15. $if(score[i][k]\ != \infty\ and\ score\ [k][j]! = \infty)$:
16. $score\ [i][j]\ =\ max(score\ [i][j], score\ [i][k]\ +\ score\ [k][j])$;

While querying for the recommended path from a source to a destination, the resultant model is traversed from the source node to the destination node with optimization on the frequency of the edges traversed in the past. Given this, when the model is traversed for the recommended route, then the edge with negative weight is excluded, and the next best route is recommended. Below are the scenarios for route recommendation with the assistance of historical travel patterns. The following are the cases representing the route recommended based on the model shown in Figure 8.

Scenario I: Route queried is with source vertex v_1 and target vertex $v_{11,}$ recommends traveling on edges $e_1 \rightarrow e_3 \rightarrow e_{18}$. The shortest path is $e_1 \rightarrow e_2 \rightarrow e_4 \rightarrow e_{15} \rightarrow e_{16}$ with a total distance of 12 kilometers, and the recommended path has a distance of 20 kilometers. The reason is that fewer people are following that path because of bad road conditions.

Scenario II: Route queried is with source vertex v_7 and target vertex v_{11} , which recommends traveling on edges $e_8 \rightarrow e_9 \rightarrow e_{11} \rightarrow e_7$ with a total distance of 17 kilometers, whereas the shortest path is $e_8 \rightarrow e_{12} \rightarrow e_{17}$ with a distance of 13 kilometers. The reason is road blockage between vertices v_8 and v_{10}.

Scenario III: Recommended path between vertices v_1 and v_7 is $e_1 \rightarrow e_2 \rightarrow e_{5,}$ as no traveling is observed on edge e6, and the well distance is not known.

Scenario IV: Route queried is with source vertex v_7 and target vertex $v_{5,}$ recommends traveling on edges $e_7 \rightarrow e_{10} \rightarrow$, even when the distance on edge e_7 is not known, but travelers are using that road, so in the recommended path, e_7 appears.

Conclusion

Existing models for route recommendation systems like Krum [1] are based on route recommendation, which captures only end-to-end trips, and intermediate routes are missing, which is overcome in the proposed work. Models like PGST [19], LZ [16], PPM [8], CTW [14] capture intermediate paths as well, but don't capture all the intermediate routes. The proposed model captures all pair paths and hence performs better in accuracy, but the number of sub-paths to be computed is larger in number and hence takes larger computation time, which is$O(V^3)$. Additionally, existing models only use static conditions and hence computation time is lesser, whereas the proposed model resolves the dynamic conditions as well and hence takes a longer computation time, but the accuracy obtained is better than existing models.

Future Work

This work addresses the issues in the existing methodology for route recommendation. The proposed model uses the historical travel pattern to leverage the knowledge hidden in the location traces patterns. The solution is not only end-to-end route recommendation but also models the intermediate sequences of paths in travel patterns. Also includes the dynamic road conditions to predict better paths. But this strategy needs to process a huge corpus of data because of intermediate path sequences and processing of dynamic attributes, and takes a long time to construct the model. The model construction can be extended to computing on a horizontally scalable platform. Data can be stored on distributed data storage like Hadoop, and computation can be extended to execute in distributed computation models like Map Reduce framework.

References

Lakshmi, A. V., Parthiban, A., & Joseph, K. S. (2024). An efficient safest route prediction-based route discovery mechanism for drivers using improved golden tortoise beetle optimizer. Journal of Experimental & Theoretical Artificial Intelligence, 1–23.

Auwal Sagir Muhammad, Longbiao Chen, Cheng Wang, Advancing safe mobility: A global analysis of research trends in safe route planning, Heliyon, Volume 9, Issue 12, 2023, e22755, ISSN 2405-8440

Yinuo Huang, Xin Jin, Miao Fan, Xunwei Yang, Fangliang Jiang, 2024, Personalized Route Recommendation Based on User Habits for Vehicle Navigation IDST '24: Proceedings of the 2024 International Conference on Intelligent Driving and Smart Transportation Pages 28 - 32

Gupta, Ajay K & Shanker, Udai. (2022). A Comprehensive Review of Map-Matching Techniques: Empirical Analysis, Taxonomy, and Emerging Research Trends. International Journal of Web Services Research. 19. 1-32. 10.4018/IJWSR.306243

Cheng, X. A travel route recommendation algorithm based on interest theme and distance matching. EURASIP J. Adv. Signal Process. 2021, 57 (2021). https://doi.org/10.1186/s13634-021-00759-x.

Ashrafi DAhmed SRahman MChowdhury SFaruqui M, (2025)Does navigation quality and safety quality motivate users' to recommend smart helmet technologies? The roles of trust and anthropomorphismThe TQM Journal10.1108/TQM-11-2024-0448Online publication date: 22-Apr-2025

Turno, Francesco Maria & Yatskiv, Irina. (2023). Graph-Based Approach for Personalized Travel Recommendations. Transport and Telecommunication Journal. 24. 423-433. 10.2478/ttj-2023-0033.

Tiwari, V.S., Arya, A. & Chaturvedi, S. Scalable prediction by partial match (PPM) and its application to route prediction. Appl Inform 5, 4 (2018).

Jinsoo Kim, Jae Hun Kim, Gunwoo Lee, GPS data-based mobility mode inference model using long-term recurrent convolutional networks, Transportation Research Part C: Emerging Technologies, Volume 135, 2022, 103523, ISSN 0968-090X, https://doi.org/10.1016/j.trc.2021.103523.

Hosseini, R., Tong, D., Lim, S., Sohn, G., & Gidófalvi, G. (2025). A framework for performance analysis of OpenStreetMap data in navigation applications: the case of a well-developed road network in Australia. Annals of GIS, 31(2), 233–250.

Wu, X., Zhang, Z. & Wan, W. Travel route recommendation with a trajectory learning model. Complex Intell. Syst. 11, 12 (2025). https://doi.org/10.1007/s40747-024-01611-z.

R. Thottolil and U. Kumar, "Assessment of Topological Pattern of Road Network: A Case Study Of Bangalore City," 2021 IEEE International India Geoscience and Remote Sensing Symposium (InGARSS), Ahmedabad, India, 2021, pp. 246-249, doi: 10.1109/InGARSS51564.2021.9792069.

Grinberger, A. Y., Minghini, M., Juhász, L., Yeboah, G., & Mooney, P. (2022). OSM Science—The Academic Study of the OpenStreetMap Project, Data, Contributors, Community, and Applications. ISPRS International Journal of Geo-Information, 11(4), 230. https://doi.org/10.3390/ijgi11040230

Tiwari, Vishnu & Arya, Arti. (2018). Distributed Context Tree Weighting (CTW) for route prediction. Open Geospatial Data, Software and Standards. 3. 10.1186/s40965-018-0052-9.

Schroedl, S., Wagstaff, K., Rogers, S. et al. Mining GPS Traces for Map Refinement. Data Min Knowl Disc 9, 59–87 (2004). https://doi.org/10.1023/B:DAMI.0000026904.74892.89

Arya, Arti & Chaturvedi, Sudha & Tiwari, Vishnu. (2024). Map reduce-based scalable Lempel-Ziv and application in route prediction. International Journal of Big Data Intelligence. 1. 1. 10.1504/IJBDI.2022.10047952.

Yu Zheng, Lizhu Zhang, Xing Xie, Wei-Ying Ma. Mining interesting locations and travel sequences from GPS trajectories. In Proceedings of International conference on World Wild Web (WWW 2009), Madrid Spain. ACM Press: 791-800.

Y. Huang, D. Wang, W. Xu, Z. Cai and F. Fu, "Accurate Map Matching Method for Mobile Phone Signaling Data Under Spatio-Temporal Uncertainty," in IEEE Transactions on Intelligent Transportation Systems, vol. 25, no. 2, pp. 1418-1429, Feb. 2024, doi: 10.1109/TITS.2023.3314631.

Tiwari, Vishnu & Arya, Arti. (2017), Horizontally scalable probabilistic generalized suffix tree (PGST) based route prediction using map data and GPS traces. Journal of Big Data. 4. 10.1186/s40537-017-0085-4.

J. Jung, S. Park, Y. Kim and S. Park, "Route Recommendation with Dynamic User Preference on Road Networks," 2019 IEEE International Conference on Big Data and Smart Computing (BigComp), Kyoto, Japan, 2019, pp. 1-7, doi: 10.1109/BIGCOMP.2019.8679379.

Froehlich, J. and Krumm, J., "Route Prediction from Trip Observations," SAE Technical Paper 2008-01-0201, 2008, https://doi.org/10.4271/2008-01-0201

Article History: Received xxxxxx Revised xxxxxx Accepted xxxxxx Published: xxxxxxx