mathematics

MDPI

Article

A Multi-Model Dynamic Selection Framework Using Deep Contextual Bandits for Urban Traffic Flow Prediction in Large-Scale Road Networks

Silai Chen [1], Shengfeng Mao [1], Zongcheng Zhang [1], Xiaoyuan Zhang [1], Yunxia Wu [1], Yangsheng Jiang [1,2,3] and Zhihong Yao [1,2,3,*]

[1] School of Transportation and Logistics, Southwest Jiaotong University, Chengdu 611756, China; chensilai@my.swjtu.edu.cn (S.C.); shengfeng.mao@my.swjtu.edu.cn (S.M.); zongcheng.zhang@my.swjtu.edu.cn (Z.Z.); zhangxiaoyuan@my.swjtu.edu.cn (X.Z.); yxwu@my.swjtu.edu.cn (Y.W.); jiangyangsheng@swjtu.edu.cn (Y.J.)
[2] National Engineering Laboratory of Integrated Transportation Big Data Application Technology, Southwest Jiaotong University, Chengdu 611756, China
[3] National United Engineering Laboratory of Integrated and Intelligent Transportation, Southwest Jiaotong University, Chengdu 611756, China
* Correspondence: zhyao@swjtu.edu.cn; Tel.: +86-15528250279

Abstract

To address the challenge of model selection in large-scale traffic flow prediction tasks, this paper proposes a dynamic multi-model selection framework based on Deep Contextual Bandits (DCB). Centered on the optimal combination of sub-models, the framework leverages contextual information of road segments to select dynamically among candidate predictors, achieving more efficient and accurate traffic flow prediction. Several mechanisms are introduced to improve strategy learning and convergence, including a baseline network, experience replay, double-model estimation, and prioritized experience sampling. A clustering-based strategy is further designed to reduce the search space and enhance the generalization and transferability. Experiments on real-world traffic datasets demonstrate that the proposed framework significantly outperforms traditional static fusion methods, reinforcement learning (RL) baselines, and mainstream spatiotemporal prediction models. In particular, the framework yields a 1.0% improvement in R^2 and a 3.2% reduction in MAE compared to state-of-the-art baselines, while reducing inference time by 43.1%. Moreover, the proposed framework shows strong capability in adaptive model selection under varying contexts, with ablation studies confirming the effectiveness of its key components.

Keywords: traffic flow prediction; deep contextual bandits; multi-model fusion; dynamic decision-making; reinforcement

MSC: 68T07

Check for updates

Academic Editor: Renato Bruni

Received: 3 January 2026
Revised: 25 January 2026
Accepted: 2 February 2026
Published: 4 February 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Traffic flow prediction in urban road networks is a prerequisite for developing strategies to alleviate congestion and enhance traffic capacity [1–3]. In large-scale urban networks, traffic flow across different road segments is influenced by various heterogeneous factors, including road network topology, spatiotemporal dependencies, and unexpected events such as traffic accidents or severe weather conditions [4–7]. For instance, traffic in central urban areas is typically shaped by the coupling effects of abrupt traffic surges during peak

hours and complex topological structures [8]. Quantitatively, a recent empirical study on urban road networks revealed significant spatial heterogeneity, with the coefficient of variation (CoV) of traffic conditions ranging widely from 0.102 to 0.57 across different zones during peak periods [9]. In contrast, suburban traffic is more susceptible to fluctuations due to weather changes or sparse vehicle flow [10]. This heterogeneity suggests that a single predictive model, constrained by limited generalization capabilities, is often insufficient to meet the accuracy and stability requirements for traffic flow forecasting across diverse road segments in large-scale networks. Consequently, it becomes a meaningful research challenge to develop a dynamic model selection framework that can accurately account for the diverse combination effects of these influencing factors when selecting appropriate prediction models for each segment [11].

Classical traffic flow prediction models for urban networks generally fall into three categories. The first comprises traditional statistical models such as ARIMA, which are typically limited in capturing complex nonlinear relationships and are highly sensitive to noise in large-scale datasets [12]. The second category includes machine learning model such as support vector regression (SVR) and random forests (RFs). These models rely heavily on handcrafted feature engineering, making them impractical for deployment across extensive and heterogeneous networks [13]. The third category, which has gained popularity recently, consists of deep learning models such as long short-term memory (LSTM) networks and graph convolutional networks (GCNs). While these models can automatically extract spatial and temporal features, research shows that their single-architecture design still lacks the flexibility to dynamically adapt to the varying characteristics of different road segments in large-scale networks [14,15]. Classical single-model approaches suffer from limited generalization ability and tend to produce high global prediction errors when applied to complex, heterogeneous urban traffic systems.

Many researchers have attempted to capture the spatiotemporal heterogeneity of traffic flow through clustering methods to address the limitations of classical single-prediction models, namely, their weak generalization capabilities and unstable prediction accuracy. By categorizing different types of traffic patterns, they aim to assign distinct prediction models accordingly, thereby improving efficiency and accuracy. For instance, Ryu et al. [16] employed the clustering of large applications based on the randomized sampling (CLARANS) algorithm to cluster spatiotemporal traffic patterns, leveraging the classification of historical data to enhance prediction performance. Huang et al. [17] proposed an enhanced entropy regularization k-means (ERKM) algorithm, which improves clustering quality by maximizing inter-cluster distances based on the K-means framework. The resulting clusters were then integrated into an attention-based convolutional LSTM architecture to construct a multimodal prediction model. Similarly, Deng et al. [18] introduced the cluster clustering-based modular integrated DNN (CC-MIDNN) model in the aviation domain, using K-means clustering to decompose flight tasks into parallel sub-networks. Their approach significantly improved prediction stability by incorporating Bayesian optimization to reduce network complexity.

While such methods enhance the generalization ability of predictive models to a certain extent via clustering-based dimensionality reduction, they are still static multi-model selection frameworks that follow a "cluster-then-predict" paradigm. This makes them poorly suited to dynamically adapting to stochastic variations in traffic flow across large-scale urban road networks. For example, using fixed clustering centers or predefined dependencies fails to capture real-time changes in road segment characteristics. Sudden congestion propagation or unexpected incidents can also break these predefined dependencies, leading to increased prediction errors. Furthermore, these approaches often overlook subtle and dynamic interactions between road segments, such as time-varying influences

from neighboring intersections or evolving traffic demand. As a result, their robustness deteriorates in noisy or volatile environments.

Nevertheless, ensemble prediction offers valuable insights for developing a dynamic multi-model selection framework. Specifically, by selecting the most suitable prediction model from a predefined pool of effective models for each road segment, it is possible to improve the stability and accuracy of traffic flow prediction across large-scale, heterogeneous road networks. In recent years, Reinforcement Learning (RL) has emerged as a promising approach to improve the generalization ability of predictive models further. RL methods operate under a two-stage "training–inference" paradigm, where the computationally intensive optimization is performed during the training phase. In contrast, the inference phase requires only lightweight online decision-making. This characteristic makes RL particularly suitable for the real-time prediction demands of large-scale urban road networks. However, current RL-based approaches predominantly focus on optimizing a single prediction model. To our knowledge, limited methodological exploration exists on applying reinforcement learning for adaptive model selection in heterogeneous road environments with dynamic segment characteristics.

Based on the above background, this paper proposes a multi-model dynamic selection framework built upon the Deep Contextual Bandit (DCB). The selection of DCB as the foundation is motivated by three key advantageous properties that align with the requirements of large-scale, heterogeneous traffic prediction:

(1) Sample Efficiency: Unlike Reinforcement Learning (RL), which requires complex state transition learning, DCB focuses on maximizing immediate rewards based on the current context. This formulation is theoretically advantageous for traffic prediction, where the optimal model depends primarily on the present pattern rather than future state sequences.
(2) Inference Efficiency and Interpretability: In contrast to soft-fusion ensembles (e.g., Mixture-of-Experts), DCB performs a deterministic selection of a single optimal submodel. This reduces computational overhead during inference and provides interpretability by explicitly identifying the preferred model type for specific traffic conditions.
(3) Handling Heterogeneity: DCB leverages deep neural networks to learn high-dimensional representations from traffic and topological features, enabling effective adaptation to spatiotemporal heterogeneity across thousands of road segments—a capability often limited in traditional bandit or static clustering approaches.

Building upon these strengths, the main contributions of this study are as follows.

(1) A DCB-based dynamic model selection framework. This study is the first to introduce the DCB framework into the domain of large-scale urban road network traffic flow prediction, developing a fully end-to-end multi-model dynamic selection system. Leveraging reinforcement learning mechanisms, the framework dynamically adapts to evolving traffic patterns (e.g., sudden congestion, demand shifts) and topological changes using real-time contextual features of road segments—such as historical traffic flow and topological attributes. It dynamically selects the optimal prediction sub-model for each segment, significantly reducing the prediction error caused by applying a single model to heterogeneous road networks and enhancing overall prediction stability. Furthermore, the framework exhibits strong scalability, making it applicable to urban road networks of varying sizes and structures.
(2) Reward-driven design with R^2 metric. We propose using the Coefficient of Determination (R^2) as the reward signal to quantify sub-model prediction accuracy. Due to its standardized nature, R^2 effectively measures the proportion of data variance explained by the model, offering a more robust and consistent evaluation metric. Un-

like traditional error-based indicators (e.g., RMSE, MAE), R^2 is less sensitive to traffic volume disparities or noise across road segments and more stable in the presence of outliers, thus improving comparability and reliability in model selection [19].

(3) A hybrid exploration strategy. We design a hybrid exploration mechanism that combines Priority Sampling with ε-greedy exploration. Priority Sampling dynamically adjusts learning focus based on the magnitude of performance differences (e.g., R^2 disparities) among sub-models across road segments, thereby accelerating policy optimization in critical regions. Meanwhile, ε-greedy maintains a level of stochastic exploration at the local segment level to preserve diversity. This strategy effectively balances global feature learning with local heterogeneity adaptation, improving robustness to noise in large-scale networks.

(4) Efficient and stable training optimization techniques. To improve training efficiency and stability, we integrate several key strategies into the DCB model: Experience Replay enhances training stability by reusing historical interaction data; A baseline-based reward formulation reduces variance caused by differences in the inherent prediction difficulty among road segments. A Double Network architecture mitigates the common problem of Q-value overestimation, leading to more stable policy learning.

Experimental results based on data from 4156 urban road segments demonstrate that the proposed DCB framework significantly outperforms traditional clustering-based methods and state-of-the-art single-model approaches in terms of prediction accuracy and robustness to noise. With its fast response and adaptive capabilities, DCB shows strong potential as a scalable tool for real-time urban traffic management, particularly in complex and interference-prone scenarios.

The remainder of this paper is structured as follows. Section 2 reviews related work on road segment traffic flow prediction and contextual bandit learning. Section 3 details the design and algorithmic implementation of the DCB framework. Section 4 presents experimental results and comparative analyses. Section 5 summarizes the findings and outlines directions for future work.

2. Literature Review

This section reviews the relevant literature to contextualize our work. First, we summarize the evolution of traffic flow prediction methodologies, highlighting the shift from single-model approaches to more complex deep learning architectures that attempt to handle spatiotemporal heterogeneity (Section 2.1). Next, we review the application of Reinforcement Learning (RL) in transportation, specifically focusing on Deep Q-Networks (DQN) and key optimization mechanisms—such as experience replay—that inspire our model design (Section 2.2). Finally, we discuss the principles of Deep Contextual Bandits (DCB) and explain how we modify these techniques to construct a dynamic model selection framework suitable for large-scale road networks (Section 2.3).

2.1. Deep Learning for Traffic Prediction

Early road segment traffic flow prediction approaches primarily relied on traditional statistical models (e.g., ARIMA) and classical machine learning methods. However, their limitations in capturing complex spatiotemporal dependencies have driven a shift toward deep learning. Convolutional Neural Networks (CNNs) were among the first deep learning models applied to capture local spatial features. For instance, Song et al. [20] demonstrated through comparative experiments that CNNs reduced prediction errors by 6% compared to multilayer perceptrons (MLPs) in short-term speed forecasting. Liu et al. [21] further enhanced area-level traffic flow prediction by incorporating residual units, thereby improving the model's generalization ability. Subsequently, Recurrent Neural Networks (RNNs) and

their variants—such as LSTM and GRU—gained popularity for their strength in modeling temporal sequences. Bharti et al. [14] introduced a PSO-Bi-LSTM hybrid model, which combines Particle Swarm Optimization (PSO) for global parameter tuning with bidirectional LSTM to capture temporal dependencies in both directions, significantly improving convergence speed and predictive accuracy.

However, conventional CNNs and RNNs struggle to model the non-Euclidean structure of road networks effectively. This limitation spurred the development of Graph Convolutional Networks (GCNs). Chen et al. [22] proposed an enhanced dynamic graph neural network (GNN) that integrates Spatial–Temporal dependency modeling with data interpolation, significantly improving the robustness of network-wide traffic flow prediction. Building upon this, Zheng et al. [23] developed the VDGCNeT model, which innovatively combines virtual dynamic graphs with Transformer architectures to capture dynamic latent relationships between road segments. Graph convolutions are then used to refine spatially fused features, leading to improved predictive performance.

Hybrid architectures and attention mechanisms have become prominent research areas as model complexity increases. Petrović et al. [24] addressed the issue of data sparsity in large-scale road networks by proposing the GCRFBC-GCRF hybrid model, which explicitly models spatial dependencies and outperforms traditional methods in traffic state estimation. Wang et al. [25] introduced PFNet, a cascaded progressive attention framework that balances computational efficiency and predictive performance in large-scale forecasting tasks. Xing et al. [26] proposed the SLTTCN model, which employs spatial linear transformers and bidirectional temporal convolutional networks for efficient large-scale traffic flow prediction.

These advancements illustrate a growing focus on spatiotemporal fusion and self-supervised learning within traffic flow prediction research. While traditional neural network models exhibit strengths in specific prediction scenarios, their performance tends to be unstable in complex and diverse settings. This highlights the urgent need for a dynamic model selection framework that can deliver consistently high accuracy across various large-scale road network conditions.

2.2. Reinforcement Learning in Transportation

The introduction of the Deep Q-learning Network (DQN) marked a significant breakthrough in reinforcement learning, enabling deep reinforcement learning for a wide range of complex decision-making tasks. The core idea of DQN is to approximate the action-value function (Q-function) using a deep neural network (DNN), which predicts the expected cumulative reward $Q(s, a)$ of taking action a given the current state s. Unlike traditional reinforcement learning methods that rely on discrete state-action spaces, DQN leverages the nonlinear representation capabilities of neural networks to directly process high-dimensional, continuous inputs (e.g., images or time-series data), thereby substantially broadening the scope of reinforcement learning applications.

The success of DQN is largely attributed to two key innovations: experience replay and the target network. Experience replay stores agent-environment transitions (i.e., states, actions, rewards, next states) in a replay buffer for training. During training, mini-batches are randomly sampled from this buffer, which breaks the temporal correlation between samples and effectively mitigates issues such as policy oscillation and overfitting. The target network, which shares the same architecture as the main Q-network but is updated at a slower rate, is used to compute the target Q-values $Q'(s', a')$. This reduces the interference of rapidly changing parameters on target estimation, improving training stability.

Many improvements to DQN have been proposed to enhance performance further. For example, Prioritized Experience Replay (PER) [27] adjusts the sampling probability of

experiences based on the temporal difference (TD) error, allowing the model to prioritize learning from high-value samples and accelerate convergence. Double DQN [28] addresses the problem of overestimated Q-values by decoupling action selection and evaluation. These improvements have been successfully applied across various domains, including robot path planning, multi-agent coordination, and traffic flow prediction. For instance, Chen et al. [29] utilized experience replay to improve decision efficiency and robustness in multi-aircraft conflict resolution. In robot navigation, Li et al. [30] adopted PER to quantify the learning potential of experiences using TD-error, further integrating importance sampling to reduce estimation bias. In traffic flow prediction, Nie et al. [31] proposed a hybrid model combining DQN with Generative Adversarial Networks (GANs) to handle traffic big data under COVID-19 conditions. Ullah et al. [32] developed a GCN–Double DQN-based framework to address the dynamic and complex nature of task offloading in vehicular networks.

However, existing DQN-based methods in traffic prediction primarily focus on optimizing a single model—such as improving network architectures or fine-tuning parameters. To the best of our knowledge, limited research has systematically investigated the application of reinforcement learning for dynamic model selection in heterogeneous road segments.

2.3. DCB-Based Dynamic Model Selection

To meet the requirements of traffic flow prediction in large-scale road networks, this study incorporates modular ideas from the Deep Q-learning Network (DQN) to modify the Deep Contextual Bandit (DCB) appropriately, aiming to enhance prediction accuracy and efficiency.

Traditional Multi-Armed Bandit (MAB) algorithms achieve optimal decision-making in dynamic environments by balancing exploration and exploitation. However, conventional MAB approaches struggle to effectively process complex contextual information in high-dimensional state spaces, which limits their applicability in real-world scenarios [33]. To overcome this limitation, the Deep Contextual Bandit (DCB) was introduced. Its core idea is to leverage Deep Neural Networks (DNNs) to automatically learn contextual feature representations and integrate dynamic exploration strategies, such as Upper Confidence Bound (UCB), Thompson Sampling, and probabilistic matching to optimize action selection and efficiently estimate expected rewards [34]. Compared with traditional MAB, DCB significantly improves decision-making efficiency in high-dimensional and nonlinear environments, making it a powerful tool for optimizing large-scale dynamic systems.

Importantly, unlike Reinforcement Learning (RL), which requires learning a state transition model and optimizing for cumulative future rewards, the contextual bandit framework only considers the immediate reward given the current context. This matches our problem setting where the optimal model for a road segment depends primarily on its current traffic pattern, and the selection at one time step does not directly affect future states. This makes DCB more sample-efficient and stable for large-scale, heterogeneous prediction tasks compared to general RL methods.

As shown in Figure 1, the input to the DCB is the contextual state $s \in S$, and the output is the expected reward for each action $a \in A$. Unlike the linear reward models used in traditional MAB, DCB models the reward function using a deep neural network, as shown in Equation (1).

$$R(s, a, \theta) = f_\theta(s, a), \tag{1}$$

where θ represents the network parameters and f_θ denotes the DNN. This nonlinear modeling capability allows DCB to capture complex context–action interactions, thereby enhancing its responsiveness to varying traffic environments.

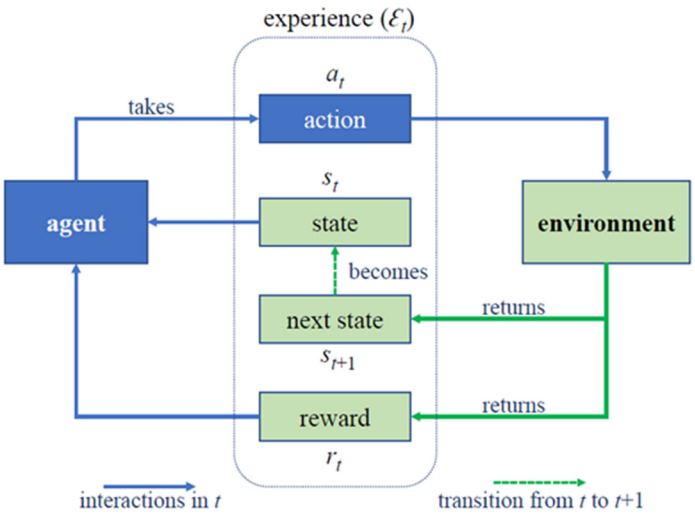


Figure 1. Network architecture of the DCB.

At the heart of DCB lies its exploration strategy, which aims to balance exploration and exploitation. Unlike the simplistic strategies used in traditional MAB, DCB typically adopts uncertainty-aware exploration mechanisms:

1. Upper Confidence Bound (UCB): Estimates an upper confidence bound for the expected reward of each action and selects the action with the highest bound. This approach encourages the exploration of actions with high potential but uncertain reward estimates.
2. Thompson Sampling: Maintains a posterior distribution over the reward function. At each step, it samples a parameter instance from this distribution and selects the action with the highest expected reward under that sample. This naturally balances exploration and exploitation.
3. Other Probabilistic Matching Strategies: These sample actions are based on the probability that each action is currently optimal, according to the estimated reward distribution.

Compared to random exploration strategies, these uncertainty-driven approaches exploit the learned feature representations and uncertainty information from the deep network, enabling more efficient exploration in high-dimensional contextual spaces. While the standard ε-greedy strategy [35] is simple and widely used, its random nature often leads to inefficiency in the DCB framework. It is typically replaced or augmented by a more intelligent strategy based on uncertainty.

In summary, the traditional ε-greedy strategy exhibits poor scalability for large-scale road networks due to inefficient exploration. Although UCB and Thompson Sampling [36] improve the intelligence of exploration, they still face challenges regarding computational complexity and implementation in high-dimensional traffic state spaces. To address this, we propose a hybrid exploration strategy that combines Priority Sampling with ε-greedy, aligning with hierarchical exploration. Priority Sampling is employed at the macro level to identify which road segments to explore. Moreover, a small amount of ε-greedy is retained at the micro level to ensure that potentially high-performing models are not overlooked.

By dynamically allocating learning resources based on quantified differences in model performance, this approach is better suited for learning the diverse characteristics of heterogeneous road segments. It offers a novel and efficient exploration paradigm for large-scale traffic flow prediction.

To bridge the research gap in dynamic model selection and leverage these advantages, the DCB framework proposed in this paper draws inspiration from DQN's experience replay and dual-network mechanisms, while introducing novel enhancements. Specifi-

cally, we redefine the action space to represent the model selection problem and design a context-aware dynamic decision mechanism. This enables adaptive selection of the optimal prediction model for each road segment based on its real-time traffic flow characteristics, thereby significantly improving the accuracy and robustness of traffic prediction across large-scale road networks.

3. Model

3.1. Deep Contextual Bandit (DCB) Model

3.1.1. Model Design Motivation and Overall Framework

The core idea behind the Deep Contextual Bandit (DCB) model is to utilize deep neural networks to learn a mapping between the contextual traffic flow characteristics of a given road segment and the predictive performance of various forecasting models. Based on this learned relationship, the DCB framework dynamically selects the most suitable prediction sub-model for each road segment. As illustrated in Figure 2, the input to the DCB model consists of contextual features of the target road segment, which capture its temporal and spatial traffic patterns. The model estimates the expected predictive performance (i.e., expected reward) of each candidate sub-model under the given contextual conditions. The sub-model with the highest expected reward is then selected to generate the final traffic flow prediction for that segment. This adaptive model selection mechanism enables the DCB framework to effectively tailor its predictions to diverse and dynamic traffic scenarios across a large-scale road network, thereby significantly enhancing both accuracy and robustness compared to static, single-model approaches. This capability to handle contextual heterogeneity aligns with the broader need in intelligent transportation systems for models that maintain consistent performance under varying conditions—a challenge also highlighted in related tasks such as vision-based crash detection across day-night scenes [37].

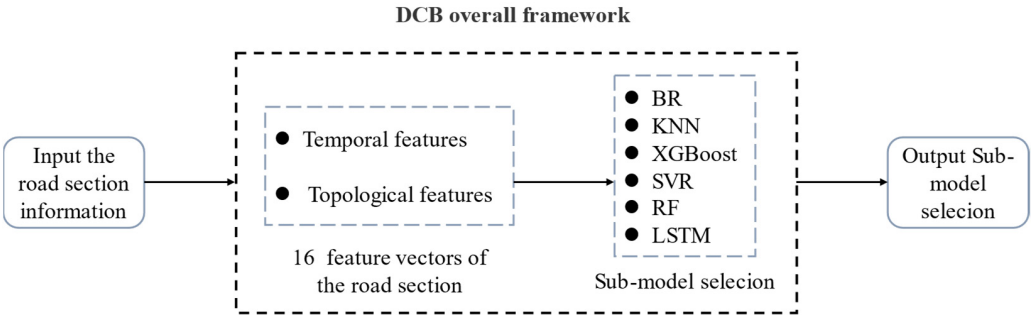


Figure 2. Framework diagram for model selection.

From a theoretical perspective, the contextual bandit framework provides regret guarantees under stationary environments, ensuring that the learning process converges to the optimal model selection policy. In non-stationary traffic conditions—such as those caused by events, holidays, or long-term trends—DCB's exploration mechanisms (e.g., the hybrid ε-greedy strategy with Priority Sampling) enable continuous adaptation. Although establishing formal regret bounds is more challenging for non-stationary bandits, our empirical design incorporates experience replay and periodic retraining to mitigate performance degradation under gradual distribution shifts. Consequently, DCB is well-suited for real-world traffic networks characterized by both short-term fluctuations and long-term evolution.

Fundamentally, DCB employs a deterministic 'hard' selection strategy, unlike the soft weighted fusion in Mixture-of-Experts (MoE). This approach not only reduces inference latency but also enhances interpretability by explicitly identifying the single most effective model for a given traffic context.

3.1.2. State-Action-Reward Ternary Modeling

To implement the model selection mechanism of the DCB framework, this paper formalizes the traffic flow prediction task as a contextual bandit problem, which is defined by a core ternary structure consisting of state, action, and reward. In each decision-making round, the model receives a context vector, which encodes the current environmental state (e.g., traffic flow patterns of the target road segment). Based on this context, the model selects and executes an action, corresponding to choosing one of the available prediction sub-models. The system then provides a reward value based on the predictive performance of the selected model, which is used to update the decision policy.

The key components of the contextual bandit formulation are illustrated in Figure 3. In this framework, the State s represents the environmental context which serves as the input for the decision policy to determine the Action (a). Crucially, the reward r is modeled as a function of both the state s and action a ($r \sim \mathcal{D}(s, a)$). This formulation explicitly captures that a sub-model's efficacy is intrinsically tied to the prevailing traffic conditions (state). The agent utilizes this state-dependent reward feedback to optimize its policy.



Figure 3. State–action–revenue triad structure.

1. State (state,s)

The state represents the context of traffic features used to describe the current road section. In this paper, 16-dimensional features are extracted, including the following three categories.

- Autocorrelation Coefficient

To portray the time-dependence of the traffic flow, this paper calculates the Autocorrelation Coefficient of the road section under different time lags (lag). Let the time series of traffic flow be x_t, and the autocorrelation coefficient of its lag is defined in Equation (2).

$$\rho_l = \frac{\sum_{t=1}^{T-l}(x_t - \overline{x})(x_{t+l} - \overline{x})}{\sum_{t=1}^{T}(x_t - \overline{x})^2}, \tag{2}$$

where $\overline{x}$ is the sample mean. The lagged value $l \in \{1, 2, 3, 4, 8, 12, 96, 672\}$ is selected.

- Statistical distribution characteristics

Skewness $Skew$, kurtosis $Kurt$, mean μ, standard σ of traffic flow; these indicators can assist the model to determine the volatility of traffic flow and abnormal behavior, as shown in Equations (3)–(6).

$$\mu = \frac{1}{T}\sum_{t=1}^{T} x_t, \tag{3}$$

$$\sigma = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(x_t - \mu)^2}, \tag{4}$$

$$Skew(x) = \frac{1}{T}\sum_{t=1}^{T}\left(\frac{x_t - \mu}{\sigma}\right)^3, \tag{5}$$

$$Kurt(x) = \frac{1}{T}\sum_{t=1}^{T}\left(\frac{x_t - \mu}{\sigma}\right)^4, \tag{6}$$

where x_t is the observed traffic volume at time step t; T is the total length of the time series; μ is the mean of the traffic flow; and σ is the standard deviation of the traffic flow.

- Topological Features

These include Degree Centrality C_D, Clustering Coefficient C, Betweenness Centrality C_B, and Eigenvector Centrality C_E. The formulas are Equations (7)–(10).

$$C_D(i) = \frac{k_i}{N-1} \tag{7}$$

$$C(i) = \frac{2E_i}{k_i(k_i-1)} \tag{8}$$

$$C_B(i) = \sum_{s \neq i \neq t} \frac{\sigma_{st}(i)}{\sigma_{st}}, \tag{9}$$

$$C_E(i) = \frac{1}{\lambda} \sum_{j \epsilon N(i)} A_{ij} C_E(j), \tag{10}$$

where N is the total number of nodes in the network; k_i is the degree of a node i, i.e., the number of edges connected to it; E_i represents the number of actual edges between the neighboring nodes of node i; σ_{st} is the total number of shortest paths between node s and node t; $\sigma_{st}(i)$ is the number of those shortest paths that pass through node i; A_{ij} is the element of the adjacency matrix, if node i is connected to node j, then $A_{ij} = 1$, otherwise it is 0; λ is the eigenvalue corresponding to the largest eigenvalue; $C_E(i)$ is the eigenvector centrality score of node i.

By integrating these three dimensions—temporal dependencies, statistical distribution patterns, and topological structures—we construct a comprehensive 16–dimensional feature vector that fully characterizes the traffic context of each road segment. This multi-view representation enables the agent to distinguish between different traffic scenarios effectively. The detailed composition and categorization of these contextual features are summarized in Table 1.

Table 1. Summary of the 16 Contextual Features.

Category	Feature Name	Description/Parameters	Count
Temporal Features	Autocorrelation Coefficients	Calculated at different time lags $l \in \{1, 2, 3, 4, 8, 12, 96, 672\}$ to capture short-term and long-term temporal dependencies.	8
Traffic State Features	Statistical Distribution Metrics	Includes Mean (μ), Standard Deviation (σ), Skewness (*Skew*), and Kurtosis (*Kurt*) of the historical traffic flow sequence.	4
Network Topology Features	Graph Structural Indicators	Includes Degree Centrality (C_D), Clustering Coefficient (C), Betweenness Centrality (C_B), and Eigenvector Centrality (C_E) to describe the node's importance in the road network.	4
Total			16

2. Action (action, a)

In the DCB framework proposed in this paper, the action set consists of multiple candidate prediction models, denoted as $\{M_1, M_2, ..., M_N\}$. At each decision step, DCB evaluates the prediction performance (i.e., action value) of each candidate model on the

current road segment based on its contextual features, and selects the model with the highest action value as the prediction model for that segment.

3. Reward (reward, r)

In traffic flow prediction tasks, commonly used error metrics such as Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) have certain limitations. First, these metrics lack standardized reward signals, which reduces their comparability across different scenarios. For instance, the absolute values of RMSE and MAE tend to fluctuate significantly during peak periods, holidays, or under weather-related disturbances, making it challenging to assess model performance over time consistently. Second, traditional metrics, especially RMSE, are highly sensitive to noise, and in the presence of unexpected events, significant error points can be disproportionately amplified, thereby distorting the overall evaluation. While RMSE and MAE can offer insights into localized prediction errors, they often fail to effectively capture cyclical patterns, trends, and structural characteristics inherent in traffic flow data.

In contrast, when used as the reward signal, the coefficient of determination (R^2) effectively addresses these issues. It provides a standardized gain signal by measuring how well the model explains the overall variance in the data, enabling more consistent model evaluation across diverse scenarios. R^2 also reflects trends and cyclical variations in traffic flow, making it better suited for complex and dynamic environments.

Therefore, this paper adopts R^2 as the reward metric in the DCB framework, and it is defined in Equation (11).

$$R^2 = 1 - \frac{SSE}{SST} = 1 - \frac{\sum_{t=1}^{T}(x_t - \hat{x}_t)^2}{\sum_{t=1}^{T}(x_t - \overline{x})^2}, \tag{11}$$

where SSE is the sum of the squares of the gap (error) between the model's predictions and the actual values, representing the degree of model error, and SST is the sum of the squares of the degree of volatility of the actual values themselves, representing the degree of complexity of the data itself. $\hat{x}_t$ is the predicted value.

3.2. Improvement Strategies

To enhance the stability, robustness, and learning efficiency of the Deep Contextual Bandit (DCB) model in traffic flow prediction tasks, this paper proposes several key optimization strategies, including the experience replay buffer, relative gain design, prioritized sampling mechanism, and dual-network architecture. The following sections elaborate on each strategy in detail.

3.2.1. Experience Replay Buffer

Traditional contextual bandit algorithms typically update model parameters based only on the state–action–reward (gain) triplet obtained during a single round of interaction with the environment, after which the data from that round are discarded. This approach severely limits the reuse of historical experience, resulting in low sample utilization. As shown in Figure 4, to fully exploit the potential value of past experiences, this paper introduces an Experience Replay Buffer (denoted as D) [38]. The buffer continuously accumulates the state–action–reward triplets generated during the interaction between the model and the environment and repeatedly samples them for training. This significantly improves sample reuse efficiency. Moreover, the experience replay mechanism breaks the temporal correlation between samples by enabling random sampling from the historical experience pool. This contributes to balancing bias and variance during policy updates, enhancing training stability and reducing the risk of oscillations and overfitting in the learning process.

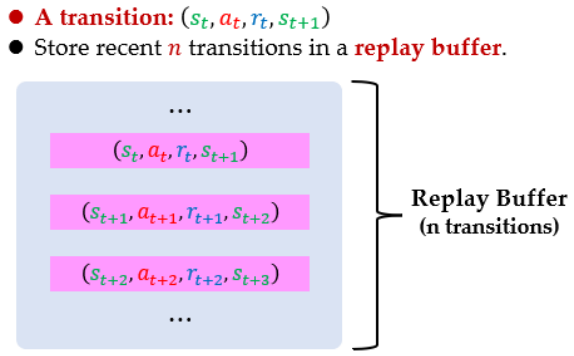


Figure 4. Experience Replay Mechanism in Deep Q-Networks.

Unlike classical deep reinforcement learning algorithms (e.g., DQN), this study does not impose a maximum storage limit on the experience replay buffer. This design is based on the observation that traffic flow prediction is not a sequential decision-making problem, and the historical experiences are generally independent. In contrast to sequential decision tasks—where redundancy and interference in the experience buffer can degrade performance—in traffic flow prediction, the accumulation of historical samples does not introduce a significant negative impact. This approach helps to maximize data utilization by ensuring that every interaction experience can be fully leveraged during subsequent policy optimization.

If the experience buffer $\mathcal{D}$ has not yet reached a predefined minimum capacity threshold during training, the model is trained using only the newly collected experiences from the current batch. Once the number of stored experiences meets the threshold, the training process switches to Prioritized Sampling based on the historical samples in the buffer, and a mini-batch training strategy is adopted to enhance training efficiency and improve the model's generalization ability.

3.2.2. Baseline Model

Although the R^2 metric is effective for evaluating model fitting performance, it can exhibit considerable fluctuations during training due to significant differences in traffic flow periodicity, volatility, and data quality across different road segments. For instance, segments with more predictable and regular patterns tend to yield higher R^2 values. In contrast, other segments, affected by noise or greater variability, may show lower R^2 values, even when the predictions are reasonably accurate. In the Deep Contextual Bandit (DCB) learning process, if the same candidate model produces highly variable prediction results across different road segments, the sample variance increases, which may negatively affect the convergence rate of the learning algorithm. To address this issue, this paper introduces a relative gain mechanism by incorporating the prediction results of a baseline model, thereby reducing sample variance. The relative gain is defined in Equation (12).

$$r = R^2_{action} - R^2_{baseline}, \tag{12}$$

where R^2_{action} denotes the actual R^2 value (goodness-of-fit) of the currently selected model, and $R^2_{baseline}$ represents the R^2 value of the baseline model under the same contextual state. The baseline model is typically selected from the candidate models that exhibit relatively stable performance and faster training speed.

By providing a consistent and representative predictive reference, the baseline model helps mitigate prediction accuracy fluctuations caused by differences in road segment characteristics. This allows the DCB model to more accurately assess the relative predictive performance of each candidate model for the same prediction target.

3.2.3. Priority Sampling

To enhance training efficiency, this paper introduces a Priority Sampling strategy. The core idea is to prioritize experiences that exhibit significant differences in model performance, thereby accelerating the learning process for critical road segments. The strategy involves the following steps:

1. Priority Calculation

For each newly generated experience m, DCB evaluates the action values Q of all candidate models. To avoid the situation where poor performance by certain models across most road segments diminishes the distinctiveness of the priority scores, the top K highest action values are selected, denoted as $\{Q_1, Q_2, ..., Q_K\}$. The priority of experience m is then defined as the average pairwise difference among these values, as shown in Equation (13).

$$Priority_m = \frac{1}{K(K-1)/2} \sum_{1 \le i \le j \le k} |Q_i - Q_j|. \tag{13}$$

A higher priority indicates greater variance in the predictive performance of different models on the given segment, suggesting higher learning value. Conversely, model selection becomes less meaningful for that segment if model performances are similar.

2. Sampling Probability

Based on the calculated priority scores of all experiences, the sampling probability of experience m is defined in Equation (14).

$$P_m = \frac{Priority_m}{\sum_{m' \in \mathcal{D}} Priority_{m'}}, \tag{14}$$

where $\mathcal{D}$ denotes the experience replay buffer.

3. Dynamic Update Mechanism

To account for model instability in the early stages of training, the priority of each sampled experience $m \, \epsilon \, \mathcal{D}$ is recalculated using the current policy network after it is selected. This dynamic update ensures that the sampling strategy remains focused on the most valuable experiences under the current state of the model.

3.2.4. Double Network Structure (Double DQN)

To enhance model stability and reduce the overestimation problem during training, this paper draws on the Double DQN architecture and introduces a dual-network strategy. This strategy consists of: A policy network with parameters θ, which estimates the state-action value $Q(s, a; \theta)$ based on the current state; A target network with parameters θ', which periodically copies parameters from the policy network and is used to compute the target value $Q'(s, a; \theta')$.

This design avoids estimation bias caused by feedback from the same network during training. Since the DCB task does not involve subsequent states, this paper adopts an action selection method that averages the outputs of the two networks, is formulated in Equation (15).

$$\overline{Q}(s, a) = \frac{Q(s, a; \theta) + Q'(s, a; \theta')}{2} \tag{15}$$

This strategy reduces Q-value overestimation, thereby improving training stability and effectively diminishing fluctuations during the training phase.

3.3. Model Training Framework

The training framework of the Deep Contextual Bandit (DCB) model proposed in this paper is illustrated in Figure 5. The overall process alternates between interaction and training phases to achieve continuous model performance optimization.

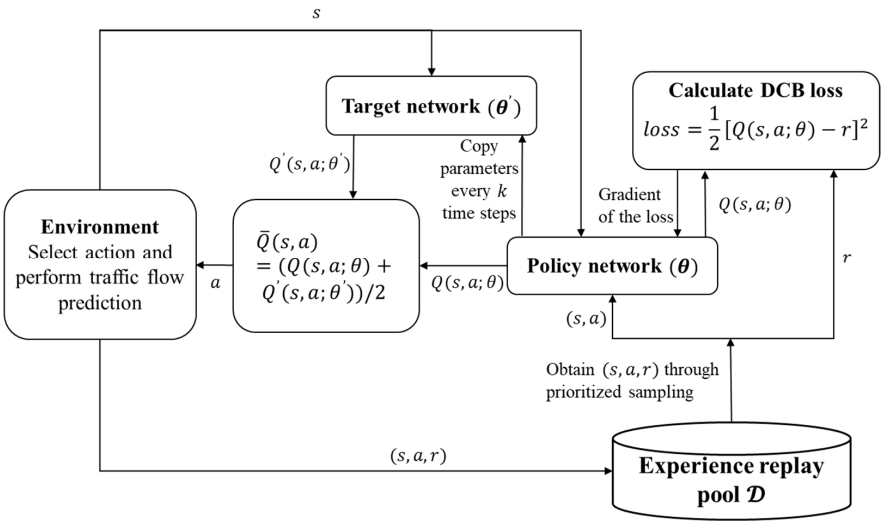


Figure 5. Flowchart of the proposed algorithm.

During each interaction round, the system integrates the improvement strategies into the framework. First, the environment generates the current context state s (i.e., the feature vector of the target road segment). Unlike state transitions in traditional MDPs, here s serves strictly as the input condition for the policy and target networks to determine the optimal action, while the reward r is used exclusively for loss calculation. Subsequently, the DCB system evaluates the state-action values of candidate prediction models under the current state s using two independent neural networks: the policy and target networks. The policy network's parameters are continuously updated, while the target network's parameters are periodically synchronized from the policy network to enhance prediction stability.

Based on the ϵ-greedy strategy, the system selects an action a—corresponding to a specific candidate traffic flow prediction model—and performs prediction on the current data window. To explicitly quantify the model's performance, the reward r is calculated as the Coefficient of Determination (R^2) for the current prediction horizon:

$$r = 1 - \frac{\sum_{t=1}^{T}(y_t - \hat{y}_t)^2}{\sum_{t=1}^{T}(y_t - \overline{y})^2} \tag{16}$$

where y_t is the ground truth traffic flow, $\hat{y}_t$ is the prediction from the selected model, and $\overline{y}$ is the mean of the observed values in the current window. This reward r, along with the state s and action a, is stored in the experience replay buffer $\mathcal{D}$.

In the training phase, the reward r serves directly as the training target to guide network optimization. Specifically, we define the loss function $L(\theta)$ as the expected squared error between the estimated Q-value and the observed reward:

$$L(\theta) = \mathbb{E}_{(s,a,r)\sim D}\left[(r - Q(s,a;\theta))^2\right] \tag{17}$$

Here, since the Contextual Bandit problem assumes no state transition (discount factor $\gamma = 0$), the standard Bellman target simplifies to the immediate reward r. To improve sample efficiency, the model applies the Prioritized Experience Replay strategy. The sampling priority is computed based on the Temporal Difference (TD) error $\delta = |r - Q(s,a;\theta^{-})|$,

emphasizing experiences where the model's current estimation deviates significantly from the actual performance (R^2). Through continuous iteration of this process, the DCB model progressively learns to map context states to the optimal model selection strategy.

The end-to-end training process of the DCB framework is detailed in Algorithm 1. It operates in an iterative loop of interaction and training phases. Key components integrated into this process include: (1) the hybrid exploration strategy for segment and action selection; (2) the dual-network (policy and target) architecture for stable Q-value estimation; (3) the experience replay buffer $\mathcal{D}$ for sample reuse; and (4) the priority sampling mechanism for efficient learning. The target network parameters θ' are synchronized with the policy network parameters θ every $C = 100$ training steps to stabilize the learning target. The algorithm terminates when the average reward (R^2) on a held-out validation set plateaus over a predefined number of epochs.

Algorithm 1. Training Procedure of DCB Framework

Input: Road segment dataset $\mathcal{R}$, candidate model pool $\{M_1, \ldots, M_N\}$, hyperparameters (Section 4.2.5), experience replay buffer $\mathcal{D}$ (initially empty), priority cache.
Output: Trained policy network parameters θ.

Initialize policy network $Q(\cdot;\theta)$ and target network $Q'(\cdot;\theta')$ with weights $\theta' = \theta$.
Pre-train candidate models and perform feature clustering as per Section 4.2.4.
for episode = 1 to E do
    // --- Interaction Phase ---
    Select a road segment s:
        With probability β, use Priority Sampling Equation (14) to select s from $\mathcal{D}$.
        Otherwise, select s uniformly from $\mathcal{R}$.
    Extract contextual feature vector $\phi(s)$ (Section 3.1.2).
    // Action selection with ϵ-greedy
    With probability ϵ: select a random action (model) a.
    Otherwise: $a = \arg max_{a'} \overline{Q}(\phi(s), a')$ where $\overline{Q}$ is from Equation (15).
    Execute model M_a to predict traffic flow for segment s.
    Observe reward $r = R^2_{M_a} - R^2_{baseline}$ Equation (12).
    Store transition $(\phi(s), a, r)$ in replay buffer $\mathcal{D}$.
    Calculate and store priority for this transition using Equation (13).

    // --- Training Phase ---
    if $|\mathcal{D}| \geq$ minimum batch size then
        Sample a mini-batch of B transitions from $\mathcal{D}$ using priority probabilities P_m Equation (14).
        for each sampled transition $\left(\phi\left(s_j\right), a_j, r_j\right)$ do
            // Compute target Q-value (simplified for contextual bandit)
            $y_j = r_j$
            // Update priority based on current prediction
            Recalculate $Priority_j$ using current $Q(\cdot;\theta)$ and Equation (14).
        end for
        // Update policy network via gradient descent
        Perform a gradient step on θ to minimize $\frac{1}{B}\sum_j \left(y_j - Q\left(\phi\left(s_j\right), a_j; \theta\right)\right)^2$.
        // Periodically update target network
        if episode mod $C == 0$ then
            $\theta' \leftarrow \theta$
        end if
    end if
end for

4. Experiments

4.1. Dataset

This study utilizes a large-scale real-world traffic flow dataset collected from the Intelligent Transportation System (ITS) of City D, China. The raw data were captured by fixed loop detectors covering 4156 road segments across the city's extensive expressway network. The traffic volume records were aggregated at 15 min intervals, spanning a four-month period from 23:45 on 26 November 2023, to 23:45 on 26 March 2024. This specific time window is significant as it encompasses major public holidays, including New Year's Day and the Spring Festival (Chinese New Year). Including these periods allows for the evaluation of the model's robustness under complex, non-stationary traffic patterns caused by holiday travel surges and demand fluctuations.

Data preprocessing was first conducted to ensure data quality. Raw data were cleaned to remove outliers and abnormal records, and missing values were addressed using a historical mean-imputation strategy, where missing entries were replaced with the average flow of the corresponding time-of-day from other days for the same road segment. To enhance training stability, all traffic flow data were normalized to the range $[0, 1]$. Unlike approaches relying on external weather or event data, the DCB framework extracts intrinsic contextual features to implicitly capture environmental variations without relying on external auxiliary data. Specifically for the performance verification of the DCB model, 100 representative road sections were randomly selected from the test set as the primary observation objects.

For the evaluation of candidate sub-models and the overall training process, the dataset was chronologically partitioned into a training set (the first 80%) and a test set (the subsequent 20%). Specifically, the training phase covers the period from 26 November 2023, to 29 February 2024, while the testing phase spans from 1 March 2024, to 26 March 2024. A sliding window approach was employed, utilizing historical traffic flow data from the past 6 h (24 time steps) to predict the traffic volume for the next 15 min interval. To ensure statistical reliability, all experiments were repeated five times.

4.2. Experimental Setup

All experiments were conducted on a workstation running Windows 10 Professional 64-bit, equipped with an Intel Xeon W-2245 processor (3.90 GHz, 16 threads), 32 GB of RAM, and an NVIDIA Quadro RTX 4000 graphics card with 8 GB of video memory. The experimental programs were implemented in Python 3.11, with the deep learning framework primarily based on PyTorch 1.13.1.

4.2.1. Evaluation Metrics

While the ultimate goal of the Deep Contextual Bandit (DCB) model is to reduce prediction error, R^2 is used as the training objective in this paper. To comparatively evaluate the effectiveness of DCB in reducing traffic flow prediction errors across large-scale road networks, two commonly used error metrics are introduced: MAE and RMSE.

1. Mean Absolute Error (MAE)

$$MAE = \frac{1}{T}\sum_{t=1}^{T}|x_t - \hat{x}_t|. \tag{18}$$

2. Root Mean Squared Error (RMSE)

$$RMSE = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(x_t - \hat{x}_t)^2}. \tag{19}$$

MAE measures the average absolute size of the error between the predicted value and the real value, which is insensitive to outliers and suitable for scenarios with a more uniform distribution of errors. RMSE represents the square root of the prediction error, with the same unit as the original data, which is more sensitive to larger errors and reflects the model's prediction accuracy more clearly.

4.2.2. Selection of Baseline Model

This study considers computational efficiency and prediction stability when selecting the baseline prediction model. Computational efficiency relates to training speed, while stability affects the reliability of reward calculation. The XGBoost (version 1.7.5) model, which achieves a good balance between these two aspects, is ultimately chosen as the baseline model to calculate the reward in the DCB framework.

4.2.3. Action Set Models

The action set constructed in this paper consists of six classical traffic flow prediction models: Bayesian Regression (BR), K-Nearest Neighbors (KNN), XGBoost, Support Vector Regression (SVR), Random Forest (RF), and Long Short-Term Memory networks (LSTM). The architectures of KNN, SVR, DNN, and LSTM are based on the designs in Salamanis et al. [39], while Bayesian Regression and Random Forest use the standard implementations provided by scikit-learn. These models are shown in Table 2. This combination balances model diversity, generalization capability, and practical applicability for large-scale road segment prediction tasks, offering a rich set of candidates for strategy optimization.

Table 2. Six classical forecasting models.

Model Category	Model Name	Brief Description
Traditional Statistical Method	Bayesian Regression (BR)	A linear model based on Bayesian inference that can quantify prediction uncertainty. It is suitable for constructing baseline models and is implemented using scikit-learn.
Machine Learning Methods	K-Nearest Neighbors (KNN)	A prediction method based on the distance between neighboring samples. It is suitable for modeling local patterns but is sensitive to high-dimensional data.
	Support Vector Regression (SVR)	A kernel-based method for modeling nonlinear relationships, offering strong generalization capabilities.
	eXtreme Gradient Boosting (SVR)	An ensemble learning model capable of handling complex nonlinear relationships between features. It is known for its high training efficiency and stable performance.
	Random Forest (RF)	An ensemble method based on multiple decision trees, offering high robustness and noise resistance. Implemented using scikit-learn.
Deep Learning Method	Long Short-Term Memory (LSTM)	A neural network model capable of capturing long-term dependencies in time series data. It is well-suited for traffic flow prediction tasks with clear periodic or trend patterns.

4.2.4. Prediction Model Parameter Candidates

Due to the high computational cost of tuning hyperparameters for each road segment, this study introduces a clustering mechanism [16] to improve the efficiency of hyperparameter training. The specific procedure is as follows.

1. Feature Clustering. First, K-Means clustering is applied to the features of each road segment. The goal is to group road segments with similar traffic dynamic characteristics into the same cluster. The optimal number of clusters is estimated using the elbow method, resulting in 14 clusters. Based on this clustering, hyperparameter configurations are determined for each model within each cluster.
2. Representative Road Segment Selection. For each cluster, the road segment closest to the cluster center is selected as the representative segment. This segment is used for subsequent sub-model training and hyperparameter tuning. This approach captures diverse traffic flow patterns, enhances model generalization, and prevents the training samples from being overly concentrated in a specific pattern, effectively reducing the hyperparameter search space while maintaining diversity.
3. Grid Search Hyperparameter Tuning. For the representative road segments identified above, grid search is employed to optimize the hyperparameters of the six candidate prediction models (BR, KNN, XGBoost, SVR, RF, and LSTM) separately. After obtaining the optimal hyperparameter sets for each cluster, the tuned parameters are uniformly assigned to the corresponding models for all road segments within that cluster.

The primary hyperparameters and their quantities for each prediction sub-model are listed below. The tuned parameters after training are provided in Table 3. For the complete set of parameter values across all cluster groups, see Table A1 in Appendix A.1.

Table 3. Nomenclature of model hyperparameters.

Model	Hyperparameters
BR	$\alpha 1$: prior on the precision of the weights, $\alpha 2$: prior on the precision of the noise, $l1$: regularization on the weights, and $l2$: regularization on the noise.
KNN	n: number of neighbors, w: weighting function, and algo: neighbor search algorithm.
XGboost	md: maximum tree depth, lr: learning rate, s: subsample ratio, and col: feature subsample ratio per tree.
SVR	c: regularization parameter, and e: epsilon parameter.
RF	nt: number of trees, md: maximum tree depth, and ms: minimum samples for split.
LSTM	ep: number of epochs, and lr: learning rate

4.2.5. Deep Contextual Bandit Model Hyperparameters

Directly involving all road segments in the hyperparameter search of the Deep Contextual Bandit (DCB) model would result in a substantial computational workload. To address this issue, we extend the aforementioned clustering approach: only the representative road segments derived from clustering are used in the hyperparameter tuning process. A comprehensive grid search is then performed to explore all possible combinations of hyperparameters. The final set of hyperparameters for the DCB model is presented in Table 4.

Table 4. Optimal hyperparameter values for DCB.

Hyperparameters	Value
Loss function	Mean Squared Error
Activation function	Relu
Number of layers	4
Number of neurons per layer:	128
Batch size	128
learning rate	0.0001
number of epochs	5000
Optimizer	Adam
K (Parameters in Priority Sampling)	3

These hyperparameters demonstrated good training stability and convergence across representative road segments from multiple clustering groups, thereby providing a reliable foundation for the subsequent learning and generalization of the DCB strategy.

4.3. Experimentation and Analysis

The computational performance of the proposed DCB framework was evaluated on a workstation equipped with an Intel Xeon W-2245 processor and an NVIDIA Quadro RTX 4000 GPU. In this setup, the total training process required approximately 27.07 h. Regarding resource consumption during the inference phase, the model demonstrated high efficiency, recording a peak GPU memory usage of 151.26 MB and a CPU memory increase of 1152.69 MB. Most importantly, the average inference latency was maintained at 12.6 ms per road segment, confirming that the framework is computationally lightweight and well-suited for real-time traffic flow prediction in large-scale networks.

4.3.1. Model Convergence and Prediction Performance

To evaluate the performance of the Deep Contextual Bandit (DCB) model in traffic flow prediction tasks, this study compares its performance on the test set using three key metrics: the coefficient of determination (R^2), MAE, and RMSE. Figure 6 presents the aggregated trajectories of these three metrics during training and contrasts them with six sub-models: Bayesian Regression (BR), KNN, XGBoost, SVR, Random Forest (RF), and LSTM.

In Figure 6, the black line represents the performance of the DCB model; The red horizontal line denotes the theoretical limit—i.e., the performance achieved by the optimal sub-model for each road segment in the test set; The colored horizontal lines indicate the performance achieved when using each corresponding sub-model alone across all segments.

1. R^2 Performance: The red horizontal line indicates the theoretical upper bound of R^2. The DCB model exhibits strong convergence around the 1100th training iteration, with its R^2 value rapidly approaching this upper bound and remaining stable thereafter. This performance is significantly better than any individual sub-model, suggesting that DCB effectively selects more appropriate sub-models under different contextual conditions, thereby enhancing overall predictive accuracy.
2. MAE Performance: The red line represents the theoretical lower bound for MAE. The MAE of the DCB model decreases steadily during training and eventually stabilizes near this lower bound, indicating strong error control capability. In comparison, other models exhibit higher MAE values and less stable behavior, lacking precision and robustness.
3. RMSE Performance: The red line shows the theoretical RMSE lower bound. The DCB model again demonstrates notable advantages, with rapid convergence and lower

error values. In the later stages of training, its RMSE stabilizes close to the theoretical minimum, further confirming its predictive accuracy and consistency.

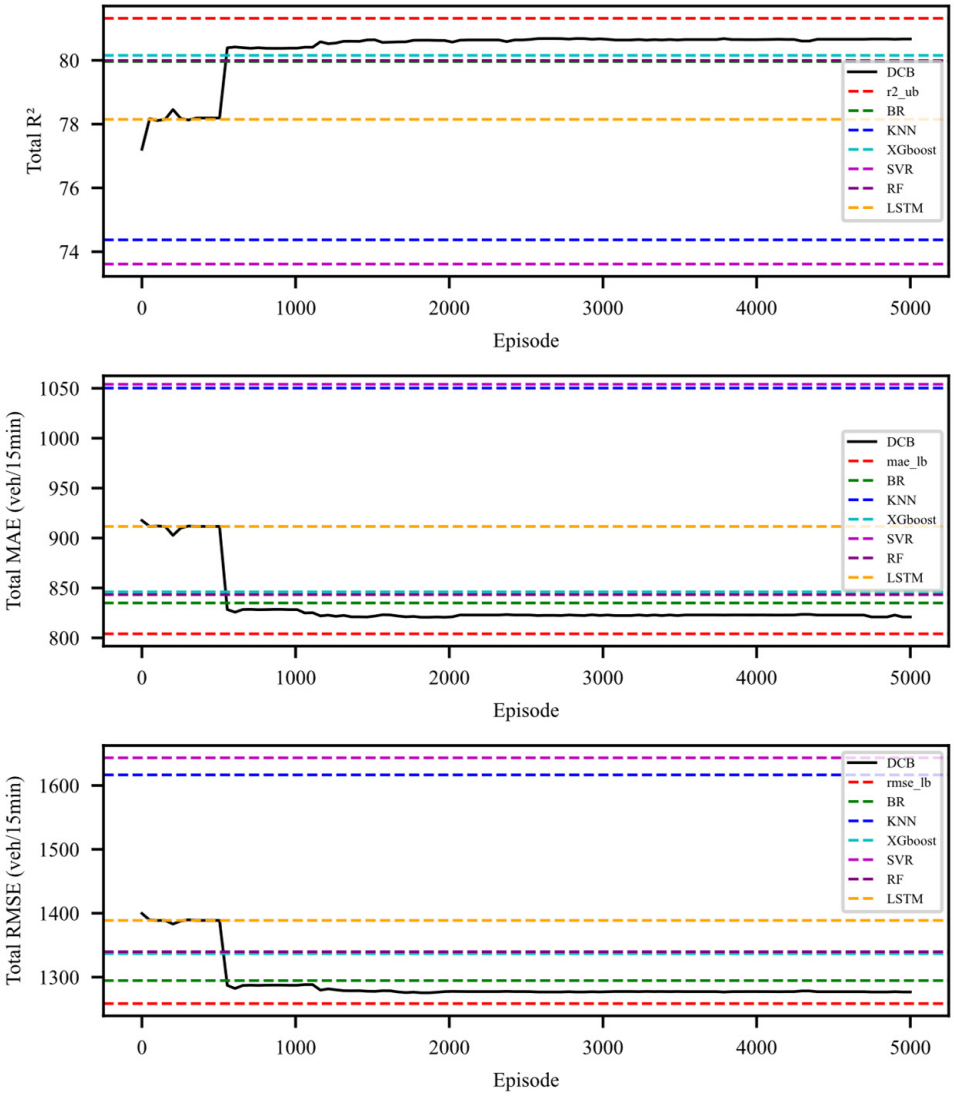


Figure 6. Training process of DCB.

Empirically, the convergence behavior observed in Figure 6 indicates that DCB achieves stability after approximately 1300 iterations, closely approaching the theoretical upper bound. To further assess robustness under non-stationarity, we analyzed the cumulative regret (Figure 7)—defined as the accumulated difference between the optimal and realized rewards. The observed sub-linear growth of the cumulative regret confirms that the learning algorithm effectively converges to the optimal policy over time. Notably, the testing phase specifically coincided with the Spring Festival, introducing severe non-stationarity through abrupt traffic surges. Even under these extreme conditions, the model maintained an R^2 score of approximately 80%, further validating the reliability of the bandit learning process against distributional shifts.

In conclusion, although the reward function used during training is based solely on R^2, the DCB model also substantially improves other error metrics. This indicates that its strategy learning process not only enhances R^2 performance but also effectively balances the control of prediction errors such as MAE and RMSE, showcasing excellent multi-metric performance and strong generalization ability.

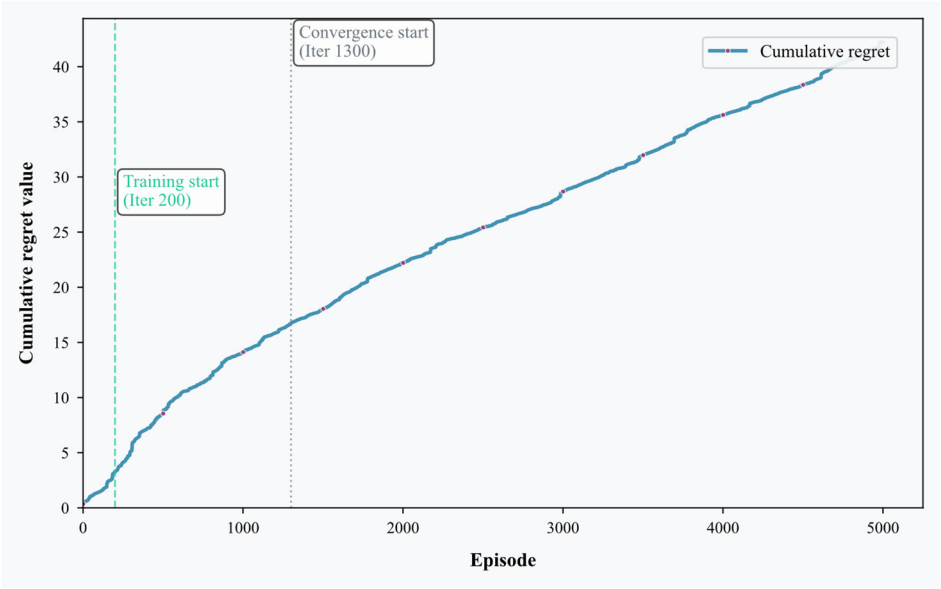


Figure 7. Cumulative regret curve of DCB.

4.3.2. Accuracy of Action Selection in DCB

To evaluate model selection accuracy (i.e., action selection) within the Deep Contextual Bandit (DCB) framework, we introduce a tolerance threshold T, recognizing that in practical scenarios, the prediction performances of different models on the same road segment can be very close. Small fluctuations in R^2 values should not be overemphasized when judging the correctness of model selection. Therefore, when the difference (gap) in R^2 values between two models for a given segment is below the threshold T, their predictive performances are considered equivalent, and any model selected from this group is deemed an acceptable choice.

We define the R^2 scores of two candidate models M_i and M_j on the same road segment as $R^2_{M_i}$ and $R^2_{M_j}$, respectively, where $i, j \leq K$. The gap is calculated in Equation (21).

$$gap_{M_i,M_j} = \frac{\left|R^2_{M_i} - R^2_{M_j}\right|}{\max\left(R^2_{M_i}, R^2_{M_j}\right)} \times 100\%. \tag{20}$$

In our experiments, we explore four threshold levels: $T \in \{2\%, 3\%, 4\%, 5\%\}$, to assess how model selection accuracy changes with varying degrees of tolerance. This approach prevents minor performance differences from misleading the evaluation of the DCB's decision-making accuracy and ensures a fairer assessment of its model selection strategy.

Figure 8 illustrates how the action selection accuracy of the DCB model evolves with the number of training iterations under different gap thresholds. As training progresses, the accuracy steadily increases across all values of threshold TTT, eventually stabilizing—indicating strong learning capability and convergence of the decision strategy. When T = 5%, the accuracy reaches as high as 97%; even with a stricter threshold of T = 2%, it consistently remains above 89%.

These results demonstrate that the DCB strategy exhibits robust model discrimination and selection ability, particularly under more lenient evaluation criteria. Regardless of the chosen gap threshold, the final accuracy consistently remains high, validating the stability and effectiveness of the DCB model in selecting optimal sub-models within contextual environments.

This experiment further confirms that the DCB model not only delivers excellent predictive performance but also demonstrates strong robustness in action selection. This makes it a reliable foundation for model integration in dynamic traffic prediction scenarios.

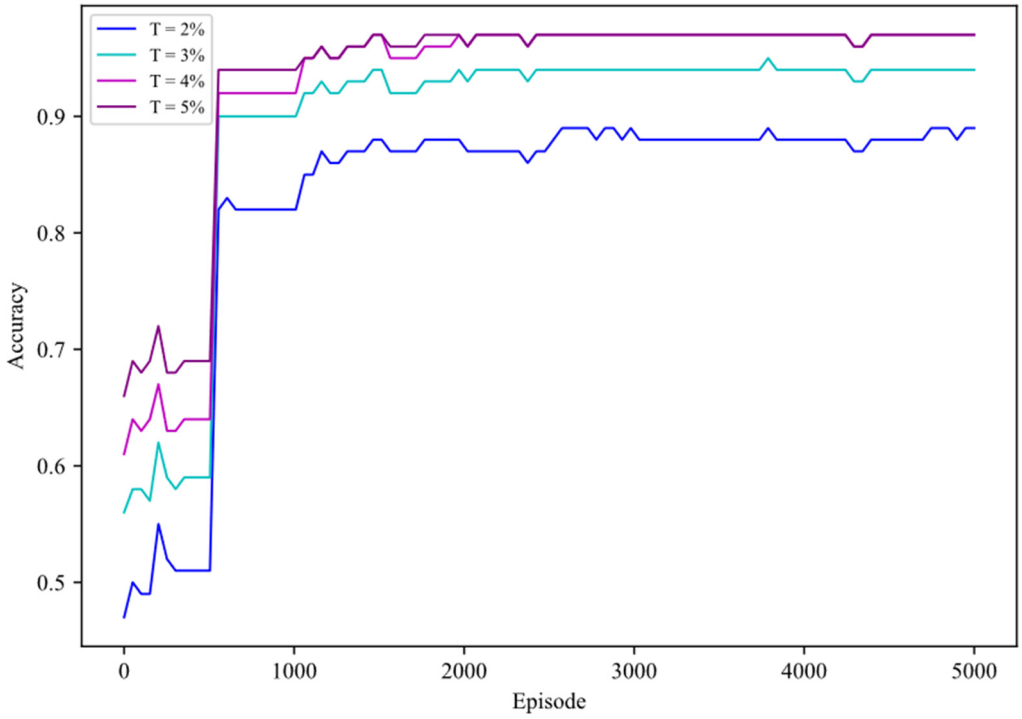


Figure 8. Accuracy of DCB.

4.3.3. Ablation Experiments

To further validate the effectiveness of each key component in the Deep Contextual Bandit (DCB) model, we conducted a series of ablation experiments by sequentially removing four core strategies from the full model and comparing their performance during training. Figure 9 illustrates the evolution of three evaluation metrics—R^2, MAE, and RMSE—for the original DCB model and its four ablated variants.

Specifically, the following model variants were evaluated:

- DCB_noBaseline: baseline model removal.
- DCB_noMemory: experience replay buffer removal.
- DCB_noDouble: removal of the double-model prediction mechanism.
- DCB_noPriority: priority sampling mechanism removal.

Each subplot in Figure 9 shows the overall test set metrics across training epochs:

1. Total R^2: The full DCB model (black curve) exhibits rapid convergence in the early training stages and maintains consistently high performance, ultimately stabilizing around 0.80. In comparison, DCB_noBaseline (cyan) and DCB_noPriority (purple) also approach the performance of the full model but converge more slowly. DCB_noMemory (green) fluctuates considerably during early training and reaches a slightly lower final R^2. DCB_noDouble (blue) performs the worst, with severe instability throughout training and poor convergence.
2. Total MAE: The trends observed in the MAE plot are consistent with the R^2 analysis. The full DCB model achieves the lowest and most stable MAE (~820), reflecting superior prediction accuracy. While DCB_noBaseline and DCB_noPriority eventually reach comparable levels, they do so slowly. DCB_noMemory exhibits unstable performance, whereas DCB_noDouble suffers from persistently high and erratic error values.

3. Total RMSE: The full DCB model demonstrates the best overall performance, maintaining the lowest and most stable RMSE throughout training. The DCB_noDouble variant stands out with a significantly higher RMSE and strong oscillations, while the remaining variants fall in between. Notably, DCB_noMemory remains consistently above the full model in terms of RMSE.

In summary, the ablation study confirms that all four core strategies in the DCB model contribute collaboratively to enhancing prediction accuracy and training stability. Among them, the baseline network and experience replay substantially impact model convergence and overall performance. Although the double-model prediction and priority sampling mechanisms have a relatively smaller effect, their removal still leads to notable performance degradation. These findings provide further empirical support for the robustness, rationality, and effectiveness of the full DCB model architecture.

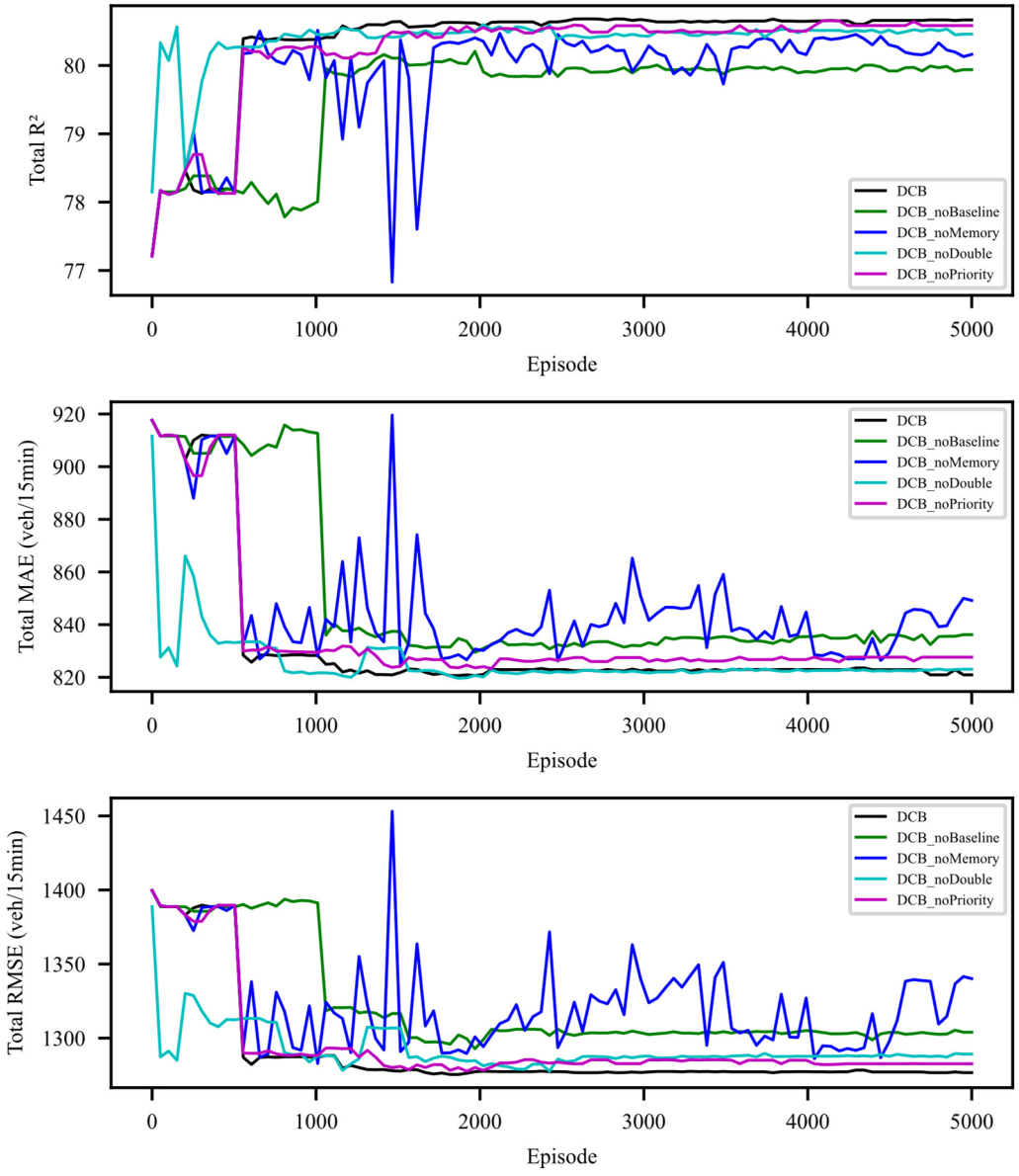


Figure 9. Ablation Experiment of DCB.

4.3.4. Analysis of Action Set and Selection Mechanism

To explicitly distinguish the performance gains derived from dynamic model selection versus those attributed solely to model diversity, this section provides a detailed analysis

of the action selection distribution. It compares the proposed framework against static ensemble strategies and model subsets.

(1) Distribution of Selected Sub-Models

Table 5 summarizes the selection frequency of each candidate sub-model by the DCB strategy across the 100 test road segments. The distribution reveals meaningful traffic-dependent specialization learned by the agent.

Table 5. Distribution of Selected Sub-Predictive Models Under DCB Strategy.

Model	BR	KNN	XGboost	SVR	RF	LSTM
Number	32	0	41	0	18	9

As observed, XGBoost and Bayesian Regression (BR) emerged as the dominant sub-models, selected for 41% and 32% of the test segments, respectively. This preference indicates their robust fitting capabilities and generalization performance across a wide range of typical traffic scenarios. Conversely, Random Forest (RF) and LSTM were selected for 18 and 9 segments, respectively. Although less frequent, their specific selection suggests that these non-linear models provide critical performance advantages under complex dynamic or structural conditions where simpler models fail.

Notably, K-Nearest Neighbors (KNN) and Support Vector Regression (SVR) were not selected for any segment. This exclusion likely reflects their limitations in scalability and robustness within high-dimensional feature spaces, prompting the DCB agent to filter them out in favor of more efficient predictors.

(2) Comparison with Static Ensembles and Model Subsets

To further validate the superiority of the dynamic selection mechanism, we compared the full DCB framework against: (i) restricted model subsets (Subset 1 & 2), (ii) static fusion methods (Voting and Bayesian Averaging [40]), and (iii) the baseline Cluster-based selection. The results are presented in Table 6.

Table 6. Performance Comparison of DCB vs. Static Ensembles and Model.

Method	R^2 (Mean ± Std)	MAE (Mean ± Std, veh/15 min)	RMSE (Mean ± Std, veh/15 min)	Average Inference Time (s)
Subset1 (“BR”, ”XGboost”, ”RF”)	80.31 ± 0.06	836.70 ± 2.04	1309.01 ± 1.81	3.05
Subset2 (“SVR”, ”RF”, ”LSTM”)	79.52 ± 0.08	871.88 ± 9.27	1370.38 ± 13.53	3.8
Voting	80.83	845.58	1313.64	12.23
Bayesian averaging	80.99	836.46	1300.99	11.94
Cluster	80.09 ± 0.03	848.07 ± 1.16	1321.90 ± 7.91	4.02
DCB	80.59 ± 0.10	823.33 ± 1.80	1281.09 ± 4.26	1.26

(3) Traffic-Dependent Specialization Analysis

To deeply investigate how the DCB framework adapts its selection strategy across time and space, we visualized the distribution of selected sub-models across three distinct dimensions: temporal dependency (quantified by autocorrelation), spatial topology (degree centrality), and traffic fluctuation (kurtosis).

As shown in Figure 10, the results reveal a distinct context-dependent selection pattern learned by the DCB agent. Temporally, LSTM is the preferred model for segments with high

autocorrelation (57.1%), effectively capturing long-term historical dependencies, whereas XGBoost becomes dominant in low-dependency scenarios (54.3%) to capture complex non-linear mappings when historical trends are weak. Spatially, the selection strategy adapts to topological complexity. While core nodes with high degree centrality require a diverse mixture of models (with XGBoost and Bayesian Regression being equally favored, ~35% each), peripheral nodes are primarily modeled by XGBoost (54.5%). Regarding traffic fluctuation, the framework exhibits a robust mechanism for stability: in high-volatility scenarios (high kurtosis), the agent overwhelmingly selects Bayesian Regression (75.0%). This suggests that the framework prioritizes the regularization and probabilistic stability of linear models to prevent overfitting to outliers. Conversely, under stable traffic conditions, XGBoost reclaims dominance (61.9%), allowing the system to exploit fine-grained non-linear patterns when the signal-to-noise ratio is high. These correlations confirm that the DCB framework has successfully learned a "principle of adaptability"—balancing model complexity and stability according to the real-time spatiotemporal state of the traffic flow.

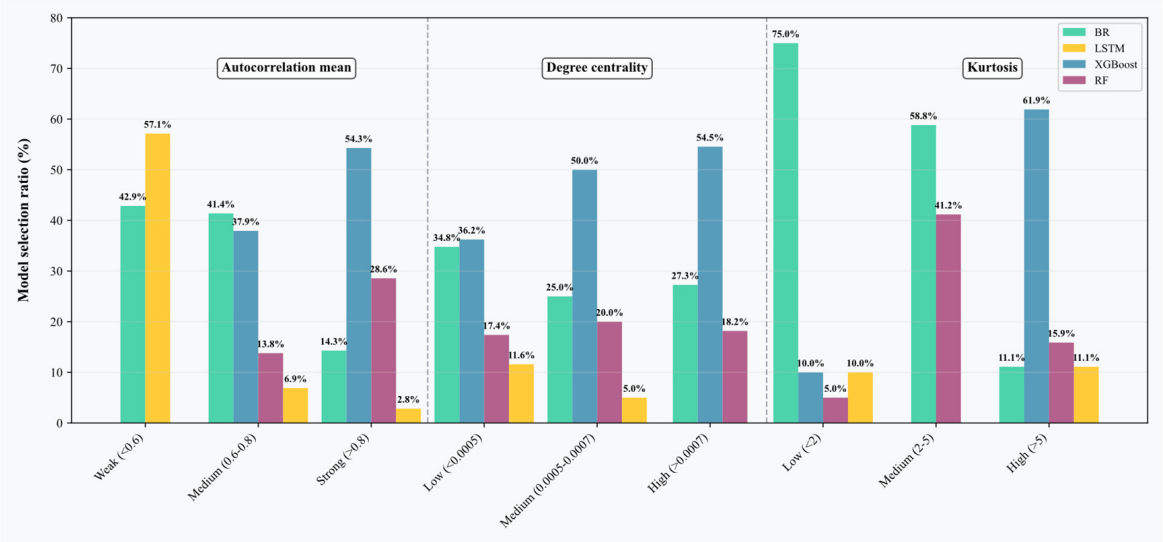


Figure 10. Distribution of selected sub-models across temporal, spatial, and fluctuation dimensions.

These results highlight two critical insights. First, while static ensembles (e.g., Voting, Bayesian Averaging) achieve comparable R^2 scores by smoothing fluctuations, DCB yields significantly lower error metrics (MAE/RMSE). This distinction indicates that dynamic selection mitigates the 'averaging bias' inherent in static fusion, ensuring superior point-wise precision closer to the ground truth. Second, DCB achieves this accuracy with reduced inference costs by executing a single optimal model rather than a computation-heavy ensemble. This offers a decisive practical advantage for real-time deployment in large-scale road networks, balancing high accuracy with operational efficiency.

In summary, these findings confirm that the performance gain of DCB stems from its context-aware dynamic selection capability rather than merely model diversity, making it a more efficient and accurate solution for large-scale deployment.

4.3.5. Comparison with Benchmark Models

To further validate the effectiveness of the proposed method, we compare the DCB model with several mainstream traffic flow prediction models, including Spatio-Temporal Graph Convolutional Networks (STGCN) [41], Attention-Based Spatial–Temporal Graph Convolutional Networks (ASTGCN) [42], Spatial–Temporal Synchronous Graph Convolutional Networks (STSGCN) [20], Multi-Scale Dynamic Regression (MSDR) [43], and

Spatial–Temporal Wavelet Attention (STWA) [44]—all of which are representative spatiotemporal modeling approaches. The comparison results are presented in Table 7.

Table 7. Performance Comparison with Mainstream Traffic Prediction Models.

Models	R^2 (Mean ± Std)	MAE (Mean ± Std, veh/15 min)	RMSE (Mean ± Std, veh/15 min)	Average Inference Time (s)
STGCN	72.76 ± 1.51	1109.66 ± 22.64	1581.12 ± 23.56	1.49
ASTGCN	80.24 ± 0.12	834.31 ± 7.95	1323.12 ± 12.09	2.46
STSGCN	78.82 ± 0.18	903.09 ± 8.01	1391.06 ± 9.25	4.61
MSDR	75.18 ± 1.00	962.27 ± 37.99	1449.34 ± 43.77	3.26
STWA	76.14 ± 0.92	1013.36 ± 32.24	1552.90 ± 44.23	2.21
DCB	80.59 ± 0.10	823.33 ± 1.80	1281.09 ± 4.26	1.26

Across the three evaluation metrics—R^2, MAE, and RMSE—as well as computational efficiency, the DCB model achieves the best overall performance. Specifically, it attains an R^2 score of 80.59 ± 0.10, which is considerably higher than all baseline methods, demonstrating its superior capability in modeling both temporal dynamics and spatial dependencies of traffic flow. Additionally, it achieves the lowest prediction errors, with an MAE of 823.33 ± 1.80 and RMSE of 1281.09 ± 4.26, where the minimal standard deviations further confirm its exceptional prediction stability compared to baselines like MSDR (±37.99 MAE) or STWA (±44.23 RMSE). Although ASTGCN performs competitively among existing methods, achieving an R^2 of 80.24 ± 0.12, it still falls short in terms of MAE and RMSE. Notably, regarding inference efficiency, DCB requires only 1.26 s on average, reducing the inference time by approximately 48.8% compared to ASTGCN (2.46 s) and 72.7% compared to STSGCN (4.61 s). This highlights the practical advantage of DCB in terms of predictive accuracy, robustness, and real-time feasibility in complex traffic environments.

In summary, the findings indicate that deterministically selecting a single optimal model per contextual scenario—instead of blending predictions through model averaging or weighted fusion—leads to improved prediction accuracy in heterogeneous road networks. This highlights a key advantage of the bandit-based dynamic selection framework over conventional soft fusion methods. Overall, the DCB model consistently exhibits superior performance in large-scale and heterogeneous traffic flow forecasting tasks, establishing it as a promising and practical solution for real-world dynamic traffic modeling.

5. Conclusions and Future Work

This paper proposes a methodological framework based on the Deep Contextual Bandit (DCB) for the model selection problem in large-scale traffic flow prediction tasks. The approach centers on selecting the optimal prediction model by leveraging contextual information from road segments to dynamically decide among multiple sub-models, thereby achieving accurate and efficient traffic flow forecasting. Compared to traditional static multi-model fusion methods, DCB introduces context awareness and policy learning mechanisms, resulting in stronger generalization capability and adaptability.

To further enhance model performance and training efficiency, several mechanism improvements are incorporated into the DCB framework: an experience pooling mechanism for efficient reuse of historical interaction data, a clustering-based optimization strategy to compress the hyperparameter search space and improve model transferability, and a relative gain mechanism to stabilize training. Experiments conducted on multiple real-world traffic datasets demonstrate that the proposed method outperforms traditional static multi-model approaches and some reinforcement learning baselines in key metrics

such as R^2, MAE, and RMSE, particularly in large-scale road network scenarios where it achieves higher accuracy and stability.

Furthermore, analysis of action selection frequency reveals that DCB adaptively selects the optimal sub-model according to the contextual characteristics of individual road segments, highlighting its strong strategy discrimination ability. By demonstrating superior performance over attention-based ensembles, mixture-of-experts, and Bayesian model averaging, the DCB framework validates that a hard-selection strategy driven by contextual bandits is more effective than soft fusion for large-scale, heterogeneous traffic prediction, offering a better balance among accuracy, interpretability, and computational efficiency. Ablation studies further confirm the critical contributions of key modules—such as baseline reward, experience pooling, dual-model estimation, and prioritized experience sampling—to overall performance gains.

Nevertheless, there remains room for further improvement. Future research directions include: (1) incorporating or replacing with other advanced prediction sub-models to push the accuracy ceiling higher; (2) integrating reinforcement learning techniques such as policy gradients or actor-critic frameworks to enhance long-term decision-making; (3) employing graph neural networks to capture spatial dependencies in traffic networks better; and (4) leveraging transfer learning to improve adaptability under dynamic environmental conditions.

The innovative multi-model dynamic selection framework based on deep contextual bandits presented here offers a novel solution to the heterogeneous traffic flow prediction challenge in large-scale road networks. It effectively enhances the generalization, accuracy, and stability of traffic forecasting models, providing a promising direction for future research and practical applications. Practically, the framework enables high-precision, segment-level model adaptation. Traffic managers can leverage this capability to implement proactive congestion mitigation and dynamic route optimization, significantly enhancing urban mobility efficiency.

Author Contributions: Conceptualization, Z.Z. and Z.Y.; methodology, S.C. and S.M.; software, Y.J. and Z.Y.; validation, S.M., Y.W. and Z.Y.; formal analysis, S.M., Z.Z. and X.Z.; investigation, X.Z.; resources, Y.J. and Z.Y.; data curation, X.Z.; writing—original draft preparation, S.C. and S.M.; writing—review and editing, S.C., S.M., Z.Z. and X.Z.; visualization, Y.W. and Z.Y.; supervision, Y.J. and Z.Y.; project administration, Y.J. and Z.Y.; funding acquisition, Y.J. and Z.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the Humanities and Social Science Foundation of the Ministry of Education in China 25YJCZH339, in part by the National Natural Science Foundation of China 72471200, in part by the Sichuan Science and Technology Program 2025NSFSC2000, and in part by the Guangxi Science and Technology Program AA21077011, and in part by the International Science and Technology Collaboration Project of Sichuan Province 24GJHZ0342.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Acknowledgments: During the preparation of this work, the authors used DeepSeek-V3.2 (an AI-assisted tool) to improve language clarity and readability. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the published article.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

Appendix A.1

Appendix A.1 provides the optimal hyperparameter configurations for all candidate predictive models across the 14 identified cluster groups. Table A1 details the tuned parameter sets for Bayesian Regression (BR), K-Nearest Neighbors (KNN), eXtreme Gradient

Boosting (XGboost), Support Vector Regression (SVR), Random Forest (RF), and Long Short-Term Memory (LSTM) networks. These configurations were obtained via grid search performed on the representative road segment of each cluster, as described in Section 4.2.4. The table systematically lists the key hyperparameters for every model–cluster combination, ensuring reproducibility and offering insight into how traffic-pattern heterogeneity is accommodated through differentiated parameter tuning. This appendix information supports the experimental transparency and allows readers to verify or extend the presented framework under comparable settings.

Table A1. Optimal hyperparameter values for all candidate predictive models.

Cluster Group	BR	KNN	XGboost
1	$\alpha1 = 1 \times 10^{-6}, \alpha2 = 0.0001,$ $l1 = 0.0001, l2 = 1 \times 10^{-6}$	$n = 9, w = 'distance',$ $algo = 'auto'$	$md = 6, lr = 0.1,$ $s = 0.8, col = 0.8$
2	$\alpha1 = 1 \times 10^{-6}, \alpha2 = 0.0001,$ $l1 = 0.0001, l2 = 0.0001$	$n = 7, w = 'distance',$ $algo = 'ball_tree'$	$md = 3, lr = 0.3,$ $s = 0.8, col = 0.6$
3	$\alpha1 = 1 \times 10^{-6}, \alpha2 = 0.0001,$ $l1 = 0.0001, l2 = 1 \times 10^{-6}$	$n = 9, w = 'uniform',$ $algo = 'auto'$	$md = 3, lr = 0.1,$ $s = 0.8, col = 0.8$
4	$\alpha1 = 0.0001, \alpha2 = 1 \times 10^{-6},$ $l1 = 1 \times 10^{-6}, l2 = 0.0001$	$n = 9, w = 'distance',$ $algo = 'auto'$	$md = 3, lr = 0.1,$ $s = 0.9, col = 0.8$
5	$\alpha1 = 1 \times 10^{-6}, \alpha2 = 1 \times 10^{-6},$ $l1 = 0.0001, l2 = 0.0001$	$n = 9, w = 'distance',$ $algo = 'kd_tree'$	$md = 3, lr = 0.1,$ $s = 0.8, col = 0.8$
6	$\alpha1 = 1 \times 10^{-6}, \alpha2 = 0.0001,$ $l1 = 0.0001, l2 = 1 \times 10^{-6}$	$n = 9, w = 'distance',$ $algo = 'kd_tree'$	$md = 6, lr = 0.1,$ $s = 0.9, col = 0.8$
7	$\alpha1 = 0.0001, \alpha2 = 1 \times 10^{-6},$ $l1 = 1 \times 10^{-6}, l2 = 0.0001$	$n = 9, w = 'uniform',$ $algo = 'ball_tree'$	$md = 3, lr = 0.3,$ $s = 0.7, col = 0.6$
8	$\alpha1 = 1 \times 10^{-6}, \alpha2 = 0.0001,$ $l1 = 0.0001, l2 = 1 \times 10^{-6}$	$n = 9, w = 'distance',$ $algo = 'auto'$	$md = 3, lr = 0.3,$ $s = 0.8, col = 0.8$
9	$\alpha1 = 0.0001, \alpha2 = 1 \times 10^{-6},$ $l1 = 1 \times 10^{-6}, l2 = 0.0001$	$n = 9, w = 'distance',$ $algo = 'auto'$	$md = 3, lr = 0.1,$ $s = 0.9, col = 0.6$
10	$\alpha1 = 0.0001, \alpha2 = 0.0001,$ $l1 = 1 \times 10^{-6}, l2 = 0.0001$	$n = 9, w = 'uniform',$ $algo = 'auto'$	$md = 3, lr = 0.1,$ $s = 0.7, col = 0.8$
11	$\alpha1 = 0.0001, \alpha2 = 1 \times 10^{-6},$ $l1 = 1 \times 10^{-6}, l2 = 0.0001$	$n = 9, w = 'distance',$ $algo = 'kd_tree'$	$md = 3, lr = 0.1,$ $s = 0.8, col = 0.8$
12	$\alpha1 = 0.0001, \alpha2 = 1 \times 10^{-6},$ $l1 = 1 \times 10^{-6}, l2 = 0.0001$	$n = 9, w = 'uniform',$ $algo = 'ball_tree'$	$md = 3, lr = 0.1,$ $s = 0.9, col = 0.8$
13	$\alpha1 = 1 \times 10^{-6}, \alpha2 = 0.0001,$ $l1 = 0.0001, l2 = 1 \times 10^{-6}$	$n = 9, w = 'uniform',$ $algo = 'kd_tree'$	$md = 3, lr = 0.1,$ $s = 0.9, col = 0.8$
14	$\alpha1 = 1 \times 10^{-6}, \alpha2 = 0.0001,$ $l1 = 0.0001, l2 = 1e - 6$	$n = 9, w = 'distance',$ $algo = 'kd_tree'$	$md = 3, lr = 0.1,$ $s = 0.9, col = 0.8$
Cluster Group	**SVR**	**RF**	**LSTM**
1	$c = 0.1, e = 0.01$	$nt = 150, md = 30,$ $ms = 2$	$ep = 100, lr = 0.01$
2	$c = 0.1, e = 2$	$nt = 100, md = 20,$ $ms = 10$	$ep = 150, lr = 0.001$
3	$c = 0.1, e = 0.01$	$nt = 150, md = 10,$ $ms = 10$	$ep = 150, lr = 0.001$
4	$c = 0.1, e = 0.01$	$nt = 50, md = 20,$ $ms = 10$	$ep = 150, lr = 0.001$
5	$c = 0.1, e = 0.01$	$nt = 150, md = 30,$ $ms = 10$	$ep = 50, lr = 0.001$
6	$c = 0.1, e = 0.01$	$nt = 150, md = none,$ $ms = 10$	$ep = 150, lr = 0.001$

Table A1. *Cont.*

7	$c = 0.1, e = 0.01$	$nt = 100, md = 20, ms = 10$	$ep = 50, lr = 0.001$
8	$c = 0.1, e = 2$	$nt = 100, md = none, ms = 10$	$ep = 100, lr = 0.001$
9	$c = 0.1, e = 0.01$	$nt = 150, md = 10, ms = 10$	$ep = 150, lr = 0.001$
10	$c = 1, e = 0.01$	$nt = 50, md = 10, ms = 10$	$ep = 100, lr = 0.001$
11	$c = 0.1, e = 0.01$	$nt = 100, md = 20, ms = 2$	$ep = 150, lr = 0.001$
12	$c = 1, e = 0.01$	$nt = 50, md = 10, ms = 2$	$ep = 100, lr = 0.01$
13	$c = 0.1, e = 0.01$	$nt = 150, md = 10, ms = 10$	$ep = 150, lr = 0.01$
14	$c = 0.1, e = 0.01$	$nt = 150, md = 20, ms = 10$	$ep = 150, lr = 0.001$

References

1. Song, Y.; Duan, H.; Cheng, Y. A Novel Fractional-Order Grey Euler Prediction Model and Its Application in Short-Term Traffic Flow. *Chaos Solitons Fractals* **2024**, *189*, 115722. [CrossRef]
2. Xu, Y.; Wei, L.; Feng, L. A Spatial-Temporal Trend-Event Decoupling Dual-Channel Framework for Traffic Flow Prediction. *Expert Syst. Appl.* **2025**, *287*, 128107. [CrossRef]
3. Zhao, W.; Yuan, G.; Zhang, Y.; Liu, X.; Liu, S.; Zhang, L. An Interpretable and Efficient Multi-Scale Spatio-Temporal Neural Network for Traffic Flow Forecasting. *Expert Syst. Appl.* **2026**, *296*, 128961. [CrossRef]
4. Ali, A.; Naeem, H.Y.; Sharafian, A.; Qiu, L.; Wu, Z.; Bai, X. Dynamic Multi-Graph Spatio-Temporal Learning for Citywide Traffic Flow Prediction in Transportation Systems. *Chaos Solitons Fractals* **2025**, *199*, 116898. [CrossRef]
5. Chen, J.; Zhang, S.; Xu, W. Scalable Prediction of Heterogeneous Traffic Flow with Enhanced Non-Periodic Feature Modeling. *Expert Syst. Appl.* **2025**, *294*, 128847. [CrossRef]
6. He, G.; Huang, W.; Zhu, Y.; Huang, M. Adaptive Spatial-Temporal Dependence Graph Convolution Neural Network for Traffic Flow Prediction. *Expert Syst. Appl.* **2025**, *281*, 127564. [CrossRef]
7. Wang, T.; Ngoduy, D.; Li, Y.; Lyu, H.; Zou, G.; Dantsuji, T. Koopman Theory Meets Graph Convolutional Network: Learning the Complex Dynamics of Non-Stationary Highway Traffic Flow for Spatiotemporal Prediction. *Chaos Solitons Fractals* **2024**, *187*, 115437. [CrossRef]
8. Yang, Y.; Huang, H.; Li, G.; Han, B.; Yuan, Z.; Ma, H. A systematic review of resilience assessment and enhancement of urban integrated transportation networks. *J. Transp. Geogr.* **2025**, *129*, 104420. [CrossRef]
9. Anna, V.A.B.K.; Chand, S.; Alsultan, A.; Dixit, V. Investigating the spatial effects of zonal factors on road traffic speed variability during peak hour. *PLoS ONE* **2026**, *21*, e0340583. [CrossRef]
10. Chen, Z.; Liu, K.; Wang, J.; Yamamoto, T. H-ConvLSTM-Based Bagging Learning Approach for Ride-Hailing Demand Prediction Considering Imbalance Problems and Sparse Uncertainty. *Transp. Res. Part C Emerg. Technol.* **2022**, *140*, 103709. [CrossRef]
11. Chen, J.; Liu, K.; Li, R.; Li, W.; Chen, Q. Optimising Built Environment to Reduce Car Use: Spatial and Attribute Heterogeneity Perspectives. *Transp. Res. Part D Transp. Environ.* **2025**, *143*, 104767. [CrossRef]
12. He, Y.; Huang, P.; Hong, W.; Luo, Q.; Li, L.; Tsui, K.-L. In-Depth Insights into the Application of Recurrent Neural Networks (RNNs) in Traffic Prediction: A Comprehensive Review. *Algorithms* **2024**, *17*, 398. [CrossRef]
13. Shaygan, M.; Meese, C.; Li, W.; Zhao, X.; Nejad, M. Traffic Prediction Using Artificial Intelligence: Review of Recent Advances and Emerging Opportunities. *Transp. Res. Part C Emerg. Technol.* **2022**, *145*, 103921. [CrossRef]
14. Bharti; Redhu, P.; Kumar, K. Short-Term Traffic Flow Prediction Based on Optimized Deep Learning Neural Network: PSO-Bi-LSTM. *Phys. A Stat. Mech. Its Appl.* **2023**, *625*, 129001. [CrossRef]
15. Zheng, Y.; Wang, S.; Dong, C.; Li, W.; Zheng, W.; Yu, J. Urban Road Traffic Flow Prediction: A Graph Convolutional Network Embedded with Wavelet Decomposition and Attention Mechanism. *Phys. A Stat. Mech. Its Appl.* **2022**, *608*, 128274. [CrossRef]
16. Ryu, U.; Wang, J.; Pak, U.; Kwak, S.; Ri, K.; Jang, J.; Sok, K. A Clustering Based Traffic Flow Prediction Method with Dynamic Spatiotemporal Correlation Analysis. *Transportation* **2022**, *49*, 951–988. [CrossRef]
17. Huang, X.; Ye, Y.; Wang, C.; Yang, X.; Xiong, L. A Multi-Mode Traffic Flow Prediction Method with Clustering Based Attention Convolution LSTM. *Appl. Intell.* **2022**, *52*, 14773–14786. [CrossRef]

18. Deng, W.; Li, K.; Zhao, H. A Flight Arrival Time Prediction Method Based on Cluster Clustering-Based Modular with Deep Neural Network. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 6238–6247. [CrossRef]
19. Mystakidis, A.; Koukaras, P.; Tjortjis, C. Advances in Traffic Congestion Prediction: An Overview of Emerging Techniques and Methods. *Smart Cities* **2025**, *8*, 25. [CrossRef]
20. Song, C.; Lin, Y.; Guo, S.; Wan, H. Spatial-Temporal Synchronous Graph Convolutional Networks: A New Framework for Spatial-Temporal Network Data Forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*; AAAI Press: Washington, DC, USA, 2020; Volume 34, pp. 914–921. [CrossRef]
21. Liu, Y.; Zheng, H.; Feng, X.; Chen, Z. Short-Term Traffic Flow Prediction with Conv-LSTM. In Proceedings of the 2017 9th International Conference on Wireless Communications and Signal Processing (WCSP), Nanjing, China, 11–13 October 2017.
22. Chen, Y.; Chen, X. A Novel Reinforced Dynamic Graph Convolutional Network Model with Data Imputation for Network-Wide Traffic Flow Prediction. *Transp. Res. Part C Emerg. Technol.* **2022**, *143*, 103820. [CrossRef]
23. Zheng, G.; Chai, W.K.; Zhang, J.; Katos, V. VDGCNeT: A Novel Network-Wide Virtual Dynamic Graph Convolution Neural Network and Transformer-Based Traffic Prediction Model. *Knowl.-Based Syst.* **2023**, *275*, 110676. [CrossRef]
24. Zheng, G.; Chai, W.K.; Zhang, J.; Katos, V. Structured Prediction of Sparse Dependent Variables for Traffic State Estimation in Large-Scale Networks. *Appl. Soft Comput.* **2023**, *133*, 109893. [CrossRef]
25. Wang, C.; Zuo, K.; Zhang, S.; Lei, H.; Hu, P.; Shen, Z.; Wang, R.; Zhao, P. PFNet: Large-Scale Traffic Forecasting with Progressive Spatio-Temporal Fusion. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 14580–14597. [CrossRef]
26. Xing, Z.; Huang, M.; Li, W.; Peng, D. Spatial Linear Transformer and Temporal Convolution Network for Traffic Flow Prediction. *Sci. Rep.* **2024**, *14*, 4040. [CrossRef] [PubMed]
27. Schaul, T.; Quan, J.; Antonoglou, I.; Silver, D. Prioritized Experience Replay. Available online: http://arxiv.org/abs/1511.05952 (accessed on 31 July 2025).
28. Van Hasselt, H.; Guez, A.; Silver, D. Deep Reinforcement Learning with Double Q-Learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*; AAAI Press: Washington, DC, USA, 2016; Volume 30. [CrossRef]
29. Chen, Y.; Hu, M.; Yang, L.; Xu, Y.; Xie, H. General Multi-Agent Reinforcement Learning Integrating Adaptive Manoeuvre Strategy for Real-Time Multi-Aircraft Conflict Resolution. *Transp. Res. Part C Emerg. Technol.* **2023**, *151*, 104125. [CrossRef]
30. Li, C.; Yue, X.; Liu, Z.; Ma, G.; Zhang, H.; Zhou, Y.; Zhu, J. A Modified Dueling DQN Algorithm for Robot Path Planning Incorporating Priority Experience Replay and Artificial Potential Fields. *Appl. Intell.* **2025**, *55*, 366. [CrossRef]
31. Nie, L.; Wang, X.; Zhao, Q.; Shang, Z.; Feng, L.; Li, G. Digital Twin for Transportation Big Data: A Reinforcement Learning-Based Network Traffic Prediction Approach. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 896–906. [CrossRef]
32. Ullah, I.; Han, Y.-H. Optimizing Vehicular Edge Computing: Graph-Based Double-DQN Approaches for Intelligent Task Offloading. *J. Supercomput.* **2024**, *81*, 76. [CrossRef]
33. Auer, P.; Cesa-Bianchi, N.; Fischer, P. Finite-Time Analysis of the Multiarmed Bandit Problem. *Mach. Learn.* **2002**, *47*, 235–256. [CrossRef]
34. Riquelme, C.; Tucker, G.; Snoek, J. Deep Bayesian Bandits Showdown: An Empirical Comparison of Bayesian Deep Networks for Thompson Sampling. *arXiv* **2018**, arXiv:1802.09127. [CrossRef]
35. Sutton, R.S.; Barto, A.G. *Reinforcement Learning: An Introduction*; MIT Press: Cambridge, MA, USA, 1998. Available online: https://ieeexplore.ieee.org/book/6267343 (accessed on 18 January 2026).
36. Chapelle, O.; Li, L. An Empirical Evaluation of Thompson Sampling. Advances in Neural Information Processing Systems, 24. 2011. Available online: https://papers.nips.cc/paper/2011/hash/e53a0a2978c28872a4505bdb51db06dc-Abstract.html (accessed on 23 May 2025).
37. Yang, Y.; Chen, X.; Wang, J.; Dong, Y.; Qie, K.; Yuan, Z. Enhancing vision-based traffic crash detection performance consistency across day-night scenes: A depth-aware and domain-adaptive network. *Accid. Anal. Prev.* **2026**, *228*, 108405. [CrossRef]
38. Fedus, W.; Ramachandran, P.; Agarwal, R.; Bengio, Y.; Larochelle, H.; Rowland, M.; Dabney, W. Revisiting Fundamentals of Experience Replay. In Proceedings of the International Conference on Machine Learning, Vienna, Austria, 12–18 July 2020.
39. Salamanis, A.I.; Lipitakis, A.-D.; Gravvanis, G.A.; Kotsiantis, S.; Anagnostopoulos, D. An Adaptive Cluster-Based Sparse Autoregressive Model for Large-Scale Multi-Step Traffic Forecasting. *Expert Syst. Appl.* **2021**, *180*, 115093. [CrossRef]
40. Raftery, A.E.; Gneiting, T.; Balabdaoui, F.; Polakowski, M. Using Bayesian Model Averaging to Calibrate Forecast Ensembles. *Mon. Weather. Rev.* **2005**, *133*, 1155–1174. [CrossRef]
41. Yu, B.; Yin, H.; Zhu, Z. Spatio-Temporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*; AAAI Press: Washington, DC, USA, 2018; pp. 3634–3640. [CrossRef]
42. Guo, S.; Lin, Y.; Feng, N.; Song, C.; Wan, H. Attention Based Spatial-Temporal Graph Convolutional Networks for Traffic Flow Forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*; AAAI Press: Washington, DC, USA, 2019; Volume 33, pp. 922–929. [CrossRef]

43. Liu, D.; Wang, J.; Shang, S.; Han, P. MSDR: Multi-Step Dependency Relation Networks for Spatial Temporal Forecasting. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*; Association for Computing Machinery: New York, NY, USA, 2022; pp. 1042–1050. [CrossRef]
44. Cirstea, R.-G.; Yang, B.; Guo, C.; Kieu, T.; Pan, S. Towards Spatio-Temporal Aware Traffic Time Series Forecasting—Full Version. *arXiv* **2022**, arXiv:2203.15737. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.