Are We There Yet? A Grand Tour of ETA Queries

Areeg Mostafa
mosta041@umn.edu
University of Minnesota, USA

Mohamed Hemdan
hemda001@umn.edu
University of Minnesota, USA

Mohamed F. Mokbel
mokbel@umn.edu
University of Minnesota, USA

ABSTRACT

This tutorial provides a comprehensive overview of the research landscape of Estimated Time of Arrival (ETA) query processing. The tutorial categorizes the research in this area based on the underlying edge weights used to compute the ETA. This includes edge weights that are represented as either static numbers, time dependent numbers, probability distributions, or feature vectors. The tutorial also covers ETA queries that are edge agnostic, i.e., computing the ETA without knowing edge weights. The tutorial concludes by discussing ETA queries in industry while pointing out to research gaps and future research directions.

PVLDB Reference Format:
Areeg Mostafa, Mohamed Hemdan, and Mohamed F. Mokbel. Are We There Yet? A Grand Tour of ETA Queries. PVLDB, 19(12): 4872 - 4877, 2026.
doi:10.14778/3827998.3828142

1 INTRODUCTION

An Estimated Time of Arrival (ETA) query aims to estimate the time needed to travel from a given source point to a given destination point. ETA queries are among the most commonly used spatio-temporal queries worldwide. For example, one ride sharing company, Uber, receives half million ETA queries per second [20]. In terms of applications, ride sharing extensively relies on accurate ETA for their fare estimation [38, 71], supply chain industry heavily relies on accurate ETA for transparency and planning [39, 67], shipping and logistics need accurate ETA to enable their customers making informed decisions [7, 62]. Meanwhile, inaccurate ETA costs delivery companies in the US around 6 billion dollars annually [17, 22]. Computing an accurate ETA is very challenging to the extent that it is called the Uber's Billion Dollar problem [6]. From an algorithmic perspective, ETA is computed using any shortest path algorithm to find a route between the source and destination points that minimizes the sum of weights of all road segments along the route. Yet, from a data perspective, all these algorithms rely on having accurate road segment weights on the road network. If the weights are not available or inaccurate, then no matter how efficient and scalable these algorithms are, the result would be inaccurate. Unfortunately, such weights are only available to commercial applications as proprietary data. Hence, research efforts for ETA queries have mainly focused on inferring and representing edge weights as a means of accurate ETA query processing.

This work is supported by NSF grants IIS-2203553 and OAC-2118285.

Proceedings of the VLDB Endowment, Vol. 19, No. 12 ISSN 2150-8097.
doi:10.14778/3827998.3828142

This 90-minute tutorial aims to provide a comprehensive overview of the ETA query research landscape. Table 1 gives a taxonomy of the techniques covered in this tutorial divided into the following five categories, based on the underlying edge weights: (1) *Static edge weights*. This is the classical category of ETA queries, where each edge weight is represented by a single static number that could correspond to the time or distance needed to go through its two end points. (2) *Time dependent edge weights*. Each edge weight is represented by a set of numbers, where each number is per a time unit, e.g., edge weights in the morning/weekdays are different from evening/weekend. (3) *Probabilistic edge weights*. Edge weights per time unit are represented by a full probability distribution rather than a single number. (4) *Feature vector edge weights*. Edge weights are represented by embedded feature vectors that incorporate contextual signals, e.g., neighbouring edge weights, and historical traffic dynamics. (5) *Edge agnostic*. This category computes the ETA without the need for any edge weight by harvesting either historical raw trajectory data or origin-destination pairs.

Table 1 also shows the trade-offs of these five categories in terms of the following six criteria: (a) *ETA Accuracy*, which is the query result accuracy compared to the ground truth, (b) *Query latency*, which is the end-to-end query execution time. (c) *Offline Precomputation*, which indicates the amount of offline precomputations needed for each category, (d) *Bulk Processing*, which indicates if the category can process a set of ETA queries in bulk, (e) *Outlier Sensitivity*, which indicates if the category is sensitive to outliers, e.g., road accidents, extreme weather conditions, or construction, (f) *Historical Data*, which indicates how much each category relies on the availability of prior historical data before processing ETA queries accurately.

2 RELEVANCE

ETA queries have been a main topic of research in various database venues, with a growing trend over the last decade. Table 2 lists 67 papers that will be covered in this tutorial, along with their publication venues/year and the ETA category they belong to. 32 papers out of these 67 were published in main database research venues, VLDB, VLDB Journal, SIGMOD, and ICDE. 26 papers out of these 32 were published in the last decade. This shows that the topic of this tutorial, ETA queries, is highly relevant to VLDB audience.

3 TUTORIAL OUTLINE

Figure 1 gives the detailed outline and timing of this 90-minute tutorial. The tutorial is mainly composed of eight parts. The first part serves as a background and introduction. The next five parts correspond to the five categories in Table 1. The seventh part discusses industrial status for ETA queries. The last part concludes the tutorial by pointing out to research gaps and open problems. This section briefly discusses the content of each part.

Criteria	Static Weights	Time Dependent Weights	Probabilistic Weights	Feature Vector Weights	Edge-Agnostic (Trajectory)	(OD)
ETA Accuracy	Very Low	Moderate	High	Very High	High	Low
Query Latency	Very Low	Low	High	Moderate	Moderate	Very Low
Offline Precomputation	Low	Medium	High	Very High	High	Medium
Bulk Processing	Full	Partial	None	None	Partial	Full
Outlier Sensitivity	No	No	Partial	Partial	Partial	No
Historical Data	None	Low	High	Very High	High	Medium

Table 1: A Taxonomy of ETA Query Approaches

Year	PVLDB	VLDBJ	SIGMOD	ICDE	SIGSPATIAL	KDD	CIKM	Others
2025			[98]$^{\mathbf{F}}$	[83]$^{\mathbf{T}}$ [89]$^{\mathbf{F}}$ [95]$^{\mathbf{T}}$				
2024	[24]$^{\mathbf{P}}$							[57]$^{\mathbf{ED}\bigstar}$
2023	[69]$^{\mathbf{T}}$		[51]$^{\mathbf{ET}}$			[26]$^{\mathbf{F}\bigstar}$ [52]$^{\mathbf{F}\bigstar}$		[43]$^{\mathbf{F}}$
2022	[90]$^{\mathbf{F}}$	[96]$^{\mathbf{S}}$			[73]$^{\mathbf{ET}}$	[5]$^{\mathbf{F}\bigstar}$	[35]$^{\mathbf{F}\bigstar}$	[34]$^{\mathbf{ED}\bigstar}$ [40]$^{\mathbf{F}}$
2021				[36]$^{\mathbf{S}}$ [97]$^{\mathbf{T}}$	[42]$^{\mathbf{F}}$	[14]$^{\mathbf{F}\bigstar}$	[10]$^{\mathbf{F}\bigstar}$	[68]$^{\mathbf{F}\bigstar}$
2020	[63]$^{\mathbf{P}}$ [70]$^{\mathbf{ET}}$	[64]$^{\mathbf{P}}$	[80]$^{\mathbf{T}}$ [91]$^{\mathbf{F}}$	[33]$^{\mathbf{ED}}$ [47]$^{\mathbf{F}}$		[15]$^{\mathbf{F}\bigstar}$ [18]$^{\mathbf{F}\bigstar}$ [29]$^{\mathbf{F}\bigstar}$		[1]$^{\mathbf{ED}}$
2019	[77]$^{\mathbf{T}}$ [92]$^{\mathbf{T}}$			[31]$^{\mathbf{P}}$	[99]$^{\mathbf{F}}$		[19]$^{\mathbf{ET}}$ [50]$^{\mathbf{ET}}$ [82]$^{\mathbf{T}}$	[75]$^{\mathbf{ED}}$
2018		[32]$^{\mathbf{P}}$ [84]$^{\mathbf{P}}$		[25]$^{\mathbf{T}}$		[48]$^{\mathbf{F}}$ [79]$^{\mathbf{F}\bigstar}$		[74]$^{\mathbf{ET}}$
2017					[49]$^{\mathbf{T}}$ [56]$^{\mathbf{T}}$ [59]$^{\mathbf{T}}$			[44]$^{\mathbf{ED}\bigstar}$
2016	[8]$^{\mathbf{P}}$			[30]$^{\mathbf{T}}$				
<2016	[85]$^{\mathbf{P}}$ [81]$^{\mathbf{S}}$ [23]$^{\mathbf{T}}$	[87]$^{\mathbf{P}}$ [41]$^{\mathbf{T}}$		[86]$^{\mathbf{P}}$	[9]$^{\mathbf{T}}$ [2]$^{\mathbf{P}}$	[58]$^{\mathbf{T}}$ [78]$^{\mathbf{T}}$		[60]$^{\mathbf{T}}$ [21]$^{\mathbf{T}}$

S Static Weights **T** Time Dependent Weights **P** Probabilistic Weights **F** Feature Vector Weights
ET Edge-Agnostic (Trajectory) **ED** Edge-Agnostic (OD) ★ Industry Paper

Table 2: Timeline and Venue of Publications Covered in This Tutorial.

Part 1: The ETA Query (10 minutes)
- Introduction of the ETA query problem
- Overview of the five approaches to the ETA query

Part 2: Static Edge Weights (10 minutes)
- Classical approaches to static routing
- Modern approaches to static routing utilizing machine learning
- Open-source applications of static routing

Part 3: Time Dependent Weights (15 minutes)
- Introduction of time dependent edge weights
- Time dependent ETA methods

Part 4: Probabilistic Weights (10 minutes)
- Introduction of probabilistic edge weights
- Series of Stochastic Routing

Part 5: Feature Vector Weights (15 minutes)
- Introduction of edge weight as a handcrafted or learned embedding
- Handcrafted embedding approaches
- Learned embedding approaches

Part 6: Edge Agnostic ETA (15 minutes)
- Introduction of edge agnostic ETA prediction
- Full trajectory based techniques
- OD based techniques

Part 7: ETA in Industry (10 minutes)
- Introduction to major pioneers ETA advances
- Map services approaches (Google/Baidu)
- Ride sharing and delivery approaches (DiDi/Uber/Lyft/-Doordash)

Part 8: Summary and Open Problems (5 minutes)
- Open problems and research gaps
- Summary and conclusion

Figure 1: The 90-minute Tutorial Outline.

3.1 Part 1: The ETA Query

This part serves as the introduction of this tutorial. It starts by presenting a formal definition of ETA queries: Given: (a) a road network represented by a directed graph G=(V,E), where V is a set of vertices representing road intersections and E is a set of weighted directed edges representing road segments, (b) an origin vertex $O \in V$, (c) a destination vertex $D \in V$, and (d) a departure time t, find the Estimated Time of Arrival (ETA) at destination D. We then show the critical role of ETA queries in a wide range of widely used daily-life applications. We finally explain Table 1 in details to introduce the five categories of ETA queries along with their trade-offs in terms of various performance measures.

3.2 Part2: Static Edge Weights ETA

This part of the tutorial covers classical ETA queries, processed as shortest/fastest path queries when edge weights are represented by a single static number that could correspond to the time or distance needed to go through the two edge end points. As pointed out in the second column of Table 1, techniques in this category suffer from: (a) a very low accuracy as single static edge weights do not capture the dynamic variations of edge weights over time. Hence, the computed ETA for any trip would only depend on the origin and destination nodes, regardless of the trip start time, which makes the ETA of low accuracy, and (b) outlier conditions are not supported, as weights are static and do not take into account accidental conditions of congested traffic, weather, or construction. Meanwhile, on the positive side, techniques in this category: (a) are very efficient in terms of query latency due to decades of research in scaling up their execution, (b) have a very low offline computational overhead, which is basically about building index structures and precomputing some of the main routes that can partially serve incoming shortest path queries, (c) can easily be processed in bulk as an all-pairs shortest path query problem, and (d) do not need any prior historical data covering the query areas. Techniques that will be discussed in this part include the classical shortest-path algorithms and speedup techniques [3, 81], machine learning based algorithms [36, 96], and the widely-used open-source routing engines [27, 61, 72].

3.3 Part 3: Time Dependent Edge Weights

Motivated by the very low accuracy of ETA queries over static edge weights, it becomes apparent that there is a need to a more accurate representation of edge weights that reflects the fact that edge weights vary over time based on traffic conditions. Time dependent edge weights was the first step to go beyond static edge weights by presenting each weight as a list of pairs (T,w), where T is a certain time period and w is the edge weight during the time period T. Various granularities may be used for time dependent edge weights. For example, a high granularity may include 168 pairs where T is each hour of the week. A lower granularity may include time periods as early morning, daytime, rush hours, weekend, etc. This part of the tutorial covers ETA queries for time dependent edge weights. As pointed out in the third column of Table 1, techniques in this category have moderate accuracy, which is much better than the case of static edge weights, mainly due to the fact that the ETA query processing takes the trip start time into account by using the edge weights that correspond to the start time. Naturally, this gives little overhead for query latency. Similar to static edge weights, techniques in this category are not sensitive to outlier conditions. Meanwhile, unlike static edge weights, techniques in this category may rely on sparse historical data and perform some pre computations to infer the time dependent edge weights in case such weights are not available. Also, bulk processing can only be partially supported as time dependent matrices must be computed per departure time unit. Techniques that will be discussed in this part of the tutorial cover the theoretical foundations of time dependent shortest paths [21, 60], time dependent shortest path algorithms and indexing [9, 77, 80, 92], and edge weight inference techniques that are based on either rules and statistics [25, 41, 49, 56, 59], machine learning [23, 30, 58, 78, 83], or deep learning [69, 82, 95, 97].

3.4 Part 4: Probabilistic Edge Weights

Motivated by the apparent higher accuracy of time dependent edge weights over static edge weights, more efforts were dedicated in coming up with even more accurate representation of edge weights than the time dependent representation. Probabilistic edge weights is a step on that direction where the scalar number that represents the edge weight in each time period is replaced by a probability distribution function. This can add confidence interval and best/average/worst case scenarios for the ETA result. In addition, probabilistic edge weights enables a whole set of variations over the classical ETA queries, including probabilistic queries like: which route/ETA gives the best chance of arriving before a given deadline, which route has the most predictable ETA, and which route/ETA has the best balance between early arrival time and the risk of ETA unpredictability. As pointed out in the fourth column of Table 1, this category gives higher accuracy than time dependent edge weights, mainly due to adding the statistics and confidence on top of the ETA result. Moreover, this category is more sensitive to outliers as the probability of such outlier events is already taken into account within the probability distribution function of the edge weights.

On the other side, ETA computations over distributions are more complex, raising query latency and preventing bulk processing. Finally, accurately inferring the probability distribution weights for each edge require having access to large amounts of historical data and significant offline precomputations. This reliance on historical data makes the category less attractive to academia, government, and small and medium enterprises, but more attractive to large industry with access to such data, whether through operating-system phone tracking, e.g., Apple and Google Maps, or app usage, e.g., ride sharing and delivery applications. Techniques that will be discussed in this part of the tutorial cover the foundational travel cost model behind stochastic path finding [85], stochastic routing and probabilistic queries with their indexing [24, 32, 63, 64, 86], and the inference of travel-time distributions [2, 8, 31, 84, 87].

3.5 Part 5: Feature Vector Edge Weights

Motivated by the tremendous success of the idea of embedded feature vector representation in various domains, including natural language processing, computer vision, and large language models, this category represents each edge weight by a dense feature vector. Edge feature vectors can be either handcrafted or learned from road attributes and historical trajectory data. In either case, the feature vector encodes rich segment-level information that aims to boost the accuracy of ETA queries by knowing various aspects and factors that impact the edge weights. In the learned case, the feature vectors automatically discover spatial dependencies across non-adjacent segments without manual feature engineering. As pointed out in the fifth column of Table 1, feature vector edge weights outperform probabilistic edge weights in terms of much higher ETA accuracy with less query latency as it is more efficient to compute vector aggregation than probabilistic distribution propagation. Meanwhile, similar to probabilistic edges, feature vector edges are sensitive to outliers, but they do not support bulk processing. On the negative side, it needs to have access to way more historical data and enough computing resources to support the required very high offline pre-computations. These negative aspects make this category more favorable to large industry, who have proprietary access to such historical data and computing resources that can be employed to provide the very high ETA query accuracy. Techniques that will be discussed in this part of the tutorial cover handcrafted feature vector techniques [26, 52, 68, 79, 98], handcrafted feature vector with graph neural networks [5, 10, 14, 15, 18, 29, 35, 42, 43, 89, 90], and learned feature vector embeddings [40, 47, 48, 91, 99]. We will also cover production systems for ETA queries at Google Maps [10], Baidu Maps [14, 15, 35], and DiDi [5, 18, 26, 29, 52, 68, 79] that employ feature vector edge weights.

3.6 Part 6: Edge Agnostic ETA

Unlike all previous categories, techniques in this category neither have access to nor aim to infer any edge weights, and hence we call it an Edge-Agnostic category. Edge-agnostic ETA techniques bypass road segments entirely and directly estimate time of arrival from either historical trajectory data or historical Origin-Destination (OD) data. This makes this category applicable to cases where the underlying road network either not available or has some erroneous/missing roads. Trajectory-based edge-agnostic ETA techniques are analogous to feature vector edge weights techniques as they both apply representation learning to historical trajectories to encode spatiotemporal patterns. The key difference is that feature vector techniques learn an embedding per road segment while edge agnostic techniques learn an embedding per trajectory. Then, a trip arrival time between source and destination points is estimated based on the learned trajectory representations that are similar to the requested trip. As pointed out in the sixth column of Table 1, compared to feature vector techniques, this category needs less data and less offline computations, yet it results in lower accuracy. Also, it has similar query latency and outlier sensitivity to feature vector techniques, though it can partially support bulk processing.

Meanwhile, OD-based edge-agnostic ETA techniques are similar to the trajectory-based ones except that they apply representation learning for historical trips that only have OD information rather than full trajectories, which may not be available. As pointed out in the last column of Table 1, this makes this category less dependent on historical data, as it is much easier to have access to OD data than full trajectories, though this would negatively impact the ETA accuracy and outlier sensitivity. Query latency is fairly low, while bulk processing is fully supported since any OD pair can be queried independently, enabling city-scale OD matrix computation. Techniques that will be discussed in this part of the tutorial cover trajectory-based techniques that learn from either, full raw trajectories [50, 74] or decomposed trajectories over grid cells or graphs [19, 51, 70, 73], and OD-based methods [1, 33, 34, 44, 57, 75]. We will also cover production systems for ETA queries at Uber [34], Lyft [57], and DiDi [44] that employ edge agnostic OD techniques.

3.7 Part 7: ETA in Industry

This part of the tutorial covers ETA query processing techniques in major industry, including Baidu Maps, Didi, DoorDash, Google Maps, Lyft, and Uber, which are the techniques marked by ★ in Table 2. In particular, at Baidu Maps, deployed ETA techniques have gone from an initial model that captures spatial and temporal traffic patterns [15], to additionally accounting for driving behavior [14], and finally to modeling traffic congestion spreads from one part of the road network to another [35]. At DiDi, deployed ETA techniques started by an initial deep learning model [79], which was further improved by incorporating richer road network representations learned by a separate model [29], to a faster inference system [18], and finally to a model that continuously updates itself as new data arrives [26]. Meanwhile, DiDi researchers have also explored ways to speed up inference by replacing the recurrent component with faster attention-based alternatives [68], richer ways of representing a trip by jointly modeling road segments, links and intersections together [5], and employing probabilistic edge weights [52]. Early DiDi-affiliated research also investigated neural network approaches to travel time estimation before these production systems were developed [44]. DoorDash employs the probabilistic paradigm where downstream systems require a full distribution over travel time rather than a single point estimate [93]. Google Maps uses a graph neural network to learn how traffic propagates and predict ETA along a route [10]. Uber takes the ETA produced by its routing engine and employs a learned model to correct it, preserving the routing engine's query latency while adding learned accuracy [34, 71, 76]. Lyft uses custom loss functions to better handle prediction errors [13], and adds a reliability scoring layer that estimates how likely a given ETA is to be accurate [57].

3.8 Part 8: Summary and Open Problems

Guided by the taxonomy of Table 1, the tutorial concludes by pointing out to several research gaps, open problems, and future directions in the research landscape of ETA queries, which will help in guiding the future research agenda for the VLDB community. This includes the dataset gap, where even though there are more publicly released datasets, they are only released in very coarse resolutions, the handling of outlier events gap that are not well supported yet, along with the growing gap between industry and academia as production commercial systems are trained on billions of proprietary trips that are not available to academia.

4 TARGET AUDIENCE AND BACKGROUND

This tutorial targets researchers, developers, and practitioners, who are interested in understanding the internals of ETA queries and how it relates to the broad areas of spatial data management, graph algorithms, probabilistic modelling, and deep learning. The tutorial will have enough introductory material to ensure that the audience gets the necessary background needed to understand its contents, hence no prior knowledge is required to understand the techniques and approaches presented in this tutorial. The tutorial will be very beneficial for graduate students as it will help them in identifying various challenges for PhD topics. Practitioners will get to know the state-of-the-art techniques and approaches for ETA queries. Finally, the tutorial will act as an invitation to the data management community to invest more efforts in the area of ETA queries.

5 RELATED RECENT TUTORIALS

Tutorials in the DB Community. Over the last four years, there were eight tutorials in VLDB, SIGMOD, ICDE addressing various topics in spatial and spatio-temporal queries, including spatial data quality [45], large-scale geospatial analytics [4], responsible data sharing [16], spatiotemporal time series [88], spatial query optimization [94], large language models for spatial queries [28, 37], and path queries on graph [46]. None of these tutorials addressed the important topic of ETA query processing. The closest to our tutorial would be [46], which discussed various forms of shortest path queries, yet, it has only focused on the case of static edge weights and did not address ETA queries for non-static edges.

Tutorials from the Presenters. The third presenter, Mohamed Mokbel, has presented 22 tutorials over the last two decades. Nine of these were presented at VLDB, SIGMOD, ICDE [11, 12, 28, 37, 53–55, 65, 66]. All of these nine tutorials were co-presented with PhD students, and none is related to this tutorial proposal. Two of these tutorials were co-presented with the second author [28, 37].

6 BIOGRAPHICAL SKETCHES

Areeg Mostafa is a PhD student at University of Minnesota. She has a masters degree in Artificial Intelligence and Data Science from the University of Hull with Distinction and a bachelors degree in Computer Engineering from the American University in Cairo with Highest Honors. Her research interests include data management and systems including spatio-temporal data.

Mohamed Hemdan is a PhD student at University of Minnesota. He has a bachelors degree in Computer Engineering from the American University in Cairo with Highest Honors, with internships at Microsoft and CERN. His research interests include spatial data management and spatial data science.

Mohamed F. Mokbel is a Distinguished McKnight University Professor at University of Minnesota. He is a recipient of the VLDB 10-Year Best Paper Award for his work on location privacy, the IEEE ICDE 10-Year Influential Paper Award for his work on big spatial data, and the ACM SIGSPATIAL 10-Year Impact Award for his work on location-based social networks. He is the Editor-in-Chief for ACM Transactions on Spatial algorithms and Systems and PC Co-Chair for VLDB 2028. Prior service includes elected Chair of ACM SIGSPATIAL and PC Vice-Chair for SIGMOD 2018. Mohamed is an ACM Fellow and IEEE Fellow.

REFERENCES

[1] S. Abbar, R. Stanojevic, and M. Mokbel. STAD: Spatio-Temporal Adjustment of Traffic-Oblivious Travel-Time Estimation. In *IEEE MDM*, 2020.

[2] M. Asghari, T. Emrich, U. Demiryurek, and C. Shahabi. Probabilistic Estimation of Link Travel Times in Dynamic Road Networks. In *ACM SIGSPATIAL*, 2015.

[3] H. Bast, D. Delling, A. Goldberg, M. Müller-Hannemann, T. Pajor, P. Sanders, D. Wagner, and R. F. Werneck. *Route Planning in Transportation Networks*. Springer, 2016.

[4] T. N. Chan, L. H. U, B. Choi, J. Xu, and R. Cheng. Large-scale Geospatial Analytics: Problems, Challenges, and Opportunities (Tutorial). In *SIGMOD*, 2023.

[5] Z. Chen, X. Xiao, Y.-J. Gong, J. Fang, N. Ma, H. Chai, and Z. Cao. Interpreting Trajectories from Multiple Views: A Hierarchical Self-Attention Network for Estimating the Time of Arrival. In *SIGKDD*, 2022.

[6] Uber's Billion Dollar Problem: Predicting ETAs Reliably And Quickly. The Code Compass, 2024. Retrieved June 29, 2026 from https://codecompass00.substack.com/p/uber-billion-dollar-problem-predicting-eta.

[7] Connecta Network. Importance of ETA, ETD, ATD, and ATA in Shipping and Logistics. Retrieved June 29, 2026 from https://www.connecta-network.com/importance-of-eta-etd-atd-ata-in-shipping-and-logistics/.

[8] J. Dai, B. Yang, C. Guo, C. S. Jensen, and J. Hu. Path Cost Distribution Estimation using Trajectory Data. *PVLDB*, 10(3), 2016.

[9] U. Demiryurek, F. Banaei-Kashani, and C. Shahabi. A Case for Time-Dependent Shortest Path Computation in Spatial Networks. In *ACM SIGSPATIAL*, 2010.

[10] A. Derrow-Pinion, J. She, D. Wong, O. Lange, T. Hester, L. Perez, M. Nunkesser, S. Lee, X. Guo, B. Wiltshire, P. W. Battaglia, V. Gupta, A. Li, Z. Xu, A. Sanchez-Gonzalez, Y. Li, and P. Velickovic. ETA Prediction with Graph Neural Networks in Google Maps. In *CIKM*, 2021.

[11] A. Eldawy and M. F. Mokbel. The Era of Big Spatial Data (Tutorial). In *ICDE*, 2016.

[12] A. Eldawy and M. F. Mokbel. The Era of Big Spatial Data (Tutorial). In *PVLDB*, 2017.

[13] M. Elmasri. Beyond L2 Loss — How we experiment with loss functions at Lyft. Lyft, Engineering, 2019. Retrieved June 29, 2026 from https://eng.lyft.com/beyond-l2-loss-how-we-experiment-with-loss-functions-at-lyft-51f9303f5d2d.

[14] X. Fang, J. Huang, F. Wang, L. Liu, Y. Sun, and H. Wang. SSML: Self-Supervised Meta-Learner for En Route Travel Time Estimation at Baidu Maps. In *SIGKDD*, 2021.

[15] X. Fang, J. Huang, F. Wang, L. Zeng, H. Liang, and H. Wang. ConSTGAT: Contextual Spatial-Temporal Graph Attention Network for Travel Time Estimation at Baidu Maps. In *SIGKDD*, 2020.

[16] R. C. Fernandez and A. Nandi. Responsible Sharing of Spatiotemporal Data (Tutorial). In *SIGMOD*, 2024.

[17] A. Frost. Traffic Technology Today. Poor Maps Costing Delivery Companies US $6bn Annually. Traffic Technology Today, 2020. Retrieved June 29, 2026 from https://www.traffictechnologytoday.com/news/mapping/poor-maps-costing-delivery-companies-us6bn-annually.html.

[18] K. Fu, F. Meng, J. Ye, and Z. Wang. CompactETA: A Fast Inference System for Travel Time Prediction. In *SIGKDD*, 2020.

[19] T.-Y. Fu and W.-C. Lee. DeepIST: Deep Image-based Spatio-Temporal Network for Travel Time Estimation. In *CIKM*, 2019.

[20] How Uber Computes ETA at Half a Million Requests per Second. Geeks for Geeks, 2025. Retrieved June 29, 2026 from https://www.geeksforgeeks.org/system-design/how-uber-computes-eta-at-half-a-million-requests-per-second/.

[21] B. George and S. Shekhar. *Time-Aggregated Graphs for Modeling Spatio-temporal Networks*. Springer, 2008.

[22] F. Glans. Unveiling the Mapping in Logistics Report: The Impact of Broken Maps on Last-Mile Deliveries. Mapillary, 2020. Retrieved June 29, 2026 from https://blog.mapillary.com/update/2020/02/14/mapping-in-logistics.html.

[23] H. Gonzalez, J. Han, X. Li, M. Myslinska, and J. P. Sondag. Adaptive Fastest Path Computation on a Road Network: A Traffic Mining Approach. In *PVLDB*, 2007.

[24] C. Guo, R. Xu, B. Yang, Y. Yuan, T. Kieu, Y. Zhao, and C. S. Jensen. Efficient Stochastic Routing in Path-centric Uncertain Road Networks. *PVLDB*, 17(11), 2024.

[25] C. Guo, B. Yang, J. Hu, and C. Jensen. Learning to Route with Sparse Trajectory Sets. In *ICDE*, 2018.

[26] J. Han, H. Liu, S. Liu, X. Chen, N. Tan, H. Chai, and H. Xiong. iETA: A Robust and Scalable Incremental Learning Framework for Time-of-Arrival Estimation. In *SIGKDD*, 2023.

[27] HeiGIT gGmbH. openrouteservice: Open Source Routing Service. Retrieved June 29, 2026 from https://openrouteservice.org/.

[28] M. Hemdan, Y. Hussein, and M. F. Mokbel. Large Language Models for Spatial Analysis Queries (Tutorial). In *ICDE*, 2025.

[29] H. Hong, Y. Lin, X. Yang, Z. Li, K. Fu, Z. Wang, X. Qie, and J. Ye. HetETA: Heterogeneous Information Network Embedding for Estimating Time of Arrival. In *SIGKDD*, 2020.

[30] H. Hu, G. Li, Z. Bao, Y. Cui, and J. Feng. Crowdsourcing-based Real-time Urban Traffic Speed Estimation: From Trends to Speeds. In *ICDE*, 2016.

[31] J. Hu, C. Guo, B. Yang, and C. S. Jensen. Stochastic Weight Completion for Road Networks using Graph Convolutional Networks. In *ICDE*, 2019.

[32] J. Hu, B. Yang, C. Guo, and C. S. Jensen. Risk-aware Path Selection with Time-varying, Uncertain Travel Costs: A Time Series Approach. *VLDB J.*, 27(2), 2018.

[33] J. Hu, B. Yang, C. Guo, C. S. Jensen, and H. Xiong. Stochastic Origin-destination Matrix Forecasting using Dual-stage Graph Convolutional, Recurrent Neural Networks. In *ICDE*, 2020.

[34] X. Hu, T. Binaykiya, E. Frank, and O. Cirit. DeeprETA: An ETA Post-processing System at Scale. arXiv:2206.02127, 2022.

[35] J. Huang, Z. Huang, X. Fang, S. Feng, X. Chen, J. Liu, H. Yuan, and H. Wang. DuETA: Traffic Congestion Propagation Pattern Modeling via Efficient Graph Learning for ETA Prediction at Baidu Maps. In *CIKM*, 2022.

[36] S. Huang, Y. Wang, T. Zhao, and G. Li. A Learning-based Method for Computing Shortest Path Distances on Road Networks. In *ICDE*, 2021.

[37] Y. Hussein, M. Hemdan, and M. F. Mokbel. Large Language Models for Spatial Analysis Queries (Tutorial). *PVLDB*, 2025.

[38] H. Ismail. Distance and ETA Calculation Caching: Optimizing Real-Time Performance in Ride-Hailing Apps. Medium, 2024. Retrieved June 29, 2026 from https://medium.com/@helal.hamed/distance-and-eta-calculation-caching-optimizing-real-time-performance-in-ride-hailing-apps-779da17529fc.

[39] P. Jakovljevic. AI-Powered Supply Chains by Oracle—An ETA Prediction Use Case. Technology Evaluation Centers (TEC), 2025. Retrieved June 29, 2026 from https://www3.technologyevaluation.com/publications/oracle-ai-supply-chains-eta-prediction-63253.

[40] T. S. Jepsen, C. S. Jensen, and T. D. Nielsen. UniTE—The Best of Both Worlds: Unifying Function-fitting and Aggregation-based Approaches to Travel Time and Travel Speed Estimation. *TSAS*, 8(4), 2022.

[41] H. Jeung, M. L. Yiu, X. Zhou, and C. S. Jensen. Path Prediction and Predictive Range Querying in Road Network Databases. *VLDB J.*, 19(4), 2010.

[42] G. Jin, H. Yan, F. Li, Y. Li, and J. Huang. Hierarchical Neural Architecture Search for Travel Time Estimation. In *ACM SIGSPATIAL*, 2021.

[43] G. Jin, H. Yan, F. Li, Y. Li, and J. Huang. Dual Graph Convolution Architecture Search for Travel Time Estimation. *ACM Transactions on Intelligent Systems and Technology, TIST*, 14(4), 2023.

[44] I. Jindal, T. Z. Qin, X. Chen, M. Nokleby, and J. Ye. A Unified Neural Network Approach for Estimating Travel Time and Distance for a Taxi Trip. arXiv:1710.04350, 2017.

[45] H. Li, B. Tang, H. Lu, M. A. Cheema, and C. S. Jensen. Spatial Data Quality in the IoT Era: Management and Exploitation (Tutorial). In *SIGMOD*, 2022.

[46] W. Li, D. Wen, L. Qin, and Y. Zhang. An Overview of Path Queries on Graphs (Tutorial). In *ICDE*, 2025.

[47] X. Li, G. Cong, and Y. Cheng. Spatial Transition Learning on Road Networks with Deep Probabilistic Models. In *ICDE*, 2020.

[48] Y. Li, K. Fu, Z. Wang, C. Shahabi, J. Ye, and Y. Liu. Multi-task Representation Learning for Travel Time Estimation. In *SIGKDD*, 2018.

[49] Y. Li, D. Gunopulos, C. Lu, and L. Guibas. Urban Travel Time Prediction using a Small Number of GPS Floating Cars. In *ACM SIGSPATIAL*, 2017.

[50] X. Lin, Y. Wang, X. Xiao, Z. Li, and S. S. Bhowmick. Path Travel Time Estimation using Attribute-related Hybrid Trajectories Network. In *CIKM*, 2019.

[51] Y. Lin, H. Wan, J. Hu, S. Guo, B. Yang, Y. Lin, and C. S. Jensen. Origin-Destination Travel Time Oracle for Map-based Services. *SIGMOD*, 1(3), 2023.

[52] H. Liu, W. Jiang, S. Liu, and X. Chen. Uncertainty-Aware Probabilistic Travel Time Prediction for On-Demand Ride-Hailing at DiDi. In *SIGKDD*, 2023.

[53] A. Magdy and M. F. Mokbel. Microblogs Data Management and Analysis (Tutorial). In *SIGMOD*, 2016.

[54] A. Magdy and M. F. Mokbel. Microblogs Data Management and Analysis (Tutorial). In *ICDE*, 2016.

[55] M. F. Mokbel and M. Sarwat. Social Networking and Mobility: A Data Management Perspective (Tutorial). In *PVLDB*, 2013.

[56] S. Mridha, N. Ganguly, and S. Bhattacharya. Link Travel Time Prediction from Large Scale Endpoint Data. In *ACM SIGSPATIAL*, 2017.

[57] R. Naik. ETA (Estimated Time of Arrival) Reliability at Lyft. Lyft, Engineering, 2024. Retrieved June 29, 2026 from https://eng.lyft.com/eta-estimated-time-of-arrival-reliability-at-lyft-d4ca2720bda8.

[58] T. Nakata and J.-I. Takeuchi. Mining Traffic Data from Probe-Car System for Travel Time Prediction. In *SIGKDD*, 2004.

[59] S. M. Nascimento, J. A. F. de Macedo, H. C. V. Lopes, T. L. C. da Silva, M. A. Casanova, and J. de Castro Machado. On computing travel time functions from Trajectory Data Streams. In *ACM SIGSPATIAL*, 2017.

[60] A. Orda and R. Rom. Shortest-path and Minimum-delay Algorithms in Networks with Time-Dependent Edge-Length. *Journal of the ACM*, 37(3), 1990.

[61] Open Source Routing Machine (OSRM). Retrieved June 29, 2026 from http://project-osrm.org/.

[62] M. Pavlenko. Estimated Time of Arrival: How to Approach ETA Calculations in Logistics. AltexSoft, 2021. Retrieved June 29, 2026 from https://www.altexsoft.com/blog/estimated-time-of-arrival/.

[63] S. A. Pedersen, B. Yang, and C. S. Jensen. Anytime Stochastic Routing with Hybrid Learning. *PVLDB*, 13(9), 2020.

[64] S. A. Pedersen, B. Yang, and C. S. Jensen. Fast Stochastic Routing Under Time-varying Uncertainty. *VLDB J.*, 29(4), 2020.
[65] I. Sabek and M. Mokbel. Machine Learning Meets Big Spatial Data (Tutorial). In *ICDE*, 2020.
[66] I. Sabek and M. F. Mokbel. Machine Learning Meets Big Spatial Data (Tutorial). *PVLDB*, 2019.
[67] Shipping ETAs: How an Estimated Time of Arrival Boosts Supply Chain Performance. Shippeo, 2021. Retrieved June 29, 2026 from https://www.shippeo.com/blog/shipping-etas-how-an-estimated-time-of-arrival-boosts-supply-chain-performance.
[68] Y. Sun, Y. Wang, K. Fu, Z. Wang, Z. Yan, C. Zhang, and J. Ye. FMA-ETA: Estimating Travel Time Entirely Based on FFN with Attention. In *ICASSP*, 2021.
[69] W. Tian, J. Shi, S. Luo, H. Li, X. Xie, and Y. Zou. Effective and Efficient Route Planning using Historical Trajectories on Road Networks. *PVLDB*, 16(10), 2023.
[70] L. Tran, M. Y. Mun, M. Lim, J. Yamato, N. Huh, and C. Shahabi. DeepTRANS: a Deep Learning System for Public Bus Travel Time Estimation using Traffic Forecasting. *PVLDB*, 13(12), 2020.
[71] DeepETA: How Uber Predicts Arrival Times. Uber Blog, 2022. Retrieved June 29, 2026 from https://www.uber.com/blog/deepeta-how-uber-predicts-arrival-times/.
[72] Valhalla Project. Valhalla: Open Source Routing Engine. Retrieved June 29, 2026 from https://valhalla.github.io/valhalla/.
[73] F. Wan, L. Li, K. Wang, L. Chen, Y. Gao, W. Jiang, and S. Pu. MTTPRE: A Multi-Scale Spatial-Temporal Model for Travel Time Prediction. In *ACM SIGSPATIAL*, 2022.
[74] D. Wang, J. Zhang, W. Cao, J. Li, and Y. Zheng. When Will You Arrive? Estimating Travel Time Based on Deep Neural Networks. *AAAI*, 32(1), 2018.
[75] H. Wang, X. Tang, Y. Kuo, D. Kifer, and Z. Li. A Simple Baseline for Travel Time Estimation using Large-scale Trip Data. *ACM Transactions on Intelligent Systems and Technology, TIST*, 10(2), 2019.
[76] K. Wang, M. Cai, J. Wang, and E. Chen. From Predictive to Generative - How Michelangelo Accelerates Uber's AI Journey. Uber Blog, 2024. Retrieved June 29, 2026 from https://www.uber.com/blog/from-predictive-to-generative-ai/.
[77] Y. Wang, G. Li, and N. Tang. Querying Shortest Paths on Time Dependent Road Networks. *PVLDB*, 12(11), 2019.
[78] Y. Wang, Y. Zheng, and Y. Xue. Travel Time Estimation of a Path using Sparse Trajectories. In *SIGKDD*, 2014.
[79] Z. Wang, K. Fu, and J. Ye. Learning to Estimate the Travel Time. In *SIGKDD*, 2018.
[80] V. J. Wei, R. C.-W. Wong, and C. Long. Architecture-intact Oracle for Fastest Path and Time Queries on Dynamic Spatial Networks. In *SIGMOD*, 2020.
[81] L. Wu, X. Xiao, D. Deng, G. Cong, A. D. Zhu, and S. Zhou. Shortest Path and Distance Queries on Road Networks: An Experimental Evaluation. *PVLDB*, 5(5), 2012.
[82] N. Wu, J. Wang, W. X. Zhao, and Y. Jin. Learning to Effectively Estimate the Travel Time for Fastest Route Recommendation. In *CIKM*, 2019.
[83] Z. Xu, L. Li, M. Zhang, Y. Xu, and X. Zhou. Ultra-flexible, Explainable, and Scalable Traffic Prediction with Dynamic Future Routes. In *ICDE*, 2025.
[84] B. Yang, J. Dai, C. Guo, C. S. Jensen, and J. Hu. PACE: A PAth-CEntric Paradigm for Stochastic Path Finding. *VLDB J.*, 27(2), 2018.
[85] B. Yang, C. Guo, and C. S. Jensen. Travel Cost Inference from Sparse, Spatio Temporally Correlated Time Series using Markov Models. *PVLDB*, 6(9), 2013.
[86] B. Yang, C. Guo, C. S. Jensen, M. Kaul, and S. Shang. Stochastic Skyline Route Planning Under Time-varying Uncertainty. In *ICDE*, 2014.
[87] B. Yang, C. Guo, Y. Ma, and C. S. Jensen. Toward Personalized, Context-aware Routing. *VLDB J.*, 24(2), 2015.
[88] B. Yang, Y. Liang, C. Guo, and C. S. Jensen. Data Driven Decision Making with Time Series and Spatio-temporal Data (Tutorial). In *ICDE*, 2025.
[89] X. Yu, J. Wang, Y. Yang, Q. Huang, and K. Qu. BIGCity: A Universal Spatiotemporal Model for Unified Trajectory and Traffic State Data Analysis. In *ICDE*, 2025.
[90] H. Yuan, G. Li, and Z. Bao. Route Travel Time Estimation on a Road Network Revisited: Heterogeneity, Proximity, Periodicity and Dynamicity. *PVLDB*, 16(3), 2022.
[91] H. Yuan, G. Li, Z. Bao, and L. Feng. Effective Travel Time Estimation: When Historical Trajectories over Road Networks Matter. In *SIGMOD*, 2020.
[92] Y. Yuan, X. Lian, G. Wang, Y. Ma, and Y. Wang. Constrained Shortest Path Query in a Large Time-dependent Graph. *PVLDB*, 12(10), 2019.
[93] C. Zhang, L. Warne, Z. Jiang, Q. Xu, H. Jenq, J. Luo, and P. Varma. Improving ETAs with Multi-Task Models, Deep Learning, and Probabilistic Forecasts. DoorDash Engineering Blog, 2024. Retrieved June 29, 2026 from https://careersatdoordash.com/blog/improving-etas-with-multi-task-models-deep-learning-and-probabilistic-forecasts/.
[94] X. Zhang and A. Eldawy. Spatial Query Optimization With Learning (Tutorial). *PVLDB*, 2024.
[95] J. Zhao, L. Li, M. Zhang, H. Ma, and X. Zhou. A Just-in-time Framework for Routing-oriented Traffic Prediction. In *ICDE*, 2025.
[96] T. Zhao, S. Huang, Y. Wang, C. Chai, and G. Li. RNE: Computing Shortest Paths using Road Network Embedding. *VLDB J.*, 31(3), 2022.
[97] G. Zheng, C. Liu, H. Wei, C. Chen, and Z. Li. Rebuilding City-wide Traffic Origin Destination from Road Speed Data. In *ICDE*, 2021.
[98] Z. Zheng, H. Yuan, M. Chen, and S. Wang. RLER-TTE: An Efficient and Effective Framework for En Route Travel Time Estimation with Reinforcement Learning. *SIGMOD*, 3(1), 2025.
[99] N. Zygouras, N. Panagiotou, Y. Li, D. Gunopulos, and L. Guibas. HTTE: A Hybrid Technique For Travel Time Estimation In Sparse Data Environments. In *ACM SIGSPATIAL*, 2019.