LEVERAGING GRAPH MACHINE LEARNING FOR PREDICTING TRAFFIC CONGESTION AND OPTIMIZING VEHICLE ROUTING

P. MADHUSOODHANAN[1], S. FELIXIA[2], K. JANAKI[3], R. KRISHNA KUMARI[1,*]

[1]Career Development Centre, College of Engineering and Technology, SRM Institute of Science and Technology, Kattankulathur, Chennai-603203, Tamilnadu, India

[2]Department of Mathematics, Panimalar Engineering College, Poonamallee, Chennai-600123, Tamil Nadu, India

[3]Department of Mathematics, Saveetha Engineering College, Saveetha Nagar, Chennai-602105, Tamilnadu, India

[*]Corresponding author: krishrengan@gmail.com

Received October. 23, 2023

Abstract. Traffic congestion in urban areas is a persistent problem with economic and environmental implications. To address this challenge, advanced technologies are increasingly being employed, and one of the most promising approaches involves graph machine learning. This article explores the application of graph machine learning in predicting traffic congestion and optimizing vehicle routing in urban traffic networks. We discuss how graph representations of road networks, combined with historical and real-time data, can be harnessed to develop machine learning models that predict congestion and optimize vehicle routes. By employing this approach, urban traffic management can become more efficient and responsive, leading to reduced congestion and improved transportation systems.

2020 Mathematics Subject Classification. 05C90, 68R10.

Key words and phrases. traffic congestion; vehicle routing; graph machine learning; urban traffic networks; real-time data; machine learning models.

1. Introduction

Traffic congestion in urban areas is a complex problem that affects the daily lives of millions of people, resulting in economic losses, increased pollution, and reduced quality of life. As cities continue to grow, managing traffic effectively becomes paramount. Traditional traffic management systems often fall short in dealing with the dynamic and intricate nature of traffic flow. This is where graph machine learning, a subfield of machine learning, proves to be a game-changer [5–7].

Graph machine learning leverages the power of graphs or networks to model and predict complex relationships and dependencies [1, 2]. In the context of traffic management, the road network is

DOI: 10.28924/APJM/11-1

represented as a graph, with intersections and road segments as nodes, and connections or traffic flow as edges. Machine learning models are then applied to this graph to predict traffic congestion and optimize vehicle routing [3,4].

In this article, we will delve into the applications of graph machine learning in predicting traffic congestion and optimizing vehicle routing. We will explore the steps involved in this process, from data collection and graph representation to the application of machine learning models for congestion prediction and route optimization. The benefits of this approach in addressing the challenges of urban traffic management are significant, as it can lead to reduced congestion, shorter travel times, and more sustainable transportation systems.

2. Graph Machine Learning Basics

In the domain of traffic management, graph machine learning serves as a foundational and transformative approach, offering an effective means of addressing the intricate dynamics of urban traffic networks. This section delves into the fundamental principles of graph machine learning and elucidates their significance in the context of traffic management.

2.1. **Graph Representation.**

Central to the domain of graph machine learning is the representation of data as interconnected graphs or networks. In the realm of traffic management, this entails the modeling of the road network as a graph. The construction of such a graph is a multifaceted process:

- **Nodes:** Nodes within the road network graph correspond to vital elements of the traffic infrastructure. These can encompass intersections, traffic signals, road segments, or other pivotal locations within the transportation network. Each node in the graph embodies a real-world geographical point of reference.
- **Edges:** Edges within the graph denote the connections between nodes. These connections encapsulate the routes that vehicles may traverse, encompassing roadways, highways, streets, and other conduits. The edges serve as the linchpin for depicting the relationships between various components of the road network.
- **Attributes:** Both nodes and edges are endowed with an array of attributes. Nodes may harbor attributes such as geographical coordinates, classifications of intersections (e.g., roundabouts, four-way stops), or information regarding the presence of traffic control elements like traffic lights. Edges, on the other hand, may carry attributes encompassing speed limits, historical traffic flow data, road conditions, weather conditions, and a multitude of other pertinent details.

The formulation of the road network in this graph-based structure affords the capacity to encapsulate and model the inherent intricacies and interconnections within the traffic infrastructure. Nodes and edges represent specific locations and their relationships, while attributes bestow essential contextual information that serves as the bedrock for subsequent machine learning analysis.

2.2. **Relevance to Traffic Management.** The pertinence of graph machine learning to the realm of traffic management is manifest when one considers the convoluted and interconnected nature of urban traffic systems. Traditional traffic management systems often rely on linear models that are ill-equipped to grapple with the dynamic and intricate components of traffic networks. Graph machine learning ascends to prominence in this milieu due to several compelling reasons:

- **Complex Relationships:** Urban traffic networks are veritable labyrinths of complex relationships and dependencies. Within the graph, nodes represent not merely geographical points but also encapsulate the state of traffic at these locations, while edges encapsulate the dynamics of traffic flow. Machine learning models can scrutinize these intricate relationships to prognosticate congestion and optimize routing in a data-driven manner.
- **Real-time Adaptability:** Traffic conditions in urban settings are exceptionally fluid, subject to rapid fluctuations. Graph machine learning models exhibit the capability to adapt in real-time as conditions evolve. This attribute renders them eminently suitable for dynamic traffic management, as they can recalibrate routing strategies and congestion predictions promptly in response to shifting conditions.
- **Data-Driven Insights:** Through the judicious analysis of historical data and real-time information, graph machine learning bequeaths a treasure trove of data-driven insights. These insights empower traffic authorities to make informed decisions with respect to traffic control, route optimization, and resource allocation. Traffic management strategies underpinned by data-driven insights are inherently more efficient and adaptable.

To synopsize, the fundamental tenets of graph machine learning, in concert with the graph representation of the road network, furnish a robust and versatile framework for addressing the labyrinthine challenges of urban traffic management. This framework excels in capturing the intricate interplay between road segments, intersections, and real-time traffic conditions. It facilitates the predictive modeling of congestion and the optimization of vehicle routing in a data-informed, adaptable, and dynamic manner.

The subsequent sections will delve even deeper into the practical applications of these concepts, elucidating the mechanisms for predicting traffic congestion and optimizing vehicle routing within this framework.

3. Predicting Traffic Congestion

Predicting traffic congestion is a multifaceted process that is pivotal in modern traffic management. It involves various stages, from data collection to real-time alerting systems, and leverages the power of graph machine learning.

3.1. **Data Collection.** Data collection is the cornerstone of predicting traffic congestion. It comprises two primary types of data:

3.1.1. *Historical Data.* Historical traffic data encompasses a vast repository of information, including traffic volume, vehicle speed, and congestion levels, recorded at various times and locations. This historical dataset is invaluable for building predictive models as it allows for the identification of recurring traffic patterns and congestion trends.

3.1.2. *Real-time Data.* Real-time data sources include sensors, GPS devices, traffic cameras, and other monitoring systems that provide up-to-the-minute information about traffic conditions. This real-time data is instrumental in keeping congestion predictions accurate and relevant.

3.2. **Building a Graph Representation.** The road network is represented as a graph, which serves as the foundation for predicting traffic congestion. This graph-based representation encapsulates the intricate structure of the road network.

3.2.1. *Nodes.* Nodes within the graph represent critical components of the traffic infrastructure, such as intersections, traffic signals, road segments, and other key locations within the transportation network. Each node corresponds to a specific geographical point in the real-world road network.

3.2.2. *Edges.* Edges of the graph denote connections between nodes, representing the routes that vehicles can traverse, such as roadways, highways, and streets. These edges capture the relationships between different components of the road network.

3.2.3. *Attributes.* Both nodes and edges can carry various attributes. Nodes may include geographical coordinates, intersection types, and the presence of traffic control elements like traffic lights. Edges may feature attributes like speed limits, historical traffic flow data, road conditions, and weather information.

3.3. **Feature Engineering for Nodes and Edges.** To predict traffic congestion accurately, it's essential to extract and define relevant features for both nodes and edges within the road network graph. These features may include:

3.3.1. *Traffic Speed.* Data on traffic speed at nodes and edges is a critical feature. It helps in understanding the current state of traffic and its flow within the network.

3.3.2. *Traffic Volume.* Historical data on traffic volume at specific times and locations allows models to identify patterns in traffic density and volume.

3.3.3. *Weather Conditions.* Weather conditions, such as rain or snow, can significantly impact traffic. Incorporating this data is essential for capturing environmental factors.

3.3.4. *Time of Day.* Traffic patterns vary throughout the day. Taking the time of day into account helps models adapt to changing traffic dynamics.

3.4. **Training Machine Learning Models for Congestion Prediction.** Machine learning models, particularly graph neural networks (GNNs), are employed to predict traffic congestion. GNNs are well-suited for handling graph-structured data and can capture dependencies and patterns within the road network.

Theorem 3.1. *Let $G = (V, E)$ be a directed or undirected graph representing a road network, where V denotes the set of nodes corresponding to locations within the network, and E denotes the set of edges representing connections between these locations. Each node $v \in V$ is associated with a feature vector $\mathbf{x}_v \in \mathbb{R}^d$, and each edge $e \in E$ is associated with a weight $w(e) \in \mathbb{R}$.*

Given historical traffic data $\mathcal{D} = \{(\mathbf{x}_v, y_v)\}$, where $\mathbf{x}_v$ is the feature vector of node v and y_v represents the congestion level at node v, the objective is to learn a function $f : \mathbb{R}^d \times \mathbb{R}^{d_v} \to \mathbb{R}$ for each node v in the graph. Here, $\mathbb{R}^{d_v}$ represents the feature vectors of the neighbors of node v.

The function f predicts the congestion level at node v based on the combination of its own feature vector and the feature vectors of its neighboring nodes, taking into account the graph structure.

Proof. The problem of predicting traffic congestion in a road network can be mathematically described as follows:

(1) Let $G = (V, E)$ be a graph, where V represents the set of nodes corresponding to locations within the road network, and E represents the set of edges connecting these locations.
(2) Each node $v \in V$ is characterized by a feature vector $\mathbf{x}_v \in \mathbb{R}^d$, where d is the dimension of the feature vector. These feature vectors capture information about the node, such as its geographical attributes, historical traffic data, or any other relevant information.
(3) Each edge $e \in E$ is associated with a weight $w(e) \in \mathbb{R}$, which may represent the distance, travel time, or other relevant metrics between connected nodes.
(4) The goal is to predict traffic congestion levels for each node in the graph. Let $\mathcal{D} = \{(\mathbf{x}_v, y_v)\}$ represent the historical traffic data, where $\mathbf{x}_v$ is the feature vector of node v, and y_v is the congestion level at node v.

To achieve this prediction, a machine learning model is trained for each node v to learn a function $f : \mathbb{R}^d \times \mathbb{R}^{d_v} \to \mathbb{R}$, where d_v is the dimension of the feature vectors of the neighbors of node v in the graph. This function takes as input the feature vector of the node itself ($\mathbf{x}_v$) and the feature vectors of its neighbors in the graph ($\mathbf{x}_{\text{neighbors}}$) and outputs a predicted congestion level.

The utilization of graph neural networks (GNNs) is well-suited for this task, as they can capture dependencies and patterns within the road network graph. GNNs excel at learning from graph-structured data and provide an effective means of predicting traffic congestion levels at various locations within the network. $\square$

3.5. **Real-Time Predictions and Alerting Systems.** The trained machine learning models provide real-time predictions of traffic congestion within the road network. These predictions are vital for proactive traffic management. To ensure timely responses to congestion, alerting systems are implemented. These systems notify traffic management authorities, commuters, and other relevant stakeholders about congestion, enabling them to take immediate action to alleviate its impact.

4. Optimizing Vehicle Routing

Optimizing vehicle routing is a crucial component of traffic management, aimed at reducing travel times, fuel consumption, and enhancing the overall efficiency of transportation systems. This section outlines the process of optimizing vehicle routing using the same road network graph, inputting vehicle locations and destination data, employing graph algorithms for optimization, and continuously updating routes based on real-time traffic data.

Theorem 4.1. *Let $G = (V, E)$ be a weighted directed graph representing a road network, where V is the set of nodes representing intersections and locations, and E is the set of directed edges representing road segments. Each edge $e \in E$ is associated with a weight function $w(e)$, representing the travel cost or distance.*

For a set of vehicles V_h with specific starting nodes $s(v) \in V$ and destination nodes $d(v) \in V$, the optimization problem of finding the most efficient vehicle routes, such that the total travel cost is minimized, can be mathematically formulated as:

Minimize:

$$\sum_{v \in V_h} \sum_{(u,v) \in P_v} w(u, v)$$

subject to:

$$P_v = \mathit{ShortestPath}(G, s(v), d(v)), \forall v \in V_h$$

$$\mathit{Congestion}(G, P_v) \leq C, \forall v \in V_h$$

where P_v is the set of paths for each vehicle, ShortestPath$(G, s(v), d(v))$ is the set of all possible shortest paths from $s(v)$ to $d(v)$ in graph G, and Congestion(G, P_v) is the congestion metric for a path P_v, and C is the maximum allowable congestion threshold.

Proof. We start by defining the mathematical optimization problem for vehicle routing. Given a road network represented as a weighted directed graph G, and a set of vehicles V_h each with specific starting and destination nodes, our objective is to find the routes for each vehicle in V_h such that the total travel cost is minimized while ensuring that the congestion on each route does not exceed a given threshold C.

To formalize this, we use the following mathematical expressions:

1. Minimize the total travel cost:

$$\sum_{v \in V_h} \sum_{(u,v) \in P_v} w(u, v)$$

where P_v represents the path taken by vehicle v from its starting node to its destination.

2. Subject to finding the shortest path for each vehicle:

$$P_v = \text{ShortestPath}(G, s(v), d(v)), \forall v \in V_h$$

This constraint ensures that each vehicle follows the shortest path from its starting node to its destination.

3. Subject to congestion constraints:

$$\text{Congestion}(G, P_v) \leq C, \forall v \in V_h$$

This constraint ensures that the congestion along each vehicle's route does not exceed the predefined threshold C.

The optimization problem aims to minimize the sum of travel costs for all vehicles, taking into account both the shortest path constraints and the congestion constraints. Solving this problem results in optimized vehicle routes that minimize travel costs while respecting congestion limitations.

This formulation aligns with mathematical optimization principles and graph theory, providing a rigorous foundation for the optimization of vehicle routing in the context of road networks. □

Theorem 4.2. *Let $G = (V, E)$ be a directed graph representing a road network, where V is the set of nodes corresponding to intersections and locations, and E is the set of directed edges representing road segments. Each edge $e \in E$ is associated with a weight function $w(e)$, representing the travel cost or distance.*

Consider a set of vehicles V_h with specific starting nodes $s(v) \in V$ and destination nodes $d(v) \in V$. The optimization problem of finding the most efficient vehicle routes, such that the total travel cost is minimized, can be mathematically formulated as an instance of the Traveling Salesman Problem (TSP).

In TSP, the objective is to find a permutation π of the set of nodes V such that the total cost of visiting the nodes in the order defined by π is minimized. The cost of visiting a node v_i followed by v_j is represented by $w(v_i, v_j)$.

The optimization problem can be defined as follows: Minimize:

$$\sum_{i=1}^{n-1} w(v_{\pi(i)}, v_{\pi(i+1)}) + w(v_{\pi(n)}, v_{\pi(1)})$$

where n is the number of nodes (vehicles) and π represents a permutation of V.

Subject to the constraints:

- *Each node is visited exactly once.*
- *The starting node of each vehicle is fixed in the permutation, i.e., $\pi(1) = s(v)$ for each $v \in V_h$.*
- *The ending node of each vehicle is also fixed, i.e., $\pi(n) = d(v)$ for each $v \in V_h$.*

Proof. The optimization problem for optimizing vehicle routing can be mathematically framed as an instance of the Traveling Salesman Problem (TSP), a well-known combinatorial optimization problem. In TSP, the objective is to find the most efficient route that visits a set of locations exactly once and returns to the starting location, minimizing the total travel cost.

To establish the equivalence between the vehicle routing problem and TSP, we map the components as follows:

(1) The nodes in the TSP correspond to the locations (intersections and destinations).
(2) The edge weights in the TSP represent the travel costs (e.g., distance or time) between locations.
(3) The starting node of each vehicle corresponds to the first node visited in the TSP permutation.
(4) The destination node of each vehicle corresponds to the last node visited in the TSP permutation.

The optimization problem is to find a permutation of the nodes that minimizes the total cost of visiting all nodes. In this context, the nodes are organized in a circular manner, ensuring that each vehicle reaches its destination. Thus, the TSP captures the essence of finding the most efficient routes for the vehicles.

By solving the TSP, we obtain an optimal ordering of nodes, which corresponds to an optimized route for each vehicle, minimizing the total travel cost while visiting all specified locations. This demonstrates that the vehicle routing problem can be mathematically reduced to solving the TSP, a classic problem in combinatorial optimization. □

4.1. **Using the Same Road Network Graph for Route Optimization.** Efficiency is achieved by capitalizing on the same road network graph used for predicting traffic congestion. This graph serves as a consistent and data-rich framework for optimizing vehicle routes.

4.2. **Input Vehicle Locations and Destination Data.** Inputting vehicle locations and destination data is critical for route optimization. This information can be sourced from a variety of systems, including GPS devices, vehicle telematics, or user input for ride-sharing services. It allows the system to determine the current location of vehicles and their intended destinations.

4.3. **Employing Graph Algorithms for Route Optimization.** Graph algorithms come to the fore in the process of route optimization. These algorithms, such as Dijkstra's algorithm, A* search, and variants thereof, are applied to the road network graph to compute the most efficient routes from each vehicle's current location to its destination. These algorithms consider factors like distance, traffic speed, and the presence of congestion to determine optimal routes.

4.4. **Continuously Updating Routes Based on Real-Time Traffic Data.** To adapt to changing traffic conditions, vehicle routes are continuously updated based on real-time traffic data. This real-time information is derived from various sources, including traffic sensors, cameras, and data from other vehicles on the road. It allows the system to dynamically reroute vehicles, avoiding congested areas, and minimizing travel times.

5. The Benefits of Graph Machine Learning

Graph machine learning offers a myriad of advantages in the realm of traffic management. This section explores these benefits in depth, including complex relationship analysis and real-time adaptability, data-driven insights for efficient traffic management, and the resultant reductions in traffic congestion and environmental impact.

5.1. **Complex Relationship Analysis and Real-Time Adaptability.** One of the primary advantages of graph machine learning is its ability to capture and analyze the complex relationships and dependencies within the road network. The graph structure allows for the modeling of intricate interconnections between different parts of the network. Additionally, the machine learning models can adapt in real-time to evolving traffic conditions, making them highly responsive to dynamic environments.

5.2. **Data-Driven Insights for Efficient Traffic Management.** Graph machine learning is inherently data-driven. By analyzing historical data and real-time information, it provides traffic management authorities with invaluable insights. These insights empower authorities to make informed decisions regarding traffic control, route optimization, and resource allocation. The result is more efficient and adaptive traffic management strategies that are responsive to real-world conditions. The following algorithm, implemented in python, demonstrates how you can gain data-driven insights for efficient traffic management. In this example, we'll use synthetic data, but in a real-world scenario, you'd gather data from various sources such as traffic cameras, sensors, or GPS devices.
Here

- We generate synthetic traffic data with timestamps and traffic flow values.
- We calculate the average traffic flow by hour and identify peak traffic hours.
- We analyze traffic trends to determine when traffic is typically heavier.
- We analyze the data for decision-making, checking for congested periods that may require traffic management interventions.
- Finally, we visualize the traffic flow and congested periods.

Algorithm 1: Traffic Data Analysis

Data: Synthetic traffic data with timestamps

Result: Data-driven insights for traffic management

Input : Traffic data: *time_intervals, traffic_data*

Output : Insights: *peak_hour, traffic_trends, congested_periods*

Generate synthetic traffic data;

for *each 15-minute interval in time_intervals* **do**

  Randomly generate traffic flow values;

Create a DataFrame;

Convert traffic data into a pandas DataFrame;

Calculate average traffic flow by hour;

for *each hour of the day* **do**

  Calculate the average traffic flow;

Identify peak traffic hour;

Find the hour with the highest average traffic flow;

peak_hour = Peak traffic hour;

Analyze traffic trends;

Calculate the average traffic flow during morning and evening rush hours;

traffic_trends = Morning rush vs. Evening rush;

Analyze traffic data for decision-making;

Set a threshold for high traffic;

Identify congested periods when traffic flow exceeds the threshold;

congested_periods = Periods of high traffic;

Visualize congested periods;

If congested periods exist, create a plot showing traffic flow with congested periods highlighted;

Display the plots;

Use matplotlib to display the plots;

Here we used synthetic data, but in practice, replace the synthetic data with real traffic data from sensors or other sources. The insights gained from this data can help in making informed decisions for efficient traffic management. The program output is essential for making data-driven decisions in traffic management and optimizing vehicle routing. The output includes information about traffic flow, congestion, and peak traffic hours, providing valuable insights for traffic management. In summary, this Python program demonstrates how to analyze synthetic traffic data to gain data-driven insights for efficient traffic management. It identifies peak traffic hours, traffic trends, and congested periods, which can inform decisions related to traffic management and interventions. In a real-world scenario, this program would be used with real traffic data collected from sensors, cameras, or other sources to make informed decisions about traffic management and optimization.

5.3. **Reduced Congestion and Environmental Impact.** One of the most significant outcomes of effective traffic management using graph machine learning is the substantial reduction in traffic congestion. Optimized vehicle routing, informed by real-time data and congestion predictions, leads to shorter travel times and fewer delays for commuters. Additionally, this reduction in congestion translates into lower fuel consumption and emissions, contributing to a decrease in the environmental impact of transportation. The benefits are not only economic but also environmental, making urban transportation more sustainable and eco-friendly.

6. Conclusion and Future Work

In an era where cities are continually growing and traffic congestion is a constant concern, the application of graph machine learning emerges as a powerful tool for traffic management. By predicting congestion and optimizing vehicle routing, it offers the potential to transform urban transportation, making it more efficient, sustainable, and enjoyable for all.

As technology continues to advance, it's likely that the application of graph machine learning will play an increasingly significant role in the way we navigate our urban environments. The future holds immense promise for this field, and areas of future work warrant exploration.

Future work in this field may include:

- Improved Data Integration: Integrating data from a broader spectrum of sources, including social media, weather stations, and individual vehicles, can enhance the accuracy and timeliness of congestion predictions. Developing robust data integration strategies will be crucial.
- Multi-Modal Transportation: The integration of various modes of transportation, including buses, trains, bicycles, and pedestrians, into the same predictive framework will result in a more holistic approach to urban mobility.
- Behavioral Analysis: Understanding the behavior of individual drivers and the choices they make in response to congestion predictions can lead to more precise and targeted traffic management strategies.

In conclusion, the application of graph machine learning in traffic management promises a brighter future for urban transportation. By predicting and mitigating congestion and optimizing vehicle routing, we can look forward to more efficient and sustainable transportation systems. The transformation has only just begun, and the journey toward smarter, more enjoyable, and environmentally responsible urban mobility continues.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] Y. Djenouri, A. Belhadi, G. Srivastava, J.C.W. Lin, Hybrid graph convolution neural network and branch-and-bound optimization for traffic flow forecasting, Future Gen. Comp. Syst. 139 (2023), 100–108. https://doi.org/10.1016/j.future.2022.09.018.

[2] J.J.Q. Yu, W. Yu, J. Gu, Online vehicle routing with neural combinatorial optimization and deep reinforcement learning, IEEE Trans. Intell. Transport. Syst. 20 (2019), 3806–3817. https://doi.org/10.1109/tits.2019.2909109.

[3] J. Li, D. Fu, Q. Yuan, et al. A traffic prediction enabled double rewarded value iteration network for route planning, IEEE Trans. Veh. Technol. 68 (2019), 4170–4181. https://doi.org/10.1109/tvt.2019.2893173.

[4] S. Luan, R. Ke, Z. Huang, et al. Traffic congestion propagation inference using dynamic Bayesian graph convolution network, Transport. Res. Part C: Emerg. Technol. 135 (2022), 103526. https://doi.org/10.1016/j.trc.2021.103526.

[5] J. Rzeszótko, S.H. Nguyen, Machine learning for traffic prediction, Fund. Inform. 119 (2012), 407–420. https://doi.org/10.3233/fi-2012-745.

[6] H. Xiong, A. Vahedian, X. Zhou, Y. Li, J. Luo, Predicting traffic congestion propagation patterns: a propagation graph approach, in: Proceedings of the 11th ACM SIGSPATIAL International Workshop on Computational Transportation Science, ACM, Seattle WA USA, 2018: pp. 60–69. https://doi.org/10.1145/3283207.3283213.

[7] Y. Yoo, J. Shin, S. Kyeong, Medicare fraud detection using graph analysis: a comparative study of machine learning and graph neural networks, IEEE Access. 11 (2023), 88278–88294. https://doi.org/10.1109/access.2023.3305962.