

**OPEN**  
**ANALYSIS**

# FollowNet: A Comprehensive Benchmark for Car-Following Behavior Modeling

Xianda Chen<sup>1</sup>, Meixin Zhu<sup>1,2,3</sup>✉, Kehua Chen<sup>3</sup>, Pengqin Wang<sup>1</sup>, Hongliang Lu<sup>1</sup>, Hui Zhong<sup>1</sup>, Xu Han<sup>4</sup>, Xuesong Wang<sup>5</sup> & Yin Hai Wang<sup>6</sup>

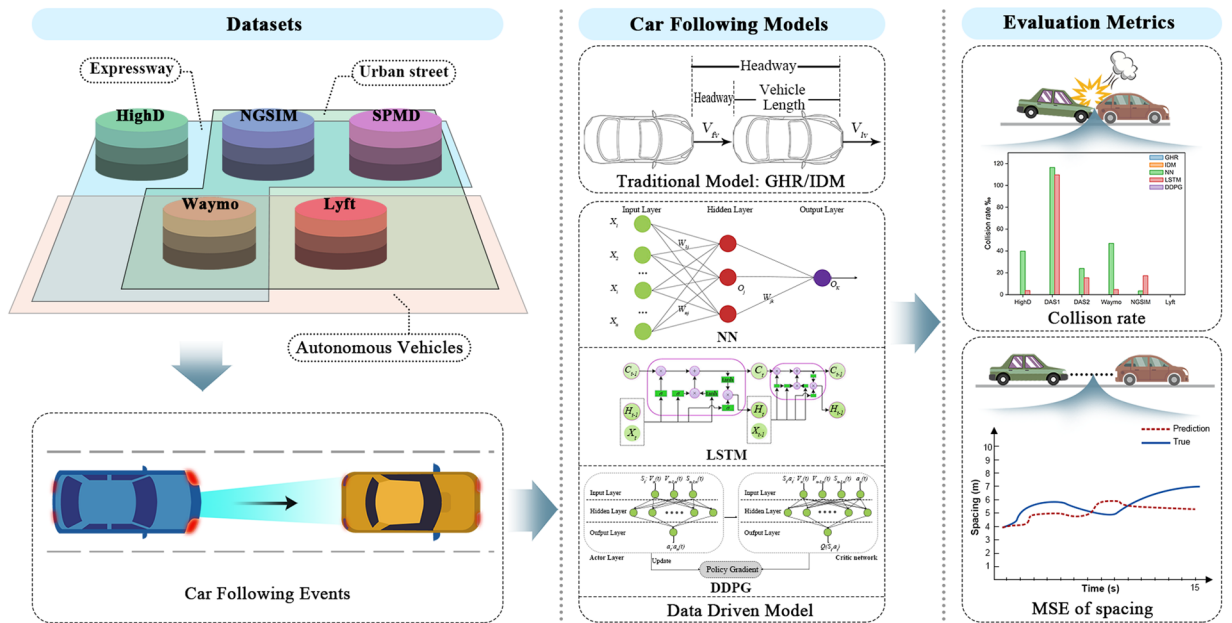
Car-following is a control process in which a following vehicle adjusts its acceleration to keep a safe distance from the lead vehicle. Recently, there has been a booming of data-driven models that enable more accurate modeling of car-following through real-world driving datasets. Although there are several public datasets available, their formats are not always consistent, making it challenging to determine the state-of-the-art models and how well a new model performs compared to existing ones. To address this gap and promote the development of microscopic traffic flow modeling, we establish the first public benchmark dataset for car-following behavior modeling. This benchmark consists of more than 80 K car-following events extracted from five public driving datasets under the same criteria. To give an overview of current progress in car-following modeling, we implemented and tested representative baseline models within the benchmark. The established benchmark provides researchers with consistent data formats and metrics for cross-comparing different car-following models, coming with open datasets and codes.

## Introduction

Car-following is the most fundamental and frequent driving behavior. It involves actions taken by a driver when following another vehicle ahead. Proper car-following behavior reduces crashes and improves traffic flow stability<sup>1-4</sup>. The corresponding car-following model is a mathematical or computational representation of the behavior exhibited by drivers when following other vehicles on the road. It describes and predicts the dynamics of following vehicles (FV) and lead vehicles (LV) movements in traffic flow and serves as a cornerstone for microscopic traffic simulation<sup>5-7</sup>.

Over the past decade, there has been a boom in data-driven car-following models, primarily due to the availability of real-world driving data and advancements in machine learning. Representative data-driven car-following models include neural network based<sup>8,9</sup>, recurrent neural network based<sup>10,11</sup>, and reinforcement learning based<sup>12-15</sup>. However, existing research has several limitations. Firstly, there is a lack of standardized data formats and evaluation criteria for car-following models. Although several public driving datasets are available, such as NGSIM and HighD, there are no uniform data formats and evaluation criteria for car-following models, making it challenging to compare newly proposed models with existing ones in terms of performance. Secondly, current research inadequately represents car-following behavior in mixed traffic flows due to limited datasets. As we enter a transitional phase where self-driving and human-driven vehicles share the road, previous studies have mainly focused on modeling car-following behavior using limited datasets that do not account for autonomous vehicles.

<sup>1</sup>Systems Hub, Intelligent Transportation Thrust, The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, 511400, China. <sup>2</sup>Guangdong Provincial Key Lab of Integrated Communication, Sensing and Computation for Ubiquitous Internet of Things, Guangzhou, 511400, China. <sup>3</sup>Department of Civil and Environmental Engineering, The Hong Kong University of Science and Technology, Hong Kong, Hong Kong. <sup>4</sup>Information Hub, Data Science and Analytics Thrust, The Hong Kong University of Science and Technology (Guangzhou), Guangzhou, 511400, China. <sup>5</sup>School of Transportation Engineering, Tongji University, Shanghai, 201804, China. <sup>6</sup>Department of Civil and Environmental Engineering, University of Washington, Seattle, 98195, Washington, USA. ✉e-mail: [meixin@ust.hk](mailto:meixin@ust.hk)



**Fig. 1** A road map of car-following benchmark: FollowNet.

To further advance the field of microscopic traffic simulation modeling, it is imperative to establish a public car-following benchmark that can address the aforementioned issues and serve as a standard dataset. Similar to how standard datasets such as ImageNet<sup>16</sup>, Microsoft COCO<sup>17</sup>, and KITTI<sup>18</sup> have contributed to their respective fields, a car-following benchmark would be beneficial for advancing the field of microscopic traffic flow modeling. To achieve this, we developed a benchmark called FollowNet. The benchmark was created by extracting car-following events from five publicly available datasets using consistent criteria. Within the benchmark, we implemented and tested five baseline car-following models, including both traditional and data-driven approaches (Fig. 1). This paper summarizes recent developments in this field, as well as evaluates the performance of mainstream models using our benchmark.

Our study has the following contributions:

- We systematically summarized existing car-following models.
- We established the first benchmark for car-following behavior with consistent data formats to streamline the development of car-following models. Our standardized car-following benchmark addresses the challenges of handling various data structures and frameworks from different datasets.
- We provided various scenarios, including mixed traffic and different road types, to test representative car-following models under standardized metrics.
- Our open-sourced data and codebase provided valuable insights for car-following data extraction, model calibration, and baseline implementation.

### Related work

**Traditional car-following models.** Since the Pipes model<sup>19</sup> was proposed in 1953, researchers have studied car-following models for 70 years. In general, existing traditional car-following models include five main categories: stimulus-response model, safe distance model, psycho-physiological model, optimal velocity model, and desired goal model<sup>20–22</sup>.

The stimulus-response model uses the relative distance or relative speed to determine the acceleration of the FV. The Gazis-Herman-Rothery (GHR) model<sup>23</sup>, the first stimulus-response model, assumes that a vehicle’s acceleration positively relates to the speed difference and that the FV’s response is expressed regarding its acceleration or deceleration behavior. The basic equation of this model can be expressed as:

$$a_n(t) = cv_n^m(t) \frac{\Delta v(t - T)}{\Delta x^l(t - T)} \tag{1}$$

where the  $n$ -th vehicle’s acceleration at time  $t$  is represented by  $a_n(t)$ ,  $\Delta v$ , and  $\Delta x$  are the speed difference and distance between the  $(n - 1)$ -th vehicle and the  $n$ -th vehicle, respectively.  $m$ ,  $c$ , and  $l$  are constants that need to be calculated, and  $T$  is the driver’s reaction time. The General Motors (GM) model<sup>24</sup> is another well-known stimulus-response model that uses the difference of the speed and space headway to determine the acceleration.

Kometani *et al.*<sup>25</sup> propose the first safety distance model. The underlying principle of the model differs from the stimulus-response model due to the LV’s unpredictable motion. This model is also known as the conflict avoidance model since the FV always keeps a minimum safe distance. The basic formula is:

$$\Delta x(t - T) = \alpha v_{n-1}^2(t - T) + \beta_1 v_n^2(t) + \beta v_n(t) + b_0 \quad (2)$$

where  $\Delta x$  denotes the distance between the  $(n-1)$ -th vehicle and the  $n$ -th vehicle,  $v_n(t)$  is the speed of the  $n$ -th vehicle at time  $t$ ,  $T$  is the driver's reaction time,  $\alpha$ ,  $\beta_1$ ,  $\beta$ ,  $b_0$  are parameters to be calibrated. Similarly, Gipps model<sup>5</sup> utilized the concept of the safe distance to facilitate car-following behavior, the driver of the FV selects a certain speed and maintains a corresponding distance from the LV to prevent a collision.

The psycho-physiological model suggests that drivers adopt strategies based on the relative motion between the LV and FV, including changes in speed and distance differences, and only react when the threshold value is exceeded. This type of model was first proposed by Michaels<sup>26</sup>, and is used in VISSIM<sup>®</sup>, a microscopic traffic simulation tool, through the Wiedemann model<sup>27</sup>.

The optimal velocity model (OVM) is introduced by Bando *et al.*<sup>28</sup>, which can account for various traffic flow phenomena, such as free flow, congested traffic, the relationship between density and traffic flow, and stop-and-go traffic waves. However, the OVM model may produce unrealistic acceleration and deceleration processes. To address this limitation, the generalized force (GF) model<sup>29</sup> is proposed, which adds the effect of negative speed difference to the optimal velocity model. However, both models neglect situations where the speed of the LV is much slower than that of the FV. The full velocity difference (FVD) model<sup>30</sup> considers the safety distance and offers a more precise acceleration function, which is given by:

$$a_n(t) = \alpha [V_n^*(\Delta X_n(t)) - V_n(t)] + \lambda (\Delta V_n(t)) \quad (3)$$

$$\lambda = \begin{cases} \lambda_0: & \Delta X_n(t) \leq s_c \\ 0: & \Delta X_n(t) > s_c \end{cases} \quad (4)$$

The first term in the acceleration function is proportional to the difference between the optimal velocity  $V_n^*(\Delta X_n(t))$  and the actual velocity  $V_n(t)$ , and the second term considers the velocity difference  $\Delta V_n(t)$  as a linear stimulus. The sensitivity coefficients are denoted by  $\alpha$  and  $\lambda$ , and  $s_c$  is a threshold value that distinguishes between following and free driving.

The desired goal model assumes that each driver has certain desired goals to achieve, such as desired following speed, desired headway, etc. The Intelligent Driver Model (IDM)<sup>31</sup> is recognized as the most commonly employed driver-based desired goal model. It stands out as a comprehensive and concise theoretical model that prioritizes safety and accident prevention. According to IDM, each driver aims to maintain a unique set of desired parameter values during car-following behavior. The model expressions are:

$$a_n(t) = a_0 \left[ 1 - \left( \frac{v_n(t)}{\tilde{v}_n} \right)^\lambda - \left( \frac{\tilde{S}_n(t)}{s_n} \right)^2 \right] \quad (5)$$

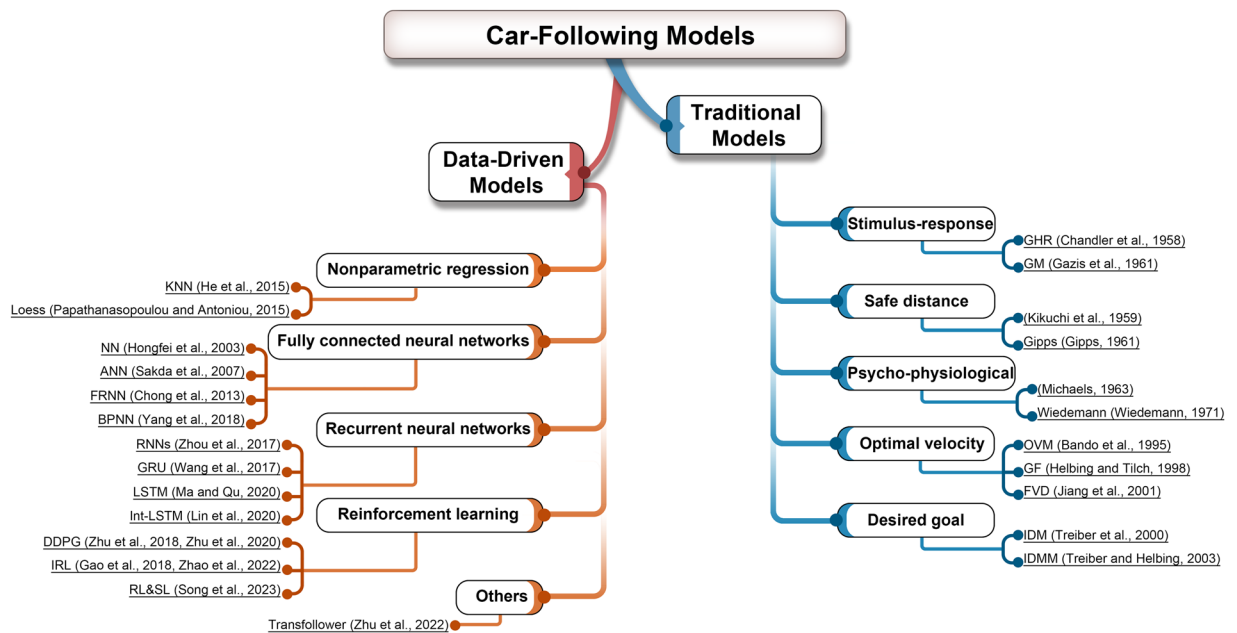
$$\tilde{S}_n(t) = S_0 + v_n(t)\tilde{T} + \frac{v_n(t)\Delta v(t)}{2\sqrt{a_0 b}} \quad (6)$$

$$\Delta v(t) = v_n(t) - v_{n-1}(t) \quad (7)$$

where  $a_n(t)$  and  $v_n(t)$  represent the acceleration and the velocity of the FV at time  $t$ , respectively. And  $S_n(t)$ ,  $\Delta v(t)$  are the spacing and relative speed between the FV and the LV. The desired maximum acceleration, comfortable deceleration, desired velocity, and desired time headway are represented by  $a_0$ ,  $b$ ,  $\tilde{v}$ , and  $\tilde{T}$ , respectively.  $S_0$  is the minimum safe headway and  $\lambda$  is a constant to be calibrated. Subsequently, the Intelligent Driver Model with Memory (IDMM)<sup>32</sup> is introduced which redesigns the IDM to incorporate memory effects and adapt driving behavior to the surrounding traffic. Furthermore, the human driver model<sup>33</sup> is applied to the IDM, which incorporates an advanced anticipative and smooth braking strategy, achieving impressive performance on real traffic data. Also, the IDM is well-designed with stochasticity to study various traffic flow oscillations, including string instability, external white acceleration noise, and indifference regions<sup>34</sup>.

Traditional car-following models may have several potential drawbacks, such as making oversimplified assumptions about driver behavior and neglecting individual driving differences or diverse driving styles. Despite these limitations, traditional ones still hold value in the study of traffic engineering and flow analysis<sup>35</sup>. For example, the IDM can cope with any traffic situation on cities, rural roads, or freeways with human-driven and autonomous leaders, and it has even been successfully applied to bicycle traffic flow and served as the basis for lane-free mixed traffic flow in developing countries such as India<sup>36</sup>. More advanced car-following models have been developed that are able to get around some of these limitations due to recent advancements in computing as well as data collection techniques.

**Data-driven car-following models.** Traditional models and data-driven models are the two main categories of car-following models, as shown in Fig. 2. Data-driven models make use of artificial intelligence techniques such as nonparametric regression, fully connected neural networks, recurrent neural networks, reinforcement learning, and other methods to predict drivers' behavior. These models learn relationships between different factors and the driver's behavior from the collected data.



**Fig. 2** The development of car-following models.

A straightforward k-nearest neighbor (KNN) based nonparametric car-following model<sup>37</sup> is presented in 2015 that forecasts the most likely driving behavior under the given circumstances. Similarly, Loess model<sup>38</sup> is also a nonparametric data-driven car-following model based on locally weighted regression.

Another common approach to predict the acceleration of the FV is to use a fully connected neural network (NN). A 4-layer neural network with two hidden layers<sup>39</sup> takes the gap distance, the relative velocity, desired velocity, and FV's speed as inputs to predict the FV's acceleration directly. The reactive agent-based car-following models are proposed by Panwai *et al.*<sup>40</sup> using artificial neural network (ANN) techniques. The results show that the models' degrees of accuracy are best for the backpropagation architecture, which is a widely used neural network algorithm for supervised learning, and the fuzzy ARTMAP architecture, which is a type of neural network that combines fuzzy logic and adaptive resonance theory. Also, one hidden layer neural network can accurately predict acceleration and Chong *et al.*<sup>9</sup> comes up with the fuzzy rule-based neural network (FRNN), which is further improved by giving the neural network an instantaneous reaction time (RT) delay<sup>41</sup>. In addition, a Back-Propagation Neural Networks (BPNN) model<sup>42</sup> combines with the Gipps model to avoid crashes.

Recurrent neural networks are another popular network structure that can model temporal problems such as vehicle following using historical information. Vanilla RNN<sup>10</sup> is utilized to simulate car-following behavior and demonstrates effective traffic oscillation prediction. Two other popular RNN variants, Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) have also been used to model car-following behavior. Ma *et al.*<sup>43</sup> utilize LSTM to address issues of gradient vanishing and exploding during long sequence training. Wang *et al.*<sup>44</sup> use GRU which can improve training efficiency to simulate car-following behavior. To address the issue of error propagation, Lin *et al.*<sup>45</sup> propose the LSTM-based interconnected car-following model (Int-LSTM).

In a reinforcement learning approach, the agent learns an optimal control policy through trial and error in an unknown environment based on a reward function. Zhu *et al.*<sup>12</sup> present a framework for modeling car-following based on deep deterministic policy gradient (DDPG), which aims to accurately reproduce human-like car-following behavior. However, human driving may not be the optimal driving operation<sup>46</sup>, and it can be optimized in terms of safety, efficiency, and comfort<sup>47</sup> rather than simply imitating human drivers. Therefore, Zhu *et al.*<sup>13</sup> propose a reward function that aims to achieve two objectives: mimicking human drivers and optimizing driving performance. Driving characteristics and human driving data are used to design the reward function and the agents are trained to learn the decision mechanism by continuously utilizing history information. Also, Hart *et al.*<sup>47</sup> propose and validate a novel car-following model based on modularized reinforcement learning. The model is trained using 100% artificial data and incorporates a multi-component reward function with weighted parameters that reflect driving behavior, which can partially address the black-box nature of conventional neural network models. It effectively updates speed and position using the Euler and ballistic methods<sup>48</sup> and outperforms other machine learning or traditional models. Other studies<sup>49–51</sup> assess individual driving characteristics and car-following behaviors using inverse reinforcement learning (IRL) or the combination of RL and Supervised Learning (SL).

In addition, a long-sequence car-following trajectory prediction model based on the Transformer attention-based model is proposed by Zhu *et al.*<sup>52</sup>, which follows a typical encoder-decoder architecture. The encoder uses multi-head self-attention to create a mixed representation of the past driving environment utilizing historical spacing and speed data as inputs. This model can effectively capture complex temporal relationships between the data, allowing it to produce a more accurate representation of the driving behavior.

Dataset	Viewpoint	Scene	Sensors	AV involved	Data Hz
HighD	External	Expressway	Camera	No	25
NGSIM	External	Expressway, Urban street	Camera	No	10
SPMD	External	Expressway, Urban street	Camera, GPS	No	10
Waymo	Driver	Expressway, Urban street	Camera, Lidar	Yes	10
Lyft	Driver	Urban street	Camera, Lidar	Yes	10

**Table 1.** Five driving datasets covered by FollowNet.

Data-driven approaches can capture the intricate correlations among variables leading to a better representation of the car-following behavior. However, their interpretability and generalizability remain a challenge. Also, standardized testing datasets and evaluation criteria are required to compare performance and determine the best model. While many public datasets are available for evaluation purposes, their data formats and standards may differ, and significant effort is required to familiarize the data structure of each dataset and extract car-following events. Therefore, creating a car-following benchmark among public datasets can simplify the model testing process and promote the development of microscopic traffic flow modeling.

## Results

We train and test two traditional and three data-driven car-following models, namely GHR, IDM, NN, LSTM, and DDPG, within the benchmark. We first present the datasets and provide a brief overview of essential concepts for subsequent analyses. We then analyze basic statistics and patterns of car-following behavior in different datasets. Finally, we evaluate the performances of baseline models using consistent metrics.

**Dataset descriptions.** The proposed benchmark consists of car-following events extracted from five mainstream public driving datasets: HighD<sup>53</sup>, Next Generation Simulation (NGSIM)<sup>54</sup>, Safety Pilot Model Deployment (SPMD)<sup>55</sup>, Waymo<sup>56</sup>, and Lyft<sup>57</sup>. Specifically, car-following events in Waymo and Lyft datasets involve mixed traffic conditions. We excluded events with more than 90% of time being still. Depending on the characteristics of each dataset, additional filtering rules may also be applied to smooth the data, as presented later in this paper. Table 1 presents basic meta information for these datasets.

**HighD** dataset<sup>53,58</sup> is released by the Institute of Automotive Engineering at RWTH Aachen University in Germany. It is a comprehensive driving dataset that provides high-precision information about vehicle positions and speeds. The dataset includes bird's-eye view videos of six different roads around Cologne, Germany, captured by a high-resolution 4K camera mounted on the drone. The vehicle position and speed information are extracted using advanced computer vision techniques, which ensures that the positioning error is under 10 cm.

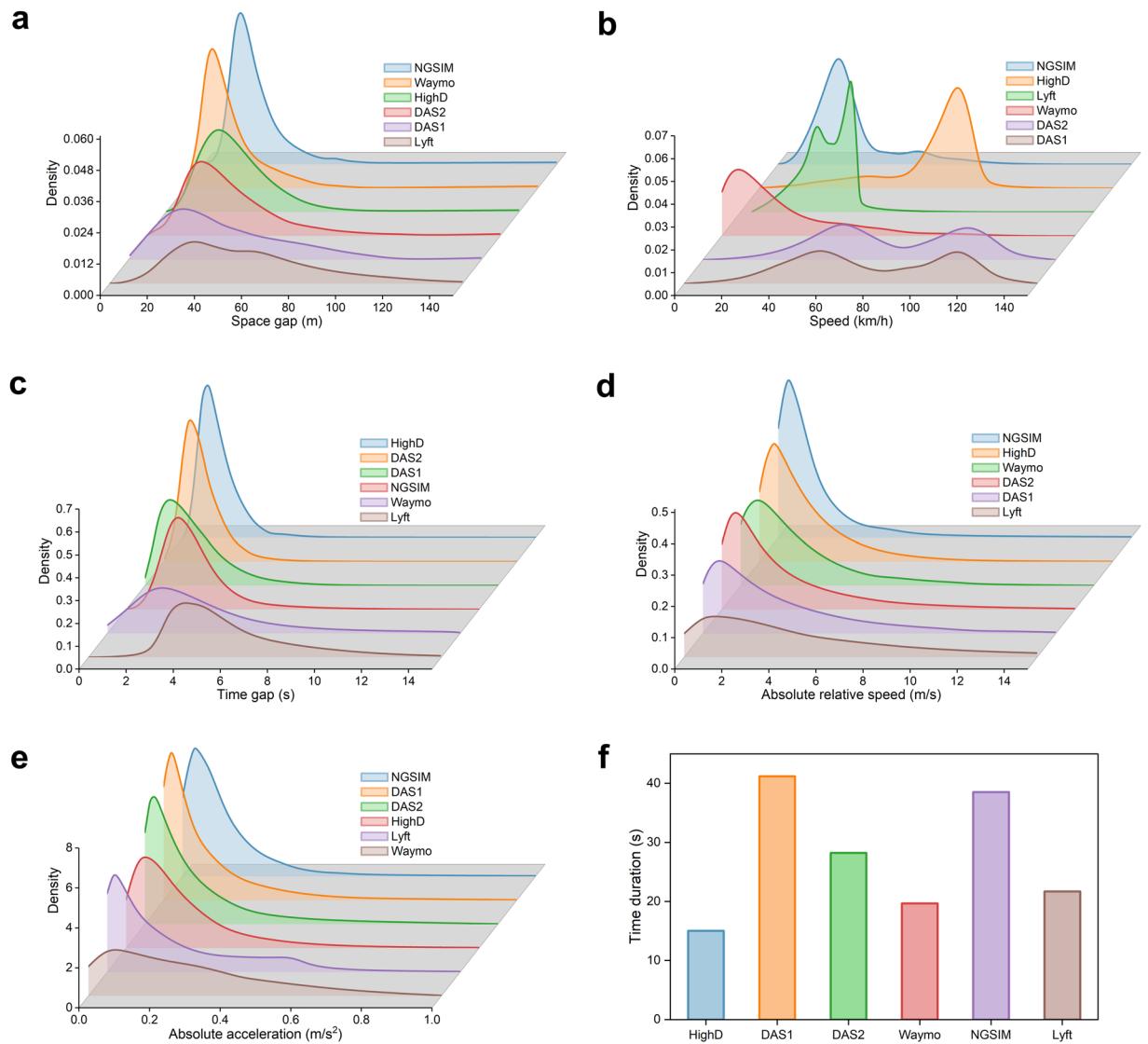
**NGSIM** (Next Generation Simulation)<sup>54</sup> is a traffic dataset created by the Federal Highway Administration (FHWA) to study the dynamics of traffic flow on expressways. The trajectory data is collected by simultaneous photography from cameras set up on the high ground. Among them, the I-80 section is located before the evening peak between 4:00 p.m. and 4:15 p.m., and the traffic condition is relatively smooth. Car-following events of I-80 are collected based on the rebuilt data<sup>59</sup> because there are measurement errors in the original data.

The **SPMD** (Safety Pilot Model Deployment) was conducted in 2014 by the U.S. Department of Transportation (USDOT) to evaluate a dedicated short-range communication technology for vehicle-to-vehicle safety applications<sup>55</sup>. The dataset includes basic safety messages, vehicle trajectory, driver-vehicle interaction data, and weather data. Most of the data presented in this dataset was gathered from vehicles that have both vehicle-to-infrastructure and vehicle-to-vehicle communication devices installed, in addition to several roadside sensors. The data acquisition system (DAS) is used to extract position and speed information from two driving datasets, namely DAS1 and DAS2. Since the spacing data can be obtained not only from sensors directly, but also theoretically inferred from the relative speed simulation of two consecutive frames, some noisy data can be filtered.

**Waymo**, a self-driving car company operated by Alphabet, the parent company of Google, released the Waymo Open Dataset<sup>56</sup> in August 2019. It includes high-resolution sensor data from lidar and cameras, along with precise 3D vehicle poses and annotations of object information. The dataset contains various driving scenarios, such as expressways and urban streets, and has a total of 1950 scenes, each lasting 20 seconds. Hu *et al.*<sup>60</sup> manually extracted car-following events based on video data of this dataset. On their basis, a total of 1440 following events were extracted.

**Lyft**<sup>57</sup> is a Level 5 autonomous driving dataset, which includes a high-definition spatial semantic map, over 55,000 3D artificial annotation frames, and data collected from 7 cameras and 3 LiDARs. This dataset is particularly useful for understanding car-following behavior in mixed traffic flows as it includes more than 170,000 scenarios of human-driven vehicles following autonomous vehicles and autonomous vehicles following human-driven vehicles. Each scene is recorded for approximately 25 seconds.

**Descriptive statistics and distributions of behavioral measures.** This section presents descriptive statistics and distributions of common car-following behavioral measurements for aforementioned datasets, as shown in Fig. 3. By normalizing the curve, the integral over the entire range of possible values is set to 1. We investigated car-following speed, spacing gap, time gap, relative speed, acceleration, and time duration across these datasets. Our analysis reveals that each dataset has unique characteristics related to its data collection location and driving scenarios. The following are key findings from the comparison of car-following behavior across different datasets.



**Fig. 3** Distributions of car-following behavioral measurements: **(a)** Average space gap during car following ( $m$ ), **(b)** Average speed during car following ( $km/h$ ), **(c)** Average time gap during car following ( $s$ ), **(d)** Average absolute relative speed during car following ( $m/s$ ), **(e)** Average absolute acceleration during car following ( $m/s^2$ ), **(f)** Average event duration.

**Space gap.** Lyft dataset has the largest average spacing gap. On the other hand, Waymo dataset has the smallest average spacing gap, showing that the autonomous vehicles in this dataset are maintaining a smaller following distance, likely due to driving in more congested urban environments. DAS1 and DAS2 datasets have similar spacing gap distributions with an average of around 25 meters. The spacing gap is important in evaluating car-following behavior and can help develop guidelines for setting safe following distances for autonomous vehicles.

**Following speed.** The DAS1 and DAS2 datasets have relatively high following speeds, with data distributions that are centered around  $60\text{ km/h}$  and  $110\text{ km/h}$ . The Waymo dataset, on the other hand, has the lowest following speeds, as the car-following events primarily occur in low-speed driving scenarios in urban streets. However, HighD exhibits high following speeds, with data distributions that are centered around  $80\text{ km/h}$ , which is related to the expressway driving scenario.

**Time gap.** The Lyft dataset exhibits the largest time gap compared to other datasets. This dataset is particularly useful for studying car-following behavior in situations with longer following distances or larger time gaps between vehicles in urban streets. Researchers can analyze this dataset to gain insights into how drivers adjust their behavior during transitions between car-following and free-driving modes. On the other hand, the HighD

dataset is characterized by small time gaps and a concentration of around 1 second. This dataset provides valuable support for studying car-following behavior on expressways, aiming to improve traffic flow and reduce congestion.

**Absolute relative speed.** Most datasets exhibit similar data distributions for absolute relative speed, suggesting that there may be some common driving behaviors across different locations and traffic conditions. However, the Lyft dataset features a different distribution, with most car-following events exhibiting absolute relative speeds in the range of 2 m/s.

**Absolute acceleration.** The statistics indicate a higher likelihood of observing higher acceleration rates in both the Waymo and Lyft datasets. Thus, car-following behaviors captured in these datasets can be characterized as relatively more aggressive. These two datasets can serve as valuable resources for developing and validating car-following models that prioritize safety and conservatism.

**Car-following duration.** The car-following duration in the HighD is set to 15 seconds, while the NGSIM and DAS1 have an average car-following duration exceeding 35 seconds. These long-duration events provide valuable data support for studying long-sequence car-following behavior, enabling us to explore both inter-driver heterogeneity and intra-driver heterogeneity in driving styles.

**Benchmark for car-following behavior modeling.** Here, we provide a brief overview of our benchmark for car-following behavior modeling.

**Data preprocessing.** Car-following events are extracted from the five datasets using the same criteria: (1) having a continuous and unchanged LV ID, and (2) with a minimum duration of 15 seconds. A total of 1930, 16658, 24247, 1440, 24093, and 12540 car-following events were extracted from the NGSIM I-80, SPMD (DAS1), SPMD (DAS2), Waymo, Lyft, and HighD datasets, respectively. Furthermore, each dataset is randomly divided into three parts: the training set, validation set, and test set, with proportions of 70%, 15%, and 15% respectively. These sets are then kept fixed throughout the experiment. However, for traditional models like IDM and GHR, there are different naming conventions in traditional and machine learning (ML) models: ML training and validation corresponds to classical calibration while ML testing corresponds to classical validation. In the ML validation process, it is important to highlight that a complete re-learning is required whenever there is any change in the set of hyperparameters. The validation process involves training the ML model using a specific set of hyperparameters and evaluating its performance on a validation set. The hyperparameters are the tunable parameters that define the behavior and characteristics of the model.

**Baseline models.** We investigate the performances of two traditional car-following models (GHR, IDM) and three data-driven models (NN, LSTM, and DDPG) across the five datasets. These models are optimized using various methods such as genetic algorithms (GA) and hyperparameter adjustments to minimize the Mean Squared Error (MSE) of spacing and reproduce human-like driving behavior.

**Evaluation metrics.** To comprehensively assess the effectiveness of car-following models in maintaining safe, efficient, and comfortable driving behavior, it is important to establish a set of standard evaluation metrics. In this study, we propose four key metrics to evaluate the performance of car-following models:

- Mean square error of spacing: This metric measures the accuracy of a model in reproducing human driving behavior by evaluating the deviation between the modeled and observed spacing between vehicles. A lower mean square error indicates a better match to human driving behavior.
- Collision rate: This metric quantifies the ability of a model to avoid collisions by measuring the frequency of collision events. A lower collision rate indicates a safer car-following model.
- Driving comfort metric (jerk): The jerk metric evaluates the smoothness of vehicle acceleration or deceleration. It measures the rate of change of acceleration and reflects the comfort level experienced by the driver and passengers. A lower jerk value indicates a smoother and more comfortable driving experience.
- Time-To-Collision (TTC) metric: The TTC metric estimates the time it takes for a vehicle to collide with the LV based on their relative positions and velocities. By analyzing TTC values, we can assess the level of safety and potential collision risk. A higher TTC value indicates a safer car-following model.

**Model benchmark performance.** We have presented the MSE of spacing, collision rate, jerk, and TTC as the standard evaluation criteria for assessing the effectiveness of car-following models in maintaining safe, efficient, comfortable driving behavior.

**MSE of spacing.** The performance of the five car-following models, namely IDM, GHR, NN, LSTM, and DDPG, are evaluated on multiple datasets, including NGSIM, HighD, Lyft, Waymo, and SPMD. The corresponding results are presented in Table 2 and visually represented in Fig. 4. It can be concluded that data-driven models, such as NN, LSTM, and DDPG, generally exhibit lower spacing errors compared to traditional models that rely on mathematical equations except for the HighD dataset. Data-driven models can capture complex relationships between input variables, leading to more accurate predictions of car-following behavior. However,

Model	HighD	DAS1	DAS2	Waymo	NGSIM	Lyft
GHR	23.76	549.72	119.92	45.52	62.18	178.01
IDM	27.50	368.95	138.69	39.17	54.97	104.32
NN	24.06	<b>56.01</b>	24.14	16.79	29.00	26.06
LSTM	<b>22.90</b>	70.69	<b>22.49</b>	<b>13.75</b>	<b>23.86</b>	<b>25.71</b>
DDPG	38.52	199.41	89.34	33.91	46.93	86.98

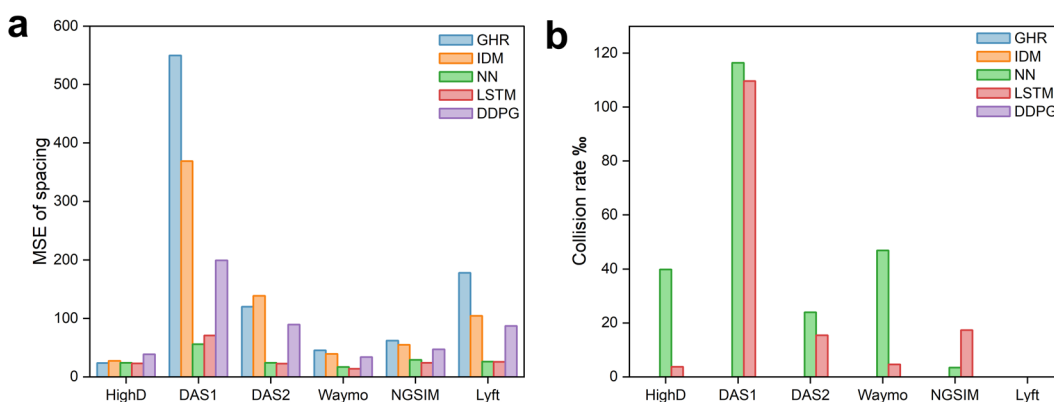
**Table 2.** Test of MSE (spacing).

Model	HighD	DAS1	DAS2	Waymo	NGSIM	Lyft
GHR	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>
IDM	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>
NN	39.87 (75)	116.44 (291)	23.92 (87)	46.29 (10)	3.46 (1)	<b>0</b>
LSTM	3.72 (7)	109.64 (274)	15.39 (56)	4.62 (1)	17.30 (5)	<b>0</b>
DDPG	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>

**Table 3.** Collision rate‰ (Number of Collisions).

Model	HighD	DAS1	DAS2	Waymo	NGSIM	Lyft
GHR	0.34	0.49	0.13	0.42	0.53	0.24
IDM	0.09	0.53	0.19	<b>0.22</b>	0.27	0.29
NN	0.08	<b>0.03</b>	<b>0.08</b>	0.32	0.75	0.15
LSTM	<b>0.04</b>	0.07	<b>0.08</b>	0.24	<b>0.25</b>	<b>0.14</b>
DDPG	33.40	8.75	23.84	7.23	8.04	31.42

**Table 4.** Mean absolute jerk.



**Fig. 4** Model benchmark performance (a) MSE of spacing, (b) Collision rate.

on the HighD dataset, which was collected only on expressways, GHR and IDM can still achieve competitive results in MSE spacing with zero collision. Therefore, traditional models still have their advantages in a single scenario like expressways. The research suggests that while data-driven models generally outperform traditional models, it is important to carefully consider the scenario in question when selecting the appropriate model.

**Collision rate.** A single collision can result in severe consequences, such as injury or death. As shown in Table 3, while traditional models exhibit higher MSE of spacing, they are capable of achieving zero collision when calibrated using GA. Compared to the NN and LSTM models, the DDPG model with a carefully crafted reward function achieves zero collision in all datasets. Notably, all models demonstrate zero collision performance in the Lyft dataset. This might be attributed to the presence of large following gaps in the events captured in this dataset, as shown in Fig. 3.

**Driving comfort.** Jerk measures the smoothness of acceleration changes during car-following. As shown in Table 4, LSTM demonstrates exceptional performance by achieving the lowest jerk values in the HighD, DAS2,



Model	HighD	DAS1	DAS2	Waymo	NGSIM	Lyft
GHR	101.38	31.58	85.53	61.10	14.03	78.70
IDM	<b>384.80</b>	29.69	<b>877.27</b>	40.63	<b>31.10</b>	73.71
NN	60.13	33.83	75.54	35.67	13.34	121.78
LSTM	127.68	69.51	119.77	104.37	13.74	<b>132.47</b>
DDPG	111.66	<b>1107.24</b>	32.98	<b>152.94</b>	12.18	46.47

**Table 5.** Minimum of the TTC.

Model	Test of MSE (spacing)	Collision rate ‰ (Number of Collisions)	Jerk	TTC
GHR	49.28	<b>0</b>	0.26	<b>25.05</b>
IDM	46.84	<b>0</b>	0.14	24.91
NN	33.38	86.29 (17)	0.14	9.91
LSTM	<b>31.23</b>	71.07 (14)	<b>0.13</b>	10.76
DDPG	53.45	<b>0</b>	22.71	22.98

**Table 6.** Model performance comparison on long duration (30 s) car-following events.

NGSIM and Lyft datasets. This indicates that LSTM effectively minimizes abrupt changes in acceleration or deceleration, resulting in smoother car-following behavior. The NN model also performs well in the DAS1 and DAS2 datasets. Notably, IDM, being a classic car-following model, exhibited competitive performance, especially with the lowest jerk value observed in the Waymo dataset. This indicates that IDM maintains smooth acceleration and deceleration profiles, further enhancing driver comfort and ride quality. On the other hand, the DDPG model displays relatively higher jerk values across all datasets, implying a less comfortable car-following behavior.

*Time-To-collision.* In regards to the minimum TTC in Table 5, IDM demonstrates superior performance by achieving the highest values in the HighD, DAS2, and NGSIM datasets. These high TTC values indicate that IDM effectively maintains safe following distances between vehicles, resulting in zero collisions in all datasets. It is worth noting that LSTM achieves the highest minimum TTC value in the Lyft dataset. On the other hand, the DDPG model obtains the highest minimum TTC value in the DAS1 and Waymo datasets.

*Experiment on 30-second car-following behavior study.* To investigate longer car-following durations in driving behavior, we extract 30-second events from the HighD dataset which has high data quality. Comparing our results to the previous 15-second experiment, we observe similar outcomes, as shown in Table 6. The LSTM model achieves the lowest MSE of spacing and jerk, while IDM and NN obtain the second-lowest jerk scores. The GHR model ranks highest in TTC. Moreover, IDM, GHR, and well-designed DDPG models achieve zero collisions, which is crucial in autonomous driving scenarios. This aspect should be emphasized and serves as an area for improvement for LSTM and NN models.

## Discussion

Based on the findings mentioned above, there are several points to discuss regarding the model performance and potential future directions for car-following research.

**Safety.** Although data-driven models achieve lower MSE of spacing compared to traditional models, collisions can still occur. Therefore, it is desirable to develop car-following models that not only achieve lower spacing errors but also have zero collision rates. Incorporating the ability to avoid collisions into data-driven models would significantly enhance their safety and make them suitable for real-world applications.

**Interpretability and generalization ability.** Data-driven car-following models often lack interpretability and generalization ability. Each model requires specific tuning based on the dataset it is trained on. Models trained on a specific dataset, such as NGSIM, may not perform as well when directly applied to a different dataset, like Lyft. This discrepancy suggests that there are underlying variations and characteristics unique to each dataset that impact the car-following behavior. Therefore, it is necessary to develop car-following models that can generalize effectively across different datasets and driving scenarios. Understanding how well models transfer their learned knowledge between datasets can help identify limitations and guide the development of more robust and adaptable car-following models.

**Expectations for future datasets.** To further enhance the performance and realism of car-following models, it is crucial to include additional factors in future datasets. For instance, incorporating data related to road conditions, and traffic signals would provide a more comprehensive understanding of the driving environment. Additionally, integrating information about surrounding vehicles and their behaviors would enable the models

to account for complex interactions and make more accurate predictions. By incorporating these additional data sources, future datasets can better represent real-world driving scenarios, facilitating the development of more robust and effective car-following models.

**More advanced algorithms.** Exploring advanced machine learning techniques can greatly enhance car-following models. For example, employing graph neural networks (GNNs) can help capture the intricate interactions between vehicles in a traffic network, leading to more accurate predictions of car-following behavior. Additionally, generative models can be used to synthesize new car-following events and expand the training data, enabling the models to learn from a larger and more diverse dataset. Moreover, leveraging meta-learning approaches can facilitate the development of adaptive car-following models that can quickly adapt to different datasets and driving conditions. These advanced algorithms have the potential to enhance the performance, adaptability, and scalability of car-following models, paving the way for more efficient and safe autonomous driving systems.

**Mixed traffic flow.** Traffic forecasting is a critical aspect of proactive urban traffic control and management<sup>61,62</sup>. As we accelerate towards the era of autonomous driving, a mixed traffic landscape unfolds, where an amalgamation of autonomous and human-driven vehicles coexist on the roadways<sup>63</sup>. This requires developing car-following models that can effectively handle the interactions between autonomous and human-driven vehicles. Using datasets that include autonomous vehicles for training, particularly the Lyft and Waymo datasets provided in this benchmark, is crucial for the development of car-following models in mixed traffic flows.

**Driving heterogeneity.** The proposed benchmark assumes homogeneous driving behavior, meaning that all vehicles follow similar patterns and behave in a consistent manner. However, in reality, driving behavior can vary significantly among different drivers, vehicle types, and traffic conditions. Therefore, incorporating driving heterogeneity into car-following models requires the development of adaptive algorithms and representative datasets that encompass different driving styles and behaviors, and traffic scenarios.

In this study, we propose FollowNet, the first benchmark of car-following behavior that includes five commonly used real-world datasets. Based on this benchmark, we have analyzed the characteristics of car-following behavior with six dimensions. Additionally, we have evaluated the performance of different car-following models, using MSE of spacing, collision rate, jerk, and TTC as evaluation metrics. The data-driven models, including NN, LSTM, and DDPG, outperform traditional models like IDM and GHR in terms of MSE of spacing. However, traditional models can achieve a zero collision rate when calibrated using GA, emphasizing the safety in car-following models is crucial. Especially, our proposed DDPGs\_Max model achieves competitive performance on the benchmark with a smaller MSE of spacing than traditional models such as IDM and GHR in most datasets, as well as zero collision rate compared to NN and LSTM models. With the exception of some RL models, NN and LSTM models do not have this robustness and generalization abilities. In terms of driving comfort measured by jerk values, NN and LSTM perform exceptionally well by achieving the lowest jerk values in multiple datasets. Also, the classic IDM demonstrates competitive performance, especially with the lowest jerk value observed in the Waymo dataset. However, the DDPG model exhibits relatively higher jerk values, indicating less comfortable car-following behavior. Lastly, IDM consistently achieves higher TTC values in most datasets, indicating effective maintenance of safe following distances and zero collisions. We believe that the establishment of the car-following benchmark with open access to data and source code will enable the development of more accurate, comfortable, and safe models, ultimately contributing to the advancement of microscopic traffic simulation models.

## Methods

This section provides the design of our benchmark. First, we describe the data preprocessing and preparation steps. Second, we introduce the employed traditional models and data-driven models. Lastly, we define the evaluation metrics used in our benchmark.

**Data preparation and preprocessing.** Data processing is to extract and deduce some specific valuable and meaningful data from many disordered and incomprehensible data<sup>64,65</sup>. The extracted car-following events utilizing a methodology similar to prior investigations<sup>12,44,66</sup>. Firstly, to guarantee that the FV follows the same LV throughout the event, the LV's ID should be continuous and unchanged. Secondly, the duration of the car-following event must be 15 seconds or longer to provide sufficient data for analysis. Lastly, the lateral distance between the FV and LV must be less than 2 meters, or the FV and LV must be on the same lane, based on the characteristics of the dataset.

In order to identify the ID of the LV that was followed continuously and record the start and end time of the following event, Algorithm 1 has been applied to all datasets. To address the issue of noise in the initial data when applying datasets, we utilized the Savitzky-Golay filter for smoothing. This filter is a finite impulse response filter that employs polynomial fitting to estimate the underlying signal and mitigate the effect of noise in the time series data.

**Baseline models.** In this benchmark, we trained and tested a total of five car-following models, consisting of two conventional models and three data-driven models. All models were trained with a standardized time duration of 15 seconds, with consistent input parameters including spacing, FV's speed, and relative speed. The acceleration of the FV served as the output for all models. The five baseline models are listed as follows:

**Algorithm 1** Find\_Lead\_Vehilce\_ID ( $x, data\_hz$ ).

---

**Input:**  $x$ : Historical record of leading vehicle ID,  $data\_hz$ : sampling frequency  
**Output:** Following ID, cumulative frame number, starting frame, end frame

- 1: count  $\leftarrow$  1
- 2:  $\mathbb{S} \leftarrow$  empty list { }
- 3:  $k \leftarrow$  0
- 4: **for**  $i \in \text{range}(1, \text{len}(x))$  **do**
- 5:   **if**  $x_i = x_{i-1}$  **then**
- 6:     count  $\leftarrow$  count + 1
- 7:   **else**
- 8:     **if** count >  $Duration * data\_hz$  **then**
- 9:        $(x^k, index_{start}^k, index_{end}^k) \leftarrow (x_{i-1}, i - \text{count}, i - 1)$
- 10:       Append  $(x^k, index_{start}^k, index_{end}^k)$  to  $\mathbb{S} \leftarrow \{ (x^0, index_{start}^0, index_{end}^0), \dots, (x^{k-1}, index_{start}^{k-1}, index_{end}^{k-1}) \}$
- 11:        $k \leftarrow k + 1$
- 12:     **end if**
- 13:     count  $\leftarrow$  1
- 14:   **end if**
- 15: **end for**
- 16: **if** count >  $Duration * data\_hz$  **then**
- 17:    $(x^k, index_{start}^k, index_{end}^k) \leftarrow (\text{Last element in } x, \text{len}(x) - \text{count}, \text{len}(x) - 1)$
- 18:   Append  $(x^k, index_{start}^k, index_{end}^k)$  to  $\mathbb{S} \leftarrow \{ (x^0, index_{start}^0, index_{end}^0), \dots, (x^{k-1}, index_{start}^{k-1}, index_{end}^{k-1}) \}$
- 19:    $k \leftarrow k + 1$
- 20: **end if**
- 21: **if**  $\mathbb{S}$  is empty **then**
- 22:   **return** -1,-1,-1,-1
- 23: **else**
- 24:   **return**  $x^0, index_{end}^0 - index_{start}^0 + 1, index_{start}^0, index_{end}^0$
- 25: **end if**

---

**IDM.** The IDM demonstrates the best predictive performance when compared with other traditional models according to Zhu *et al.*<sup>67</sup>. We proceeded to train this model with the objective of minimizing the MSE of spacing by employing GA with pool calibration (concatenating the calibration data part and finding a single parameter set reflecting the average driver) to determine the most effective IDM parameter set.

**GHR.** The GHR model assumes that drivers adjust their speed based on the distance to the preceding vehicle and their own desired speed. Similar to the IDM, GA was utilized to discover the optimal parameter values<sup>67</sup>.

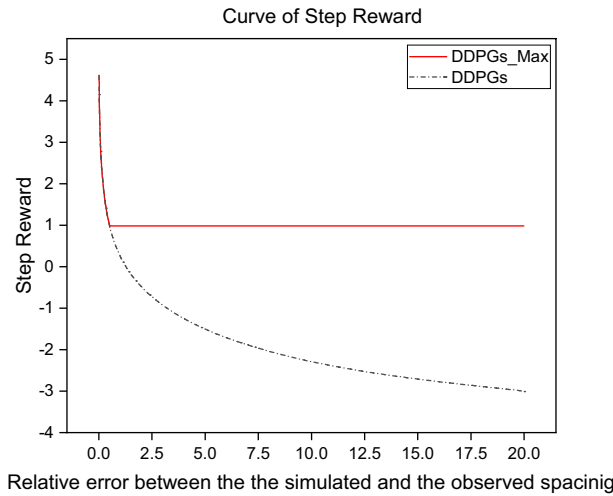
**NN.** To forecast the future acceleration of the FV, a neural network model consisting of three feedforward layers is employed. The Adam optimizer with a learning rate of 0.001 is utilized for optimizing models. Following this, the MSE of spacing is utilized as the loss function for optimizing the network.

**LSTM.** The LSTM model consists of an encoder component and a linear layer. The encoder takes the input data and computes both the encoded representation and the hidden state of the final layer. Subsequently, the linear layer maps the hidden state to the output value, which undergoes a tanh activation function and is multiplied by a constant representing the acceleration limit. In order to enhance the performance of the LSTM model, several hyperparameters have been fine-tuned. These adjustments include the hidden size, the number of LSTM layers, and the utilization of a dropout probability of 0.1 to enhance the model's resilience.

**DDPG.** DDPGs<sup>12</sup> has been proven to be effective in minimizing the discrepancy between the simulated and actual actions in order to reproduce human-like driving. However, the proposed reward function can not be generalized to all datasets. Therefore, a modified reward function  $r_t$  (DDPGs\_Max) with the max operation and collision penalty is proposed in this paper, which can achieve less collision and higher accuracy. It is expressed as below:

$$r_t = \max\left(-\log\left(\left|\frac{S_{n-1,n}(t) - S_{n-1,n}^{obs}(t)}{S_{n-1,n}^{obs}(t)}\right|\right), H\right) + CollisionCheck * Penalty \quad (8)$$

where  $S_{n-1,n}(t)$  and  $S_{n-1,n}^{obs}(t)$  are the simulated and observed spacing at time step  $t$ , respectively.  $H$  is the step reward, which is an adjustable parameter, set to 1 here. The collision check with penalty is set to let the agent learn in the direction of collision reduction in the RL environment. The condition for the end of training is that



**Fig. 5** Reward function of DDPGs\_Max and DDPGs.

a collision occurs or the exploration of the entire *car-following* event is completed. For a single step, the max operation borrowed from the concept of max pooling is used to ensure that the agent can complete the exploration of the entire event as much as possible. In other words, the reward comes from two aspects. Firstly, it is based on how closely the value of the *single-step* simulation matches the observed value. Secondly, it includes a reward for ensuring that there are no collisions during each time step of the simulation., as shown in Fig. 5.

**Evaluation metrics.** Since there is no universally standardized set of evaluation metrics for assessing the effectiveness of a car-following model in maintaining safe, efficient, and comfortable driving behavior, it is essential to establish criteria that effectively capture these key aspects. For this purpose, we utilize MSE of spacing, collision rate, jerk, and TTC as evaluation metrics. A lower MSE score indicates a better alignment between the model and the data, resulting in improved accuracy when predicting the spacing between vehicles. On the other hand, high collision rates suggest that the model is ineffective in avoiding collisions. While a model with a low MSE score can accurately predict vehicle spacing, it may not excel in collision avoidance. For one car-following event, the MSE of spacing can be expressed as:

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N (S_{n-1,n}(t) - S_{n-1,n}^{obs}(t))^2 \quad (9)$$

where  $N$  is the total number of observations, and  $i$  is an observation index.

Similarly, we define collision rate as the number of car-following events where the spacing between vehicles is less than zero divided by the total number of car-following events in the test dataset. The formula for collision rate is as follows:

$$\text{Collision Rate} = \frac{\text{Number of events with spacing} < 0}{\text{Total number of car - following events}} \quad (10)$$

Like the MSE metric, the collision rate is easy to interpret and provides a standardized way to compare the performance of different algorithms across different datasets.

Jerk refers to the rate of change of acceleration over time and reflects the smoothness of vehicle movements. Lower jerk values indicate smoother acceleration and deceleration, leading to a more comfortable driving experience. The jerk metric can be calculated using the formula:

$$\text{Jerk} = \frac{a_n(t) - a_n(t-1)}{dt} \quad (11)$$

where  $a_n(t)$  and  $a_n(t-1)$  represent the acceleration of the FV at time  $t$  and time  $t-1$ , respectively.

Furthermore, TTC represents the time it would take for the two vehicles to collide if they maintained their current states. A higher TTC value indicates a safer following distance between vehicles. The formula for calculating TTC is as follows:

$$\text{TTC}(t) = - \frac{S_{n-1,n}(t)}{\Delta V_{n-1,n}(t)} \quad (12)$$

where  $S_{n-1,n}$  stands for the spacing distance between the FV and the LV, and  $\Delta V_{n-1,n}$  is the relative speed (LV speed – FV's speed).

By incorporating these additional metrics, we can better assess the overall performance of car-following models in terms of accuracy, driving comfort, and safety.

### Data availability

The car-following events extracted from five public driving datasets are available via Figshare<sup>68</sup>.

### Code availability

You can access the car-following data and apply baseline algorithms with trained models, including the parameters of traditional models like IDM and GHR, for each dataset. These datasets and models are publicly available on GitHub at <https://github.com/HKUST-DRIVE-AI-LAB/FollowNet>.

Received: 7 August 2023; Accepted: 1 November 2023;

Published online: 25 November 2023

### References

1. Wang, X., Zhu, M., Chen, M. & Tremont, P. Drivers' rear end collision avoidance behaviors under different levels of situational urgency. *Transp. Res. Pt. C-Emerg. Technol.* **71**, 419–433 (2016).
2. Li, L., Peng, X., Wang, F.-Y., Cao, D. & Li, L. A situation-aware collision avoidance strategy for car-following. *IEEE-CAA J. Automatica Sin.* **5**, 1012–1016 (2018).
3. Groelke, B., Earnhardt, C., Borek, J. & Vermillion, C. A predictive command governor-based adaptive cruise controller with collision avoidance for non-connected vehicle following. *IEEE Trans. Intell. Transp. Syst.* **23**, 12276–12286 (2021).
4. Wang, X., Zhang, X., Guo, F., Gu, Y. & Zhu, X. Effect of daily car-following behaviors on urban roadway rear-end crashes and near-crashes: A naturalistic driving study. *Accid. Anal. Prev.* **164**, 106502 (2022).
5. Gipps, P. G. A behavioural car-following model for computer simulation. *Transp. Res. Pt. B-Methodol.* **15**, 105–111 (1981).
6. Panwai, S. & Dia, H. Comparative evaluation of microscopic car-following behavior. *IEEE Trans. Intell. Transp. Syst.* **6**, 314–325 (2005).
7. Treiber, M. & Kesting, A. Traffic flow dynamics. *Traffic Flow Dynamics: Data, Models and Simulation*, Springer-Verlag Berlin Heidelberg (2013).
8. Zheng, J., Suzuki, K. & Fujita, M. Car-following behavior with instantaneous driver-vehicle reaction delay: A neural-network-based methodology. *Transp. Res. Pt. C-Emerg. Technol.* **36**, 339–351 (2013).
9. Chong, L., Abbas, M. M., Flintsch, A. M. & Higgs, B. A rule-based neural network approach to model driver naturalistic behavior in traffic. *Transp. Res. Pt. C-Emerg. Technol.* **32**, 207–223 (2013).
10. Zhou, M., Qu, X. & Li, X. A recurrent neural network based microscopic car following model to predict traffic oscillation. *Transp. Res. Pt. C-Emerg. Technol.* **84**, 245–264 (2017).
11. Huang, X., Sun, J. & Sun, J. A car-following model considering asymmetric driving behavior based on long short-term memory neural networks. *Transp. Res. Pt. C-Emerg. Technol.* **95**, 346–362 (2018).
12. Zhu, M., Wang, X. & Wang, Y. Human-like autonomous car-following model with deep reinforcement learning. *Transp. Res. Pt. C-Emerg. Technol.* **97**, 348–368 (2018).
13. Zhu, M. *et al.* Safe, efficient, and comfortable velocity control based on reinforcement learning for autonomous driving. *Transp. Res. Pt. C-Emerg. Technol.* **117**, 102662 (2020).
14. Tang, T.-Q., Gui, Y. & Zhang, J. Atac-based car-following model for level 3 autonomous driving considering driver's acceptance. *IEEE Trans. Intell. Transp. Syst.* **23**, 10309–10321 (2021).
15. Li, D. & Okhrin, O. Modified ddpg car-following model with a real-world human driving experience with carla simulator. *Transp. Res. Pt. C-Emerg. Technol.* **147**, 103987 (2023).
16. Krizhevsky, A., Sutskever, I. & Hinton, G. E. Imagenet classification with deep convolutional neural networks. *Communications of the ACM* **60**, 84–90 (2017).
17. Lin, T.-Y. *et al.* Microsoft coco: Common objects in context. In *Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part V 13*, 740–755 (Springer, 2014).
18. Geiger, A., Lenz, P., Stiller, C. & Urtasun, R. Vision meets robotics: The kitti dataset. *International Journal of Robotics Research (IJRR)* (2013).
19. Pipes, L. A. An operational analysis of traffic dynamics. *Journal of applied physics* **24**, 274–281 (1953).
20. Brackstone, M. & McDonald, M. Car-following: a historical review. *Transp. Res. Pt. F-Traffic Psychol. Behav.* **2**, 181–196 (1999).
21. Saifuzzaman, M. & Zheng, Z. Incorporating human-factors in car-following models: a review of recent developments and research needs. *Transp. Res. Pt. C-Emerg. Technol.* **48**, 379–403 (2014).
22. Olstam, J. J. & Tapani, A. *Comparison of Car-following models*, vol. 960 (Swedish National Road and Transport Research Institute Linköping, 2004).
23. Chandler, R. E., Herman, R. & Montroll, E. W. Traffic dynamics: studies in car following. *Operations research* **6**, 165–184 (1958).
24. Gazis, D. C., Herman, R. & Rothery, R. W. Nonlinear follow-the-leader models of traffic flow. *Operations research* **9**, 545–567 (1961).
25. Kometani, E. Dynamic behavior of traffic with a nonlinear spacing-speed relationship. *Theory of Traffic Flow (Proc. of Sym. on TTF (GM))* 105–119 (1959).
26. Michaels, R. Perceptual factors in car-following. *Proc. of 2nd ISTTF (London)* 44–59 (1963).
27. Wiedemann, R. Simulation des strabenverkehrsflusses in schriftenreihe des instituts fir verkehrswesen der universitiit karlsruhe (1974).
28. Bando, M., Hasebe, K., Nakayama, A., Shibata, A. & Sugiyama, Y. Dynamical model of traffic congestion and numerical simulation. *Phys. Rev. E* **51**, 1035 (1995).
29. Helbing, D. & Tilch, B. Generalized force model of traffic dynamics. *Phys. Rev. E* **58**, 133 (1998).
30. Jiang, R., Wu, Q. & Zhu, Z. Full velocity difference model for a car-following theory. *Phys. Rev. E* **64**, 017101 (2001).
31. Treiber, M., Hennecke, A. & Helbing, D. Congested traffic states in empirical observations and microscopic simulations. *Phys. Rev. E* **62**, 1805 (2000).
32. Treiber, M. & Helbing, D. Memory effects in microscopic traffic models and wide scattering in flow-density data. *Phys. Rev. E* **68**, 046119 (2003).
33. Treiber, M., Kesting, A. & Helbing, D. Delays, inaccuracies and anticipation in microscopic traffic models. *Physica A: Statistical Mechanics and its Applications* **360**, 71–88 (2006).
34. Treiber, M. & Kesting, A. The intelligent driver model with stochasticity-new insights into traffic flow oscillations. *Transportation research procedia* **23**, 174–187 (2017).

35. Kanagaraj, V. & Treiber, M. Self-driven particle model for mixed traffic and other disordered flows. *Physica A: Statistical Mechanics and its Applications* **509**, 1–11 (2018).
36. Kurtc, V. & Treiber, M. Simulating bicycle traffic by the intelligent-driver model-reproducing the traffic-wave characteristics observed in a bicycle-following experiment. *Journal of traffic and transportation engineering (English edition)* **7**, 19–29 (2020).
37. He, Z., Zheng, L. & Guan, W. A simple nonparametric car-following model driven by field data. *Transp. Res. Pt. B-Methodol.* **80**, 185–201 (2015).
38. Papatathanasopoulou, V. & Antoniou, C. Towards data-driven car-following models. *Transp. Res. Pt. C-Emerg. Technol.* **55**, 496–509 (2015).
39. Hongfei, J., Zhicai, J. & Anning, N. Develop a car-following model using data collected by “five-wheel system”. In *Proceedings of the 2003 IEEE International Conference on Intelligent Transportation Systems*, vol. 1, 346–351 (IEEE, 2003).
40. Panwai, S. & Dia, H. Neural agent car-following models. *IEEE Trans. Intell. Transp. Syst.* **8**, 60–70 (2007).
41. Khodayari, A., Ghaffari, A., Kazemi, R. & Braunstingl, R. A modified car-following model based on a neural network model of the human driver effects. *IEEE Trans. Syst. Man Cybern. Part A-Syst. Hum.* **42**, 1440–1449 (2012).
42. Yang, D., Zhu, L., Liu, Y., Wu, D. & Ran, B. A novel car-following control model combining machine learning and kinematics models for automated vehicles. *IEEE Trans. Intell. Transp. Syst.* **20**, 1991–2000 (2018).
43. Ma, L. & Qu, S. A sequence to sequence learning based car-following model for multi-step predictions considering reaction delay. *Transp. Res. Pt. C-Emerg. Technol.* **120**, 102785 (2020).
44. Wang, X. *et al.* Capturing car-following behaviors by deep learning. *IEEE Trans. Intell. Transp. Syst.* **19**, 910–920 (2017).
45. Lin, Y. *et al.* Platoon trajectories generation: A unidirectional interconnected lstm-based car-following model. *IEEE Trans. Intell. Transp. Syst.* **23**, 2071–2081 (2020).
46. Chai, C. & Wong, Y. D. Fuzzy cellular automata model for signalized intersections. *Comput.-Aided Civil Infrastruct. Eng.* **30**, 951–964 (2015).
47. Hart, F., Okhrin, O. & Treiber, M. Formulation and validation of a car-following model based on deep reinforcement learning. Preprint at <https://arxiv.org/abs/2109.14268> (2021).
48. Treiber, M. & Kanagaraj, V. Comparing numerical integration schemes for time-continuous car-following models. *Physica A: Statistical Mechanics and its Applications* **419**, 183–195 (2015).
49. Gao, H., Shi, G., Xie, G. & Cheng, B. Car-following method based on inverse reinforcement learning for autonomous vehicle decision-making. *Int. J. Adv. Robot. Syst.* **15**, 1729881418817162 (2018).
50. Zhao, Z. *et al.* Personalized car following for autonomous driving with inverse reinforcement learning. In *2022 International Conference on Robotics and Automation (ICRA)*, 2891–2897 (IEEE, 2022).
51. Song, D., Zhu, B., Zhao, J., Han, J. & Chen, Z. Personalized car-following control based on a hybrid of reinforcement learning and supervised learning. *IEEE Trans. Intell. Transp. Syst.* (2023).
52. Zhu, M. *et al.* Transfollower: Long-sequence car-following trajectory prediction through transformer. Preprint at <https://arxiv.org/abs/2202.03183> (2022).
53. Krajewski, R., Bock, J., Kloeker, L. & Eckstein, L. The highd dataset: A drone dataset of naturalistic vehicle trajectories on german highways for validation of highly automated driving systems. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2118–2125 (IEEE, 2018).
54. Next generation simulation (ngsim) vehicle trajectories and supporting data. *ITS DataHub through Data.transportation.gov*. <https://doi.org/10.21949/1504477> (2016).
55. Bezzina, D. & Sayer, J. Safety pilot model deployment: Test conductor team report. *Report No. DOT HS 812*, 18 <https://data.transportation.gov/Automobiles/Safety-Pilot-Model-Deployment-Data/a7qq-9vfe> (2014).
56. Sun, P. *et al.* Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2446–2454, <https://waymo.com/open/> (2020).
57. Houston, J. *et al.* One thousand and one hours: Self-driving motion prediction dataset. In *Conference on Robot Learning*, 409–418, <https://woven.toyota/en/prediction-dataset> (PMLR, 2021).
58. Krajewski, R., Bock, J., Kloeker, L. & Eckstein, L. The highd dataset. <https://www.highd-dataset.com> (2018).
59. Montanino, M. & Punzo, V. Trajectory data reconstruction and simulation-based validation against macroscopic traffic patterns. *Transp. Res. Pt. B-Methodol.* **80**, 82–106 (2015).
60. Hu, X., Zheng, Z., Chen, D., Zhang, X. & Sun, J. Processing, assessing, and enhancing the waymo autonomous vehicle open dataset for driving behavior research. *Transp. Res. Pt. C-Emerg. Technol.* **134**, 103490 (2022).
61. Yan, H., Ma, X. & Pu, Z. Learning dynamic and hierarchical traffic spatiotemporal features with transformer. *IEEE Transactions on Intelligent Transportation Systems* **23**, 22386–22399 (2021).
62. Ma, X. *et al.* Forecasting transportation network speed using deep capsule networks with nested lstm models. *IEEE Transactions on Intelligent Transportation Systems* **22**, 4813–4824 (2020).
63. Dai, Z., Liu, X. C., Chen, X. & Ma, X. Joint optimization of scheduling and capacity for mixed traffic with autonomous and human-driven buses: A dynamic programming approach. *Transportation Research Part C: Emerging Technologies* **114**, 598–619 (2020).
64. MA, X., LUAN, S. & YU, X. *Traffic Data Management Technology in ITS*, 97–149 (Springer Nature Singapore, Singapore, 2022).
65. Ma, X. & Chen, X. Public transportation big data mining and analysis. In *Data-Driven Solutions to Transportation Problems*, 175–200 (Elsevier, 2019).
66. Zhao, D., Guo, Y. & Jia, Y. J. Trafficnet: An open naturalistic driving scenario library. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, 1–8 (IEEE, 2017).
67. Zhu, M. *et al.* Modeling car-following behavior on urban expressways in shanghai: A naturalistic driving study. *Transp. Res. Pt. C-Emerg. Technol.* **93**, 425–445 (2018).
68. CHEN, X. *et al.* FollowNet: A Comprehensive Benchmark for Car-Following Behavior Modeling, *figshare*, <https://doi.org/10.6084/m9.figshare.c.6777810.v1> (2023).

## Acknowledgements

This study is supported by the National Natural Science Foundation of China under Grant 52302379, Guangzhou Basic and Applied Basic Research Project 2023A03J0106, Guangdong Province General Universities Youth Innovative Talents Project under Grant 2023KQNCX100, and Guangzhou Municipal Science and Technology Project 2023A03J0011.

## Author contributions

Xianda Chen and Meixin Zhu conducted the experiments and analysed the results. All authors reviewed the manuscript.

## Competing interests

The authors declare no competing interests.

### Additional information

**Correspondence** and requests for materials should be addressed to M.Z.

**Reprints and permissions information** is available at [www.nature.com/reprints](http://www.nature.com/reprints).

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2023