Large Language Model-Enhanced Multi-Level Feature Fusion Network for Autonomous Driving Behavior Classification

Xiangyu Li
Departmental of Civil and Environmental Engineering
Northwestern University
Evanston, IL, USA
xiangyuli2027@u.northwestern.edu

Xi Cheng
Departmental of Systems Engineering
Cornell University
Ithaca, NY, USA
xi.cheng@berkeley.edu

Ying Chen*
Departmental of Civil and Environmental Engineering
Northwestern University
Evanston, IL, USA
y-chen@northwestern.edu

***Abstract*—Accurate classification of autonomous vehicle (AV) driving behaviors is critical for optimizing autonomous driving systems, diagnosing operational issues, and enhancing road safety. This paper presents the Large Language Model-Enhanced Multi-Level Feature Fusion Network (LLM-MLFFN), a novel framework designed to address the complexities of multidimensional driving data. The proposed framework integrates priors from large-scale pre-trained models and employs a multimodal approach to enhance classification accuracy. LLM-MLFFN comprises three core components: (1) Multi-Level Feature Extraction Module: Extracts statistical, behavioral, and dynamic features to capture the quantitative aspects of driving behaviors; (2) Semantic Description Module, leverages large language models to transform raw data into high-level semantic features, enhancing interpretability; and (3) Dual-Channel Multi-Modal Feature Fusion Network: Combines numerical and semantic features using weighted attention mechanisms to improve robustness and prediction accuracy. Evaluation on the Waymo Open Trajectory dataset demonstrates the superior performance of LLM-MLFFN, achieving a classification accuracy of 94%, surpassing existing machine learning models. Ablation studies further validate the critical contributions of multimodal fusion, feature extraction strategies, and LLM-derived semantic reasoning. While challenges such as spatial dimension alignment during fusion remain, the framework highlights opportunities for refining attention mechanisms and enhancing computational efficiency. This study represents a significant advancement in AV behavior analysis, paving the way for safer and more efficient autonomous driving systems.**

***Index Terms*—Autonomous Vehicles, Driving Behavior Classification, Large Language Models, Multimodal Feature Fusion, Waymo Open Trajectory Dataset**

I. Introduction

In the past decade, rapid advancements in autonomous vehicle (AV) technologies have significantly transformed the transportation sector. With the potential to enhance road safety, reduce traffic congestion, and improve fuel efficiency, AVs are at the forefront of innovation in the automotive industry. However, despite their promising advantages, fully autonomous systems remain a complex challenge due to the intricacies of real-world driving environments and human-vehicle interactions. According to reports by the National Highway Traffic Safety Administration (NHTSA), approximately 94% of traffic accidents are associated with human errors, such as distraction and impaired judgment [1], and in 2022, distracted driving alone accounted for 3,308 fatalities and an estimated 289,310 injuries [2]. These findings underscore the critical need for safer and more reliable alternatives, such as AVs.

To achieve widespread adoption and seamless integration of AVs into existing traffic systems, bridging the gap between human-like and machine-driven behaviors is essential. Studies suggest that mimicking human driving patterns enables AVs to exhibit behaviors that are more predictable and comprehensible to human road users, fostering smoother interactions [3]. However, this approach poses significant challenges. For instance, the Insurance Institute for Highway Safety (IIHS) found that even with human-like driving behavior and advanced 360-degree sensor systems, AVs would prevent only about one-third of crashes, as many incidents arise from human factors such as speeding and aggressive maneuvers by other drivers [4]. Furthermore, overly conservative driving strategies by AVs, while potentially reducing fatal crashes, can lead to increased rear-end collisions and traffic bottlenecks in mixed-flow environments, particularly in complex scenarios such as intersections and four-way stops [5]. These findings highlight the delicate balance required in designing AV behavior models that align with human expectations while maintaining safety and efficiency.

An essential aspect of AV development lies in understanding and classifying driving behaviors, a critical area of research for improving road safety and enabling intelligent transportation systems [3], [5], [6]. Effective classification of AV driving be-

havior is crucial for optimizing driving algorithms, diagnosing operational malfunctions, assessing system safety and reliability, and building public trust. Recent advancements include semantic interaction models for predicting driving behavior [7], conditional imitation learning for end-to-end driving [8], multimodal trajectory prediction using deep networks [9], and cooperative perception frameworks leveraging vehicle-to-infrastructure (V2X) datasets [10]. Moreover, the ability of AVs to emulate human-like driving behaviors is critical for achieving smooth integration into mixed traffic environments, where human drivers often rely on implicit cues to interpret the actions of surrounding vehicles.

Despite considerable progress in this domain, current methods exhibit notable limitations. Advanced machine learning and deep learning models, including Random Forests [11], Convolutional Neural Networks (CNNs) [12], Long Short-Term Memory networks (LSTMs) [13], and Transformer models [14], have been widely applied in classifying driving behavior of AVs, leveraging their capabilities to process diverse and complex data for accurate behavior prediction. However, these models often struggle to manage the inherent complexities of large-scale multimodal datasets and fail to capture nuanced patterns of AV driving behavior [15]. Additionally, existing approaches predominantly focus on human driver behavior classification or the short-term trajectory prediction of AVs, leaving the broader, more stable behavioral traits of AVs underexplored. These gaps underscore the need for novel frameworks that can integrate multimodal data sources and leverage advanced methodologies to achieve robust and accurate driving behavior classification.

Recent breakthroughs in large language models (LLMs), such as GPT-4 [16], offer transformative potential in driving behavior classification. However, the direct application of LLMs to provide answers often results in low accuracy. Instead, this research focuses on feature space alignment and weighting, allowing LLMs to play a complementary role in multimodal feature fusion. By assigning higher weights to extracted features from other modalities and leveraging LLMs for converting numerical features into semantic dimensions, the proposed methodology achieves deep-level feature enhancement while maintaining robust classification performance.

This paper introduces the **Large Language Model-Enhanced Multi-Level Feature Fusion Network (LLM-MLFFN)**, a novel framework designed to address the aforementioned challenges. By incorporating the strengths of LLMs [17], [18] and multimodal data fusion techniques [19]–[21], this research provides a comprehensive and robust approach to AV driving behavior classification. Specifically, the proposed LLM-MLFFN framework integrates statistical, behavioral, and dynamic features with high-level semantic descriptions generated by LLMs. This integration enables a deeper understanding of driving behavior and enhances the accuracy and interpretability of classification results.

The novelty of this paper lies in its unique approach to combining priors from large-scale pre-trained models with multimodal feature fusion. The LLM-MLFFN leverages the semantic reasoning capabilities of LLMs to transform raw numerical data into high-level semantic representations, which are then fused with numerical features using a dual-channel architecture. This multimodal fusion strategy allows the model to capture both quantitative and qualitative aspects of AV behavior, addressing the limitations of existing methods.

The proposed LLM-MLFFN framework consists of three key modules:

1) *Multi-Level Feature Extraction Module:* This module extracts a comprehensive range of features, including basic statistical features (e.g., mean, standard deviation, kurtosis), driving behavior features (e.g., acceleration change rate, number of hard brakes), and dynamic features (e.g., speed-acceleration correlation). These features provide detailed quantitative insights into AV behavior.
2) *Semantic Description Module:* Leveraging LLMs, this module transforms raw data into high-level semantic features through techniques like one-shot learning and prompt engineering. The semantic features enhance the interpretability of driving behavior, offering rich contextual understanding.
3) *Dual-Channel Multi-Modal Feature Fusion Network:* This module integrates statistical and semantic features using weighted attention mechanisms. The dual-channel design ensures effective and balanced information fusion, improving the accuracy of classification performance.

This paper offers several key contributions to the field of AV behavior analysis:

- *Novel Integration of LLMs in AV Behavior Classification:* By utilizing LLMs to generate semantic features, the study bridges the gap between numerical data and high-level semantic reasoning, offering a holistic understanding of AV driving behavior.
- *Comprehensive Multimodal Feature Fusion Framework:* The dual-channel architecture seamlessly combines statistical and semantic features, addressing the limitations of traditional single-modal approaches.

The remainder of this paper is organized as follows: Section II reviews the related literature and existing methodologies for the classification of driving behavior. Section III details the proposed LLM-MLFFN framework, including its architecture and implementation. Section IV presents the experimental evaluation and results, and Section V concludes with discussions on future research directions.

II. Literature Review

The classification and analysis of driving behaviors are integral to advancing AV technologies. While early studies relied on single-modality approaches, recent advancements in multimodal learning and the integration of large-scale pre-trained models have opened new avenues for understanding complex driving behaviors. This section reviews the evolution of driving behavior classification methodologies, emphasizing the progression from single-modality methods to multimodal fusion and the transformative role of LLMs.

A. Single-Modality Approaches

1) Vision-Based Methods: Vision-based methods utilize camera systems, either in-cabin or external, to capture visual cues relevant to driving behaviors. In-cabin cameras monitor driver-related metrics such as facial expressions, gaze, and head posture, aiding in detecting fatigue or distraction [22]–[25]. External cameras analyze environmental factors, including lane deviations and interactions with traffic, providing critical context for behavior analysis [26], [27]. While effective in diverse scenarios, vision-based methods require substantial computational resources and are sensitive to environmental conditions such as lighting and weather [28], limiting their robustness in real-world applications.

2) Sensor-Based Methods: Sensor-based approaches rely on vehicle-integrated sensors, such as accelerometers, gyroscopes, GPS, and speedometers, to capture quantitative driving data. These methods excel in detecting abrupt maneuvers, aggressive driving, and compliance with speed regulations. Algorithms like Support Vector Machines (SVMs), Random Forests (RFs), LSTM networks, and Graph Neural Networks (GNNs) have been widely employed for sensor data analysis [29]–[31]. However, despite their accuracy and resilience to environmental factors, sensor-based methods often lack semantic richness, limiting their ability to capture higher-order contextual behaviors.

3) Smartphone-Based Methods: The proliferation of smartphones has enabled portable and cost-effective methods for driving behavior analysis. By leveraging embedded sensors, such as accelerometers and GPS, smartphone-based systems can monitor behaviors in real time [32], [33]. However, these methods suffer from variability in sensor quality across devices, limited battery life, and potential privacy concerns [34]. Although accessible, they are less effective for high-precision tasks and AV-specific applications.

B. Multimodal Fusion Frameworks

The limitations of single-modality approaches have spurred the development of multimodal fusion frameworks, which integrate data from multiple sources to achieve comprehensive behavior analysis. By combining quantitative data (e.g., sensor readings) with qualitative data (e.g., images or text), multimodal approaches aim to capture complementary aspects of driving behavior.

1) Fusion Strategies: Fusion strategies can be broadly categorized into early fusion, where data is combined at the input level, and late fusion, where features are merged after independent processing. Recent advancements have introduced weighted attention mechanisms to prioritize significant features during the fusion process, enhancing model performance [35]. These strategies aim to balance contributions from diverse modalities, addressing issues of feature imbalance and spatial misalignment.

2) Advances in Multimodal Learning: Deep learning architectures, such as multi-stream CNNs and attention-based transformers [36], have demonstrated promise in multimodal learning. For instance, the integration of vision and sensor data has proven effective in identifying complex behaviors, such as evasive maneuvers or high-speed decision-making [19]. However, existing frameworks often treat multimodal data streams as separate entities, neglecting semantic interactions that could enrich behavior classification.

C. Emergence of Language Models

The integration of semantic reasoning through LLMs represents a significant advancement in driving behavior classification. LLMs such as GPT-4 [16], LLaMA [37] and PaLM [38] have shown the ability to extract high-level semantic features from structured and unstructured data. By converting raw numerical data into descriptive semantic representations, LLMs bridge the gap between quantitative and qualitative analysis. In this research, LLMs are not used to directly infer driving behaviors, as such applications often yield low accuracy. Instead, the focus is on feature space alignment and weighting. LLMs are leveraged to transform sensor and vision-derived data into semantic dimensions, allowing for meaningful feature fusion. By assigning higher weights to numerical features from other modalities and lower weights to LLM-extracted features, this approach balances the contributions of each modality and mitigates the risk of feature dominance. This strategy enhances the interpretability and robustness of the multimodal system while achieving deep-level feature reinforcement.

D. Filling the Gaps in Existing Studies

Despite significant advancements, several gaps persist in the existing literature on driving behavior classification. Vision-based methods are often resource-intensive and susceptible to environmental disruptions, while sensor- and smartphone-based methods focus on isolated quantitative features, neglecting semantic richness. Multimodal frameworks, though promising, frequently fail to capture semantic interactions between modalities, treating each data type as static and independent. Additionally, the capabilities of LLMs for semantic enrichment have not been fully leveraged in this domain.

This research introduces the **LLM-MLFFN** to address these gaps. The framework integrates the quantitative precision of sensor data with the semantic reasoning capabilities of LLMs, employing a dual-channel architecture that aligns and weights features for optimal fusion. Key advancements include:

- Seamless integration of numerical and semantic features, capturing both quantitative and qualitative aspects of driving behavior.
- Robust multimodal fusion achieved through weighted attention mechanisms and feature alignment.
- Improved classification accuracy and interpretability, setting a new benchmark for AV behavior analysis.

This approach not only bridges the gap between single-modality and multimodal frameworks but also establishes a pathway for integrating advanced semantic reasoning into AV behavior classification.

III. Methodology

We elaborate the methodology in detail, presenting the overall architecture of the LLM-MLFFN model and its key modules.

A. Overall Architecture of LLM-MLFFN

As illustrated in Figure 1, the LLM-MLFFN model architecture integrates numerical and semantic features to enhance classification performance.The general architecture of the LLM-MLFFN model adopts a dual channel multimodal feature fusion strategy, effectively integrating textual and numerical features. The model consists of three core modules: the multilevel feature extraction module, the LLM semantic description module, and the dual channel multimodal feature fusion network.

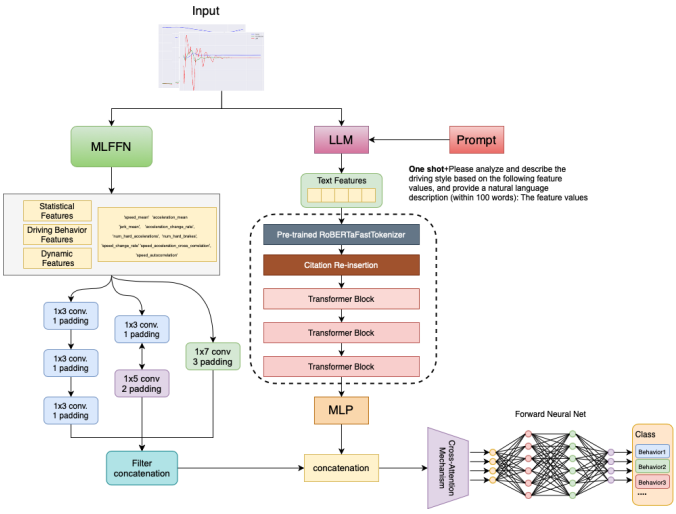

Fig. 1. The overall architecture of the LLM-MLFFN model.

First, the Multi-Level Feature Extraction Module captures key numerical features of driving behavior by extracting statistical and dynamic features from driving data, providing a comprehensive description of various dimensions of driving behavior. Second, the LLM Semantic Description Module leverages LLMs to transform raw driving data into high-level semantic representations. Using natural language processing techniques, this module captures contextual and semantic information in driving behavior, enabling the understanding of complex patterns and subtle nuances within the data. Finally, the Dual-Channel Multimodal Feature Fusion Network combines numerical features and semantic descriptions, improving the model's performance in complex driving behavior classification tasks.

The architectural design of the LLM-MLFFN model is centered on leveraging the complementary strengths of numerical and textual features to enhance driving behavior classification. Numerical features provide precise quantitative insights, while textual features contribute enriched semantic context. By employing an efficient feature fusion strategy, the model effectively integrates and reinforces information, thereby improving its capacity to comprehend and predict driving behaviors. This dual-channel approach establishes a robust framework capable of addressing the complexities of driving behavior classification in diverse and intricate environments.

Additionally, the flow diagram of the LLM-MLFFN model is presented in Figure 2, outlining the sequential steps in the process. The details of these steps are elaborated in the subsequent subsections of this section.

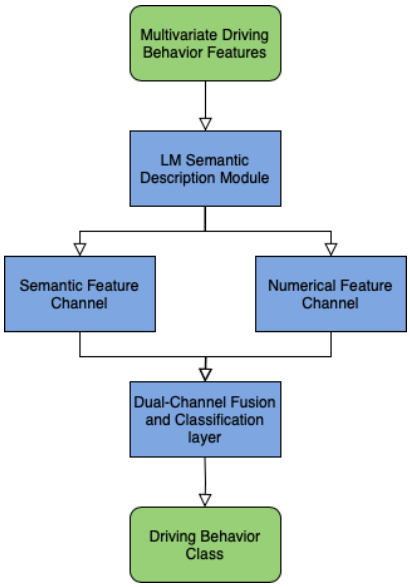

Fig. 2. Flow Diagram of the LLM-MLFFN Model for Driving Behavior Classification.

B. Multi-Level Feature Extraction Module

The Multi-Level Feature Extraction Module constitutes the foundational component of the LLM-MLFFN framework, designed to systematically extract a wide range of informative features from raw driving behavior data. This module aims to provide a holistic representation of driving behavior by capturing its various dimensions. Specifically, the module focuses on three key categories of features: basic statistical features, which summarize the overall data distribution; driving behavior features, which quantify specific driving actions and patterns; and dynamic features, which capture temporal dependencies and interactions within the data. By integrating these diverse feature sets, the module ensures a robust and comprehensive description of driving behaviors, forming a critical basis for downstream semantic processing and classification tasks.

1) Basic Statistical Features: Basic statistical features provide a comprehensive summary of raw driving behavior data, encapsulating key characteristics such as central tendency, variability, and distribution shape. These features serve as a foundation for understanding the dataset's intrinsic properties and are defined as follows:

- *Mean (μ_i):* Represents the average value of the data, serving as an indicator of central tendency:

$$\mu_i = \frac{1}{T}\sum_{t=1}^{T} F_i(t). \tag{1}$$

- *Standard Deviation (σ_i):* Measures the dispersion or variability of the data around the mean, providing insights into its spread:

$$\sigma_i = \sqrt{\frac{1}{T}\sum_{t=1}^{T} \left(F_i(t) - \mu_i\right)^2}. \tag{2}$$

- *Maximum Value ($F_{i,max}$):* Indicates the highest observed value within the dataset, reflecting its upper boundary:

$$F_{i,\text{max}} = \max_t F_i(t). \tag{3}$$

- *Minimum Value ($F_{i,min}$):* Represents the lowest observed value in the dataset, reflecting its lower boundary:

$$F_{i,\text{min}} = \min_t F_i(t). \tag{4}$$

- *Median ($F_{i,median}$):* Highlights the midpoint of the dataset, offering a robust measure of central tendency less sensitive to outliers:

$$F_{i,\text{median}} = \text{median}(F_i). \tag{5}$$

- *25th Percentile ($F_{i,quantile25}$):* Represents the value below which 25% of the data falls, providing insights into the lower quartile of the distribution:

$$F_{i,\text{quantile25}} = \text{quantile}(F_i, 0.25). \tag{6}$$

- *75th Percentile ($F_{i,quantile75}$):* Indicates the value below which 75% of the data lies, offering information about the upper quartile of the distribution:

$$F_{i,\text{quantile75}} = \text{quantile}(F_i, 0.75). \tag{7}$$

- *Kurtosis (κ_i):* Measures the sharpness of the distribution's peak, providing information about the presence of heavy tails or outliers:

$$\kappa_i = \text{kurtosis}(F_i). \tag{8}$$

- *Skewness (γ_i):* Quantifies the asymmetry of the data distribution, distinguishing between left-skewed and right-skewed patterns:

$$\gamma_i = \text{skewness}(F_i). \tag{9}$$

These individual features are then aggregated into a comprehensive statistical feature vector:

$$\begin{aligned}\mathbf{F}_{\text{stat}} = [&\mu_i, \sigma_i, F_{i,\text{max}}, F_{i,\text{min}}, F_{i,\text{median}},\\ &F_{i,\text{quantile25}}, F_{i,\text{quantile75}}, \kappa_i, \gamma_i]_{i=1}^{N}.\end{aligned} \tag{10}$$

By analyzing these basic statistical features, the model effectively captures central tendencies, variability, and distributional characteristics of the driving behavior data. This foundational understanding enables the identification of critical patterns that inform subsequent modeling and classification tasks with enhanced accuracy and robustness.

2) Driving Behavior Features: The driving behavior features focus on capturing specific actions and responses during driving, such as the *acceleration change rate* ρ_a, *number of hard accelerations* N_{accel}, *number of hard brakes* N_{brake}, *number of hard turns* N_{turn}, and *speed change rate* ρ_v. By setting predefined thresholds and calculating corresponding statistical metrics, these features effectively identify and quantify aggressive driving behaviors and habits.

$$\rho_a = \frac{1}{T-1}\sum_{t=2}^{T} |a(t) - a(t-1)|, \tag{11}$$

$$N_{\text{accel}} = \sum_{t=1}^{T} \mathbb{I}(a(t) > 2), \tag{12}$$

$$N_{\text{brake}} = \sum_{t=1}^{T} \mathbb{I}(a(t) < -2), \tag{13}$$

$$N_{\text{turn}} = \sum_{t=1}^{T} \mathbb{I}(|j(t)| > 2), \tag{14}$$

$$\rho_v = \frac{1}{T-1}\sum_{t=2}^{T} |v(t) - v(t-1)|, \tag{15}$$

In the equations above, $\mathbb{I}(\cdot)$ represents the indicator function, which equals 1 if the condition is satisfied and 0 otherwise. These metrics are combined to form the driving behavior feature vector:

$$\mathbf{F}_{\text{behavior}} = [\rho_a, N_{\text{accel}}, N_{\text{brake}}, N_{\text{turn}}, \rho_v]. \tag{16}$$

3) Dynamic Features: Dynamic features delve deeper into the temporal dependencies and dynamic patterns of driving behaviors, providing a detailed analysis of their variations over time. These features include *speed-acceleration cross-correlation* $\rho_{v,a}$, *acceleration-jerk cross-correlation* $\rho_{a,j}$, *speed autocorrelation* ρ_v^{auto}, and *acceleration autocorrelation* ρ_a^{auto}. By calculating the correlations between variables at different time points, these features reveal the sequential dependencies and dynamic characteristics inherent in driving behaviors, offering more nuanced and comprehensive insights for subsequent classification tasks.

$$\rho_{v,a} = \text{corr}(v, a), \tag{17}$$

$$\rho_{a,j} = \text{corr}(a, j), \tag{18}$$

$$\rho_v^{\text{auto}} = \text{autocorr}(v), \tag{19}$$

$$\rho_a^{\text{auto}} = \text{autocorr}(a). \tag{20}$$

Here, $\text{corr}(x, y)$ represents the Pearson correlation coefficient between x and y, and $\text{autocorr}(x)$ denotes the autocorrelation of x. These statistical measures highlight the interplay between various dynamic factors influencing driving behaviors.

The dynamic feature vector aggregates these metrics as follows:

$$\mathbf{F}_{\text{dynamic}} = [\rho_{v,a}, \rho_{a,j}, \rho_v^{\text{auto}}, \rho_a^{\text{auto}}]. \tag{21}$$

4) Feature Processing and Preparation: The extracted features, comprising basic statistical features, driving behavior features, and dynamic features, are combined, normalized, and prepared for downstream classification tasks in a structured and systematic manner.

Feature Output Vector: The feature extraction process yields a comprehensive feature vector by concatenating the three types of features:

$$\mathbf{F} = [\mathbf{F}_{\text{stat}}, \mathbf{F}_{\text{behavior}}, \mathbf{F}_{\text{dynamic}}] \in \mathbb{R}^D, \tag{22}$$

where D represents the dimensionality of the vector, calculated as:

$$D = 9N + 5 + 4, \tag{23}$$

with N denoting the number of numerical features extracted from the dataset. In this study, $N = 34$, resulting in a feature vector dimension $D = 310$.

Feature Normalization: To ensure uniform scaling and comparability across features, the feature vector is normalized to standardize its values. This process mitigates the influence of feature scale on the model and enhances its convergence properties during training. Normalization is performed as follows:

$$F_{\text{scaled}} = \frac{F - \mu_F}{\sigma_F}, \tag{24}$$

where μ_F and σ_F are the mean and standard deviation of feature F, respectively, computed from the training data. This step centers the features at zero and scales them to unit variance, ensuring that all features contribute equally during learning.

Feature Export: The normalized feature vector, $\mathbf{F}_{\text{scaled}}$, is exported in a structured format as rows of a `DataFrame`, accompanied by the corresponding labels for supervised learning. This comprehensive dataset, containing normalized features and ground truth labels, is optimized for efficient use in model training and evaluation.

By integrating these steps into a unified framework, the feature processing pipeline ensures high-quality input for classification models, thereby enhancing the accuracy, robustness, and interpretability of driving behavior predictions.

C. Feature Analysis

In the analysis of driving style features, we examined and interpreted the feature distributions of various driving behavior types (Aggressive, Assertive, Conservative, and Moderate) across multiple dimensions, as illustrated in Fig. 3. This analysis yielded several critical insights:

For the $speed\_mean$ feature, the distributions of Aggressive and Assertive driving styles skew toward higher average speeds, indicating a preference for faster driving. Notably, the Aggressive style exhibits significantly higher speed averages compared to other categories. In contrast, Conservative and Moderate styles are concentrated in lower-speed regions, consistent with the cautious nature of Conservative drivers. This demonstrates that $speed\_mean$ effectively distinguishes between aggressive and conservative driving behaviors, particularly excelling in identifying Aggressive driving.

The $acceleration\_mean$ feature shows substantial overlap across all driving styles, especially near zero. This overlap suggests that this feature offers limited discriminatory power for distinguishing between driving styles, as the mean acceleration appears balanced across all categories, failing to provide sufficient differentiation.

In terms of $jerk\_mean$ (mean rate of change of acceleration), the distributions for Conservative and Moderate driving styles are more concentrated, whereas Aggressive and Assertive styles are more dispersed. This indicates that $jerk\_mean$ effectively captures the smoothness of driving behavior. Conservative drivers exhibit lower jerk values, reflecting smoother driving patterns, while Aggressive drivers show higher jerk values, indicating more abrupt acceleration and deceleration. Thus, jerk_mean demonstrates good sensitivity in differentiating aggressive and conservative driving styles, capturing the volatility of Aggressive driving.

For $acceleration\_change\_rate$, Aggressive and Assertive styles are concentrated in high-change-rate regions, reflecting more frequent and intense acceleration changes. Conservative styles, in contrast, are associated with lower change rates, reflecting steadier driving patterns. This feature is highly effective in distinguishing aggressive from conservative driving behaviors, particularly in highlighting the high variability characteristic of Aggressive driving.

The $num\_hard\_accelerations$ feature provides a clear distinction among driving styles. Aggressive driving exhibits significantly higher counts of hard accelerations compared to other styles, while Conservative and Moderate styles are concentrated in regions with fewer hard accelerations. This indicates that $num\_hard\_accelerations$ is a strong feature for identifying aggressive driving behavior, providing intuitive insights into acceleration habits.

For the num_hard_brakes feature, Aggressive driving styles show notably higher counts of hard braking events, reflecting frequent and abrupt deceleration maneuvers. Conservative and Moderate styles exhibit minimal hard braking, indicating smoother and more stable driving patterns. Consequently, num_hard_brakes emerges as a key feature for identifying aggressive driving behaviors.

The $speed\_change\_rate$ feature is more concentrated in negative regions for Aggressive driving, suggesting frequent deceleration. In contrast, Moderate and Conservative styles are closer to zero, indicating less variability in speed. This highlights the significant fluctuations in speed characteristic of Aggressive driving, making $speed\_change\_rate$ a strong feature for differentiating this style from others.

Regarding $speed\_acceleration\_cross\_correlation$, Conservative and Moderate styles exhibit more concentrated distributions in regions of higher correlation, indicating smoother relationships between speed and acceleration. Aggressive and Assertive styles, on the other hand, show more dispersed distributions, reflecting greater instability in their acceleration-deceleration patterns. This feature is particularly effective in identifying Conservative driving styles, especially in distinguishing between smooth and aggressive behaviors.

Lastly, the $speed\_autocorrelation$ feature reveals concentrated distributions near zero, with Conservative and Moderate styles showing greater concentration in higher autocorrelation regions. This indicates that the feature captures the regularity of driving patterns, particularly reflecting the smoothness of Conservative driving. However, its contribution to distinguishing different driving styles is limited and serves primarily as a supplementary feature.

In conclusion, the features $num\_hard\_accelerations$, num_hard_brakes, $acceleration\_change\_rate$, and $speed\_change\_rate$ are the most effective for differentiating driving styles, particularly between Aggressive and Conservative driving. These features effectively capture acceleration, braking, and speed variation patterns, serving as critical indicators for identifying distinct driving behaviors. While $speed\_mean$ and $jerk\_mean$ provide supplementary information, their overlapping distributions limit their overall contribution to classification.

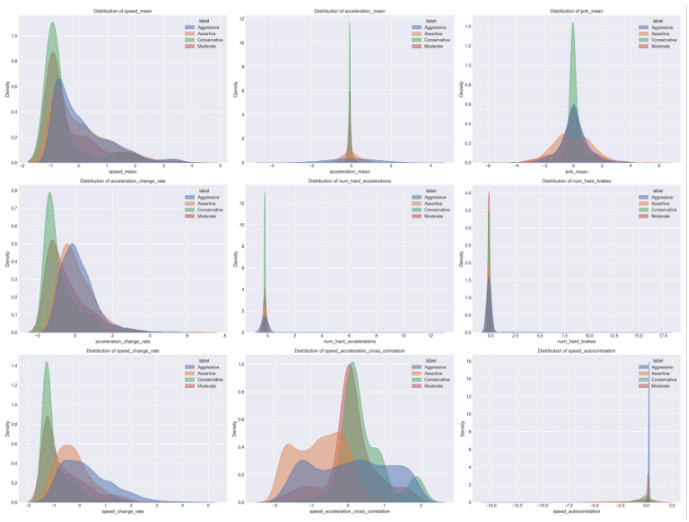

Fig. 3. Feature distributions for different driving behavior types (Aggressive, Assertive, Conservative, and Moderate).

D. LLM Semantic Description Module

The LLM Semantic Description Module leverages LLMs, such as GPT-4, to enhance driving behavior analysis by transforming numerical features into rich semantic descriptions. The primary objective of this module is to bridge the gap between structured numerical data and natural language, enabling deeper insights and improved classification of driving styles.

Given a numerical feature vector $\mathbf{F} = [f_1, f_2, \ldots, f_D]$, where D represents the dimensionality of the feature space, the module constructs a structured prompt $P(\mathbf{F})$. This prompt incorporates all feature values and their meanings in a predefined format, ensuring that the LLM fully understands the context and significance of each feature. The LLM processes the structured prompt and generates a detailed semantic analysis S:

$$S = \text{LLM}(P(\mathbf{F})), \tag{25}$$

where S represents the natural language description of the driving behavior.

The generated analysis S typically includes an overall assessment of driving behavior, specific descriptions of driving patterns, and explanations of key behavioral traits. For instance, the analysis might highlight insights such as the driver's high variability in acceleration change rates or frequent hard acceleration events.

Prompt Design: The prompts are meticulously designed to include all numerical features and their values in a structured format, enabling the LLM to capture the significance and context of each feature. A one-shot prompting approach is used, where the formatted prompt $P(\mathbf{F})$ is fed into the LLM, generating concise yet insightful natural language descriptions S.

Result Integration: The generated semantic analysis S is appended to the original dataset as a high-level semantic feature, providing enriched contextual information for subsequent feature fusion and classification tasks.

By employing this semantic processing methodology, the LLM Semantic Description Module effectively bridges the gap between numerical and semantic data, enabling high-quality multimodal feature integration. This significantly enhances the performance and robustness of the LLM-MLFFN model in complex driving behavior classification tasks. The example prompts are formulated as follows:

Prompt for Driving Behavior Analysis

Example: 1 Aggressive
Feature Values:
acceleration_autocorrelation: 0.498655829
acceleration_change_rate: -0.540905602
etc...

User Instruction:
Please analyze the driving style based on the following feature values and describe it in natural language within 100 words.

LLM (i.e., GPT4o) Response:
The driver exhibits frequent and significant acceleration and deceleration, as indicated by high acceleration autocorrelation and acceleration change rate. The high jerk values and frequent occurrences of hard accelerations, brakes, and turns suggest an aggressive driving style. Additionally, the speed metrics show high and fluctuating speeds, reinforcing the characterization of this driver's style as aggressive. Overall, this driver demonstrates an aggressive driving behavior.

E. Dual-Channel Multi-Modal Feature Fusion Network

The Dual-Channel Multi-Modal Feature Fusion Network seamlessly integrates high-level semantic features derived from the LLM Semantic Description Module with numerical features from the Multi-Level Feature Extraction Module. This

design leverages the complementary strengths of semantic and numerical data, enabling robust and accurate classification of driving behaviors. The network architecture comprises three main components: the Semantic Feature Channel, the Numerical Feature Channel, and the Modal Fusion and Classification layer.

1) Semantic Feature Channel: The Semantic Feature Channel processes high-level semantic features generated from structured prompts. The detailed workflow includes:

- **RoBERTa Encoding:** A pre-trained RoBERTa model (e.g., RoBERTa-base) [39] encodes the natural language input S, yielding a 768-dimensional feature vector:

$$\mathbf{E} = \text{RoBERTa}(S), \quad \mathbf{E} \in \mathbb{R}^{768}. \tag{26}$$

- **Dimensionality Reduction:** The feature vector is reduced to 128 dimensions via a fully connected layer:

$$\mathbf{E}_{\text{mapped}} = \text{ReLU}(\mathbf{W}_s\mathbf{E} + \mathbf{b}_s), \quad \mathbf{E}_{\text{mapped}} \in \mathbb{R}^{128}. \tag{27}$$

- **Regularization:** Dropout regularization prevents overfitting:

$$\mathbf{E}_{\text{final}} = \text{Dropout}(\mathbf{E}_{\text{mapped}}). \tag{28}$$

The resulting semantic feature vector $\mathbf{E}_{\text{final}} \in \mathbb{R}^{128}$ is used for downstream tasks.

2) Numerical Feature Channel: The Numerical Feature Channel processes numerical data through a combination of multi-scale convolutions, attention mechanisms, and deep feature extraction:

- **Multi-Scale Convolutions** [40]: Convolutions with kernel sizes $k \in \{3, 5, 7\}$ extract features at varying temporal resolutions:

$$\mathbf{C}_k = \text{ReLU}(\text{Conv1D}_k(\mathbf{F})), \quad \mathbf{C}_k \in \mathbb{R}^{64\times L}, \tag{29}$$

  where L is the sequence length.
- **Feature Concatenation:** The outputs are concatenated along the channel dimension:

$$\mathbf{C}_{\text{concat}} = \text{Concat}(\mathbf{C}_3, \mathbf{C}_5, \mathbf{C}_7), \quad \mathbf{C}_{\text{concat}} \in \mathbb{R}^{192\times L}. \tag{30}$$

- **Spatio-Temporal Attention:** An attention mechanism emphasizes important features while suppressing noise:

$$\mathbf{Q} = \mathbf{W}_Q\mathbf{C}_{\text{concat}}, \quad \mathbf{K} = \mathbf{W}_K\mathbf{C}_{\text{concat}}, \quad \mathbf{V} = \mathbf{W}_V\mathbf{C}_{\text{concat}}, \tag{31}$$

$$\mathbf{A} = \text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}}\right)\mathbf{V}, \tag{32}$$

  where $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V \in \mathbb{R}^{d_k\times 192}$, and d_k is the attention dimension.
- **Deep Feature Processing:** Two layers of 1D convolution, batch normalization, and ReLU activation refine the attended features:

$$\mathbf{A}_{\text{deep}} = \text{ReLU}(\text{BatchNorm1D}(\text{Conv1D}(\mathbf{A}))). \tag{33}$$

- **Feature Pooling and Projection:** Adaptive max pooling compresses the features into a fixed size, followed by a fully connected layer to project into a 128-dimensional space:

$$\mathbf{F}_{\text{final}} = \text{ReLU}(\mathbf{W}_f\mathbf{F}_{\text{pooled}} + \mathbf{b}_f), \quad \mathbf{F}_{\text{final}} \in \mathbb{R}^{128}. \tag{34}$$

The output $\mathbf{F}_{\text{final}} \in \mathbb{R}^{128}$ is the processed numerical feature vector.

3) Modal Fusion and Classification: The outputs of the Semantic Feature Channel ($\mathbf{E}_{\text{final}}$) and the Numerical Feature Channel ($\mathbf{F}_{\text{final}}$) are concatenated to form a 256-dimensional fused feature vector:

$$\mathbf{F}_{\text{fused}} = \text{Concat}(\mathbf{E}_{\text{final}}, \mathbf{F}_{\text{final}}), \quad \mathbf{F}_{\text{fused}} \in \mathbb{R}^{256}. \tag{35}$$

The fused vector is passed through two fully connected layers for classification:

$$\mathbf{H} = \text{ReLU}(\mathbf{W}_1\mathbf{F}_{\text{fused}} + \mathbf{b}_1), \tag{36}$$

$$\text{Logits} = \mathbf{W}_2\mathbf{H} + \mathbf{b}_2, \tag{37}$$

where $\mathbf{W}_1 \in \mathbb{R}^{256\times 256}$, $\mathbf{b}_1 \in \mathbb{R}^{256}$, $\mathbf{W}_2 \in \mathbb{R}^{K\times 256}$, and $\mathbf{b}_2 \in \mathbb{R}^K$. These logits represent the final classification probabilities across K driving behavior categories.

IV. Numerical Experiments

A. Dataset

The proposed LLM-MLFFN model was trained and evaluated using a trajectory dataset derived from the Waymo Open Dataset [41]. The dataset underwent several preprocessing steps, including outlier removal and denoising, to ensure the quality and reliability of the data. The processed dataset captures three critical features of autonomous vehicle (AV) driving behavior: speed, acceleration, and jerk. These features were utilized comprehensively for both training and testing purposes.

The dataset comprises 2,704 trips, with most trips having a duration of approximately 20 seconds and a recording interval of 0.1 seconds. To enhance data relevance, trips where the speed remained consistently at or below zero were filtered out, resulting in 2,695 meaningful trajectories for analysis. These trajectories provide a robust foundation for understanding and classifying diverse AV driving behaviors.

B. Implementation Pipeline

The implementation of the proposed LLM-MLFFN model involves five key stages: feature extraction, semantic enhancement, feature fusion, training configuration, and model evaluation.

1) Feature Extraction: Feature extraction is performed to quantitatively describe driving behaviors within each time window. It consists of three components:

- **Basic Statistical Features:** Features such as mean and standard deviation are calculated to provide a quantitative summary of the dataset's central tendencies and dispersion.
- **Driving Behavior Features:** Features like the number of hard accelerations and hard brakes are derived by setting

specific thresholds and performing statistical calculations. These features capture explicit driving behavior patterns.
- **Dynamic Features:** By calculating correlations between different time points, such as speed-acceleration cross-correlation, these features capture the temporal dependencies and dynamic changes in driving behavior.

2) Semantic Enhancement: The semantic enhancement process leverages the natural language processing capabilities of LLMs, such as GPT-4, to convert traditional numerical features into high-level semantic descriptions. Specific prompts are designed to generate semantic information that captures the context and patterns underlying driving behavior. For example, a prompt might generate a description like "The driver exhibits frequent hard braking, indicating an aggressive driving style." This transformation enhances the model's understanding of driving behaviors.

3) Feature Fusion: During the feature fusion stage, a cross-attention mechanism is employed to integrate numerical and semantic features. By computing the relationships between these two modalities, the cross-attention mechanism dynamically adjusts the weights of each feature type, thereby improving the model's ability to classify driving behaviors accurately.

4) Training Configuration: The LLM-MLFFN model is trained using supervised learning. The input feature vectors are used during training, with labels guiding backpropagation and parameter updates. The training process is optimized for classification tasks using a cross-entropy loss function. To prevent overfitting, techniques such as Dropout and L2 regularization are employed. The Adam optimizer is utilized to ensure stable and efficient gradient updates, improving the model's convergence speed and stability.

5) Model Evaluation: After training, the model is evaluated using metrics including accuracy, precision, recall, and F1-score. Cross-validation is applied to ensure objectivity in evaluation, with the dataset randomly split into 80% for training, 10% for validation, and 10% for testing. Each metric is computed as follows:

$$\text{Precision} = \frac{TP}{TP + FP}, \tag{38}$$

$$\text{Recall} = \frac{TP}{TP + FN}, \tag{39}$$

$$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}, \tag{40}$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}, \tag{41}$$

where TP, TN, FP, and FN represent the number of true positives, true negatives, false positives, and false negatives, respectively.

This detailed pipeline ensures a robust and systematic approach to training and evaluating the LLM-MLFFN model, achieving high reliability in driving behavior classification tasks.

C. Comparison of Models

The proposed LLM-MLFFN model was compared against several benchmark models commonly used for multivariate time series classification (details are listed in Table I):

- **LSTM:** Long Short-Term Memory networks (LSTM) [42] are recurrent neural networks designed to capture long-term dependencies in sequential data. Leveraging memory cells and gating mechanisms, LSTM has proven effective in tasks involving sequential data, such as time series classification and prediction.
- **MLP:** Multi-Layer Perceptrons (MLP) [43] are feedforward neural networks consisting of multiple interconnected layers of neurons. As a general-purpose model, MLPs have demonstrated strong performance across a variety of tasks, including classification, regression, and function approximation.
- **FCN:** Fully Convolutional Networks (FCN) [43] are primarily designed for tasks such as image segmentation. By replacing fully connected layers with convolutional layers, FCNs can handle input data of varying dimensions, making them adaptable for time series analysis.
- **LSTM-FCN:** The LSTM-FCN hybrid model [44] combines the strengths of LSTM and FCN architectures, using LSTM layers to identify temporal dependencies and FCN layers for efficient feature extraction.
- **GRU-FCN:** Similar to LSTM-FCN, the GRU-FCN hybrid model [45] combines Gated Recurrent Units (GRU) with FCN layers for time series data classification, providing an efficient alternative to LSTM-based models.
- **mWDN:** Multi-Scale Weighted Dense Networks (mWDN) [46] employ multi-scale dilated convolution layers and weighted dense connections to capture both local and global patterns in time series data, ensuring effective classification.
- **MLSTM-FCN:** The Multi-Scale LSTM-FCN hybrid model [47] integrates LSTM and FCN layers with multi-scale processing to capture temporal dependencies across various scales while extracting relevant features.
- **TST:** The Time Series Transformer (TST) [48], based on the Transformer architecture, is specifically designed for time series data. It uses self-attention mechanisms to capture temporal dependencies and has demonstrated strong performance across numerous time series classification tasks.
- **GAF-ViT:** The GAF-ViT model [33] combines attention mechanisms with Vision Transformers (ViT) to address multivariate time series classification. By leveraging graph structures and visual models, GAF-ViT effectively exploits spatial and temporal features, leading to improved classification accuracy.

Feature-Engineered vs. Non-Feature-Engineered Models: Experiments were conducted to compare the performance of feature-engineered models (those utilizing the proposed multi-level feature extraction module) with non-feature-engineered models. Results clearly indicate that feature-

TABLE I
COMPARISON OF FEATURE-ENGINEERED AND NON-FEATURE-ENGINEERED MODELS ACROSS DIFFERENT METRICS

Model	Accuracy		Precision		Recall		F1-Score	
	Non-Feat.	Feat.	Non-Feat.	Feat.	Non-Feat.	Feat.	Non-Feat.	Feat.
LSTM	0.7166	0.8888	0.6227	0.8925	0.4836	0.8888	0.4955	0.8895
MLP	0.8321	0.8824	0.8584	0.8829	0.6721	0.8824	0.7394	0.8812
FCN	0.8075	0.7519	0.7915	0.7615	0.6540	0.7519	0.7040	0.6943
LSTM-FCN	0.8032	0.8909	0.8080	0.8981	0.6334	0.8909	0.6940	0.8934
GRU-FCN	0.6909	0.8877	0.5536	0.8955	0.4554	0.8877	0.4782	0.8893
mWDN	0.9005	0.8684	0.8595	0.8801	0.8224	0.8684	0.8385	0.8703
MLSTM-FCN	0.8182	0.8299	0.8003	0.8409	0.6843	0.8299	0.7311	0.8140
TST	0.7508	0.7701	0.7896	0.7622	0.4985	0.7701	0.5586	0.7347
GAF-ViT	0.9209	0.9219	0.8679	0.8800	0.8826	0.8900	0.8747	0.8850
LLM-MLFFN	0.9145	**0.9430**	0.9158	**0.9464**	0.9145	**0.9430**	0.9135	**0.9414**

engineered models significantly outperform their non-feature-engineered counterparts across all evaluation metrics, as demonstrated in Fig. 4. This highlights the effectiveness of the proposed feature extraction module.

Performance Comparison: The performance of LLM-MLFFN and benchmark models was evaluated based on accuracy, precision, recall, and F1-score, as shown in Fig. 5. Experimental results demonstrate that the proposed LLM-MLFFN model outperforms all baseline models across all evaluation metrics. Notably, the LLM-MLFFN model exhibits superior performance under both feature-engineered and non-feature-engineered settings, confirming its robust multi-level feature extraction and fusion capabilities. While models like GAF-ViT and mWDN also perform well, LLM-MLFFN achieves higher precision and stability across various time series tasks due to its deep feature extraction and semantic modeling capabilities.

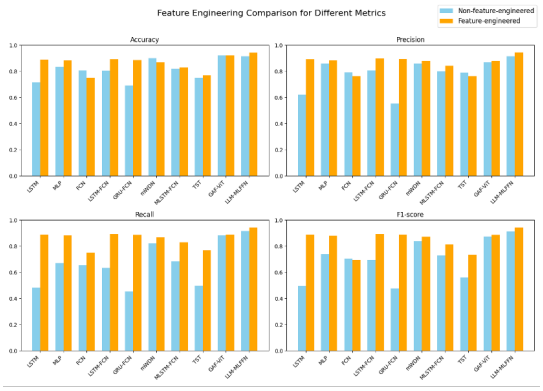

Fig. 4. Comparison of models before and after incorporating the multi-level feature extraction module.

D. Ablation Study

To evaluate the contribution of each component in the LLM-MLFFN model, we conducted an ablation study by systematically removing or replacing key modules within the framework. The results, summarized in Table II, highlight the significance of the individual components in improving the overall performance.

The removal of the spatio-temporal attention mechanism resulted in a marked decline in performance, with reductions

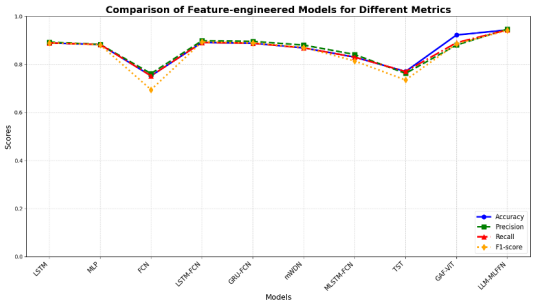

Fig. 5. comparison of different models across metrics: accuracy, precision, recall, and F1-score.

observed across all evaluation metrics, including accuracy, precision, recall, and F1-score. This demonstrates the critical role of the spatio-temporal attention mechanism in capturing the temporal and spatial dependencies present in the data.

Similarly, excluding the multi-scale convolution module led to a further decline in performance, particularly in precision and F1-score. These results underscore the importance of multi-scale convolutions in extracting multi-level features and enhancing the model's understanding of nuanced driving behaviors.

When the model was tested using only semantic features or numerical features individually, the classification performance was significantly lower than the complete model. This was especially evident in precision and recall, which further validates the necessity of fusing numerical and semantic features. The integration of these features enables the model to leverage the strengths of both modalities, achieving a comprehensive representation of driving behaviors.

Overall, the ablation study highlights the effectiveness of each module within the LLM-MLFFN model. The spatio-temporal attention mechanism, multi-scale convolution module, and the integration of numerical and semantic features collectively contribute to the superior classification performance of the model. These findings reinforce the advantages of the dual-channel multi-modal feature fusion strategy in addressing the challenges of complex driving behavior classification.

TABLE II
ABLATION STUDY RESULTS

ID	Description	Accuracy	Precision	Recall	F1 Score
Baseline	*Complete Model*	*0.9430*	*0.9464*	*0.9430*	*0.9414*
Experiment 1	Remove the Spatiotemporal Attention Mechanism	0.9311	0.9333	0.9311	0.9298
Experiment 2	Remove Multiscale Convolution	0.9359	0.9409	0.9359	0.9343
Experiment 3	Use Only Text Features	0.9145	0.9158	0.9145	0.9135
Experiment 4	Use Only Numerical Features	0.9144	0.9161	0.9144	0.9147

V. CONCLUSION

This study proposes the **Large Language Model-Enhanced Multi-Level Feature Fusion Network (LLM-MLFFN)** for autonomous driving behavior analysis, aiming to achieve a comprehensive understanding and classification of driving behaviors by integrating statistical, behavioral, and dynamic features. Through the design of a multi-level feature extraction module, a semantic description module, and a dual-channel multi-modal feature fusion network, the proposed framework significantly improves the accuracy and classification performance of autonomous driving behavior analysis.

Experimental results on the Waymo Open Trajectory dataset demonstrate that the LLM-MLFFN model outperforms existing methods across multiple evaluation metrics, showcasing its superiority in capturing the complexity of autonomous vehicle behaviors. Ablation studies further validate the critical contributions of the multi-level feature extraction and semantic description modules, while confirming the effectiveness of the dual-channel architecture in leveraging the complementary strengths of numerical and semantic features. The proposed framework effectively combines numerical and textual features, offering a holistic approach to understanding and analyzing complex driving behaviors.

Future research should explore the application of the proposed method in more complex traffic environments and investigate further optimization of the model's real-time processing capabilities and computational efficiency. Incorporating additional types of sensor data and more diverse driving scenarios will enhance the robustness and generalizability of the model. These directions represent critical steps toward advancing the applicability and scalability of the LLM-MLFFN framework, ensuring its relevance to real-world autonomous driving systems.

ACKNOWLEDGMENT

REFERENCES

[1] National Highway Traffic Safety Administration (NHTSA). Critical reasons for crashes investigated in the national motor vehicle crash causation survey. Technical report, U.S. Department of Transportation, 2015.

[2] National Highway Traffic Safety Administration (NHTSA). Distracted driving 2022 statistics. Technical report, U.S. Department of Transportation, 2022.

[3] Wilko Schwarting, Alyssa Pierson, Javier Alonso-Mora, Sertac Karaman, and Daniela Rus. Social behavior for autonomous vehicles. *Proceedings of the National Academy of Sciences*, 116(50):24972–24978, 2019.

[4] Insurance Institute for Highway Safety (IIHS). Safety benefits of automated vehicles could be limited by failure to eliminate most crashes, 2020.

[5] Khan Muhammad, Amin Ullah, Jaime Lloret, Javier Del Ser, and Victor Hugo C de Albuquerque. Deep learning for safe autonomous driving: Current challenges and future directions. *IEEE Transactions on Intelligent Transportation Systems*, 22(7):4316–4336, 2020.

[6] Mozhgan Nasr Azadani and Azzedine Boukerche. Driving behavior analysis guidelines for intelligent transportation systems. *IEEE transactions on intelligent transportation systems*, 23(7):6027–6045, 2021.

[7] Joey Hong, Benjamin Sapp, and James Philbin. Rules of the road: Predicting driving behavior with a convolutional model of semantic interactions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8454–8462, 2019.

[8] Felipe Codevilla, Matthias Müller, Antonio López, Vladlen Koltun, and Alexey Dosovitskiy. End-to-end driving via conditional imitation learning. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4693–4700. IEEE, 2018.

[9] Henggang Cui, Vladan Radosavljevic, Fang-Chieh Chou, Tsung-Han Lin, Thi Nguyen, Tzu-Kuo Huang, Jeff Schneider, and Nemanja Djuric. Multimodal trajectory predictions for autonomous driving using deep convolutional networks. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 2090–2096. IEEE, 2019.

[10] Haibao Yu, Wenxian Yang, Hongzhi Ruan, Zhenwei Yang, Yingjuan Tang, Xu Gao, Xin Hao, Yifeng Shi, Yifeng Pan, Ning Sun, et al. V2x-seq: A large-scale sequential dataset for vehicle-infrastructure cooperative perception and forecasting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5486–5495, 2023.

[11] Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

[12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

[13] S Hochreiter. Long short-term memory. *Neural Computation MIT-Press*, 1997.

[14] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

[15] Bernhard Jaeger, Kashyap Chitta, and Andreas Geiger. Hidden biases of end-to-end driving models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8240–8249, 2023.

[16] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[17] Hao Shao, Yuxuan Hu, Letian Wang, Guanglu Song, Steven L Waslander, Yu Liu, and Hongsheng Li. Lmdrive: Closed-loop end-to-end driving with large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15120–15130, 2024.

[18] Licheng Wen, Daocheng Fu, Xin Li, Xinyu Cai, Tao Ma, Pinlong Cai, Min Dou, Botian Shi, Liang He, and Yu Qiao. Dilu: A knowledge-driven approach to autonomous driving with large language models. *arXiv preprint arXiv:2309.16292*, 2023.

[19] Aditya Prakash, Kashyap Chitta, and Andreas Geiger. Multi-modal fusion transformer for end-to-end autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7077–7087, 2021.

[20] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving.

In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020.

[21] Hao Shao, Letian Wang, Ruobing Chen, Hongsheng Li, and Yu Liu. Safety-enhanced autonomous driving using interpretable sensor fusion transformer. In *Conference on Robot Learning*, pages 726–737. PMLR, 2023.

[22] Dan Li, Xin Zhang, Xiaofan Liu, Zhicheng Ma, and Baolong Zhang. Driver fatigue detection based on comprehensive facial features and gated recurrent unit. *Journal of Real-Time Image Processing*, 20(2):19, 2023.

[23] Bogusław Cyganek and Sławomir Gruszczyński. Hybrid computer vision system for drivers' eye recognition and fatigue monitoring. *Neurocomputing*, 126:78–94, 2014.

[24] Dan Witzner Hansen and Qiang Ji. In the eye of the beholder: A survey of models for eyes and gaze. *IEEE transactions on pattern analysis and machine intelligence*, 32(3):478–500, 2009.

[25] Fangming Qu, Nolan Dang, Borko Furht, and Mehrdad Nojoumian. Comprehensive study of driver behavior monitoring systems using computer vision and machine learning techniques. *Journal of Big Data*, 11(1):32, 2024.

[26] Scott Ettinger, Shuyang Cheng, Benjamin Caine, Chenxi Liu, Hang Zhao, Sabeek Pradhan, Yuning Chai, Ben Sapp, Charles R Qi, Yin Zhou, et al. Large scale interactive motion forecasting for autonomous driving: The waymo open motion dataset. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9710–9719, 2021.

[27] Raghuraman Gopalan, Tsai Hong, Michael Shneier, and Rama Chellappa. A learning approach towards detection and tracking of lane markings. *IEEE Transactions on Intelligent Transportation Systems*, 13(3):1088–1098, 2012.

[28] Khan Muhammad, Tanveer Hussain, Hayat Ullah, Javier Del Ser, Mahdi Rezaei, Neeraj Kumar, Mohammad Hijji, Paolo Bellavista, and Victor Hugo C de Albuquerque. Vision-based semantic segmentation in scene understanding for autonomous driving: Recent achievements, challenges, and outlooks. *IEEE Transactions on Intelligent Transportation Systems*, 23(12):22694–22715, 2022.

[29] Raman Kumar and Anuj Jain. Driving behavior analysis and classification by vehicle obd data using machine learning. *The Journal of Supercomputing*, 79(16):18800–18819, 2023.

[30] Supriya Sarker and Md Mokammel Haque. An approach towards domain knowledge-based classification of driving maneuvers with lstm network. In *Proceedings of International Joint Conference on Advances in Computational Intelligence: IJCACI 2020*, pages 469–484. Springer, 2021.

[31] Sergio Casas, Cole Gulino, Renjie Liao, and Raquel Urtasun. Spagnn: Spatially-aware graph neural networks for relational behavior forecasting from sensor data. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9491–9497. IEEE, 2020.

[32] Jair Ferreira, Jnior, Eduardo Carvalho, Bruno V. Ferreira, Cleidson de Souza, Yoshihiko Suhara, Alex Pentland, and Gustavo Pessin. Driver behavior profiling: An investigation with different smartphone sensors and machine learning. *PLOS ONE*, 12(4):1–16, 04 2017.

[33] Junwei You, Ying Chen, Zhuoyu Jiang, Zhangchi Liu, Zilin Huang, Yifeng Ding, and Bin Ran. Exploring driving behavior for autonomous vehicles based on gramian angular field vision transformer. *IEEE Transactions on Intelligent Transportation Systems*, 2024.

[34] Eleni Mantouka, Emmanouil Barmpounakis, Eleni Vlahogianni, and John Golias. Smartphone sensing for understanding driving behavior: Current practice and challenges. *International journal of transportation science and technology*, 10(3):266–282, 2021.

[35] Wenzhuo Liu, Jianli Lu, Junbin Liao, Yicheng Qiao, Guoying Zhang, Jiayin Zhu, Bozhang Xu, and Zhiwei Li. Fmdnet: Feature-attention-embedding-based multimodal-fusion driving-behavior-classification network. *IEEE Transactions on Computational Social Systems*, 2024.

[36] Kashyap Chitta, Aditya Prakash, Bernhard Jaeger, Zehao Yu, Katrin Renz, and Andreas Geiger. Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(11):12878–12895, 2022.

[37] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[38] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113, 2023.

[39] Yinhan Liu. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 364, 2019.

[40] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. *CoRR*, abs/1511.07122, 2015.

[41] Xiangwang Hu, Zuduo Zheng, Danjue Chen, Xi Zhang, and Jian Sun. Processing, assessing, and enhancing the waymo autonomous vehicle open dataset for driving behavior research. *Transportation Research Part C: Emerging Technologies*, 134:103490, 2022.

[42] Alex Graves and Alex Graves. Long short-term memory. *Supervised sequence labelling with recurrent neural networks*, pages 37–45, 2012.

[43] Zhiguang Wang, Weizhong Yan, and Tim Oates. Time series classification from scratch with deep neural networks: A strong baseline. In *2017 International joint conference on neural networks (IJCNN)*, pages 1578–1585. IEEE, 2017.

[44] Fazle Karim, Somshubra Majumdar, Houshang Darabi, and Shun Chen. Lstm fully convolutional networks for time series classification. *IEEE access*, 6:1662–1669, 2017.

[45] Nelly Elsayed, Anthony S Maida, and Magdy Bayoumi. Deep gated recurrent and convolutional network hybrid model for univariate time series classification. *arXiv preprint arXiv:1812.07683*, 2018.

[46] Jingyuan Wang, Ze Wang, Jianfeng Li, and Junjie Wu. Multilevel wavelet decomposition network for interpretable time series analysis. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2437–2446, 2018.

[47] Fazle Karim, Somshubra Majumdar, Houshang Darabi, and Samuel Harford. Multivariate lstm-fcns for time series classification. *Neural networks*, 116:237–245, 2019.

[48] George Zerveas, Srideepika Jayaraman, Dhaval Patel, Anuradha Bhamidipaty, and Carsten Eickhoff. A transformer-based framework for multivariate time series representation learning. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pages 2114–2124, 2021.