# Driver Maneuver Interaction Identification with Anomaly-Aware Federated Learning on Heterogeneous Feature Representations

MAHAN TABATABAIE, University of Connecticut, USA
SUINING HE, University of Connecticut, USA

Driver maneuver interaction learning (DMIL) refers to the classification task with the goal of identifying different driver-vehicle maneuver interactions (e.g., left/right turns). Existing conventional studies largely focused on the centralized collection of sensor data from the drivers' smartphones (say, inertial measurement units or IMUs, like accelerometer and gyroscope). Such a centralized mechanism might be precluded by data regulatory constraints. Furthermore, how to enable an adaptive and accurate DMIL framework remains challenging due to (i) complexity in heterogeneous driver maneuver patterns, and (ii) impacts of anomalous driver maneuvers due to, for instance, aggressive driving styles and behaviors.

To overcome the above challenges, we propose AF-DMIL, an **A**nomaly-aware **F**ederated **D**river **M**aneuver **I**nteraction **L**earning system. We focus on the real-world IMU sensor datasets (e.g., collected by smartphones) for our pilot case study. In particular, we have designed three heterogeneous representations for AF-DMIL regarding spectral, time series, and statistical features that are derived from the IMU sensor readings. We have designed a novel heterogeneous representation attention network (HetRANet) based on spectral channel attention, temporal sequence attention, and statistical feature learning mechanisms, jointly capturing and identifying the complex patterns within driver maneuver behaviors. Furthermore, we have designed a densely-connected convolutional neural network in HetRANet to enable the complex feature extraction and enhance the computational efficiency of HetRANet. In addition, we have designed within AF-DMIL a novel anomaly-aware federated learning approach for decentralized DMIL in response to anomalous maneuver data. To ease extraction of the maneuver patterns and evaluation of their mutual differences, we have designed an embedding projection network that projects the high-dimensional driver maneuver features into low-dimensional space, and further derives the exemplars that represent the driver maneuver patterns for mutual comparison. Then, AF-DMIL further leverages the mutual differences of the exemplars to identify those that exhibit anomalous patterns and deviate from others, and mitigates their impacts upon the federated DMIL. We have conducted extensive driver data analytics and experimental studies on three real-world datasets (one is harvested on our own) to evaluate the prototype of AF-DMIL, demonstrating AF-DMIL's accuracy and effectiveness compared to the state-of-the-art DMIL baselines (on average by more than 13% improvement in terms of DMIL accuracy), as well as fewer communication rounds (on average 29.20% fewer than existing distributed learning mechanisms).

CCS Concepts: • **Human-centered computing** → *Ubiquitous and mobile computing*.

Additional Key Words and Phrases: Driver maneuver interaction learning, anomaly-aware federated learning, heterogeneous maneuver representation learning.

**ACM Reference Format:**
Mahan Tabatabaie and Suining He. 2023. Driver Maneuver Interaction Identification with Anomaly-Aware Federated Learning on Heterogeneous Feature Representations . *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 7, 4, Article 180 (December 2023), 28 pages. https://doi.org/10.1145/3631421

Authors' addresses: Mahan Tabatabaie, School of Computing, College of Engineering, University of Connecticut, Storrs, CT, USA, mahan.tabatabaie@uconn.edu; Suining He, School of Computing, College of Engineering, University of Connecticut, Storrs, CT, USA, suining.he@uconn.edu.


2474-9567/2023/12-ART180 $15.00
https://doi.org/10.1145/3631421

## 1 INTRODUCTION

With the pervasive availability of various sensing techniques, driver behavior analytics have become a key enabler for myriads of ubiquitous computing and mobile applications [17, 30, 34, 51, 61]. Analyzing driver's behaviors can further reduce the inattentive maneuvers and human driving errors [46], and provide timely and potentially useful feedback regarding their driving behaviors (e.g., through the advanced driver assistance system or ADAS). Furthermore, with the ever-growing prevalence of connected autonomous vehicles (CAVs) [51], the insights from driver behavior analytics can be leveraged for autonomous driving model training [32, 52, 57] toward more scalable CAV system development.

To enable ubiquitous driver data analytics, we focus in this work on the driver maneuver interaction learning (DMIL), i.e., identifying various types of driver maneuver behaviors (for instance, left/right turns) given measurements from sensors available on the drivers' smartphones or installed inside the vehicles (illustrated in Fig. 1). Among various sensors explored in the prior studies [8, 15, 72], the mobile/vehicle-equipped inertial measurement units (IMUs) can provide fine-grained maneuver behavior insights, and enable the ubiquitous DMIL applications [68], especially under GPS-less or low-light urban environments [65], where GPS- or camera-based techniques may not perform satisfactorily. Therefore, in this prototype DMIL study, we particularly take into account the IMUs, particularly the accelerometer and gyroscope in our experimental designs, as the data sources of our case study.

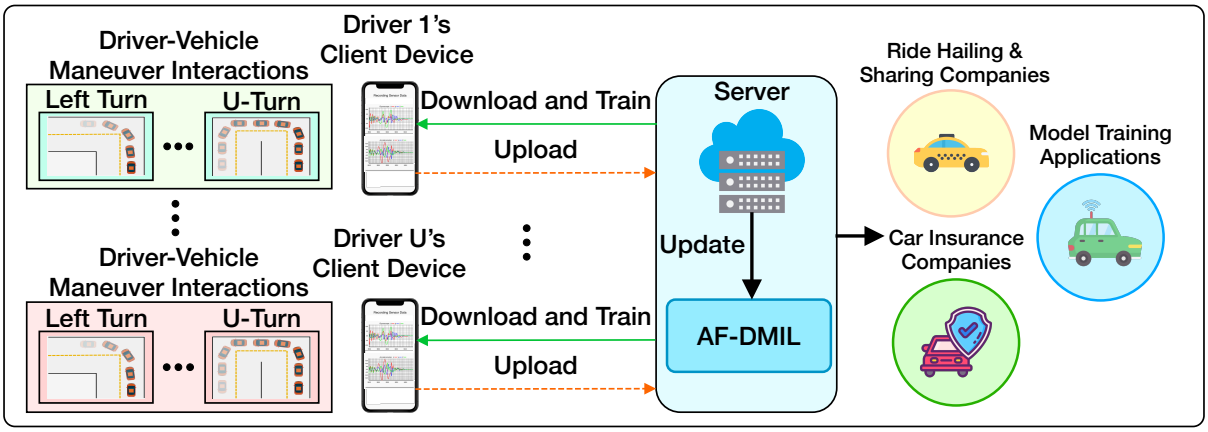


Fig. 1. Design motivations and enabled applications of a federated DMIL framework.

Existing DMIL studies [36, 49, 63] largely consider the centrally-collected data harvested from the drivers' smartphones, such as the IMU sensor time series. These centrally-collected sensor data samples might contain sensitive information with potential data privacy risks, including fingerprinting drivers through the IMU time series [43]. Thanks to the advances in the computation and storage capability of the mobile devices, a distributed learning framework based on federated learning (FL) [41, 56] emerges as a potential workaround. Our studies here aim to design a federated DMIL system in a decentralized manner using the model weight parameters that are based on the local model training on multiple drivers' mobile devices (client devices or smartphones). A global model will be iteratively updated through aggregation of the new local model weight parameters by a communication round between the client devices and the server. Then, the client device leverages the global model and further fine-tunes it, say, based on their locally-collected and stored sensor datasets, into a local model. This way, a federated DMIL system can be enabled to incorporate the driver maneuver data from a large number of participating drivers (e.g., applications of vehicle crowdsensing or ride-sharing fleet [28]). With

privacy-preserving implications, such as the locally-stored sensor data does not leave the client device, we can train the DMIL model in a decentralized manner.

Toward such a federated DMIL system, this study focuses on addressing the following two major research challenges:

**Challenge A — How to effectively capture the heterogeneous feature representations of driver maneuvers:** Existing DMIL approaches often rely on the input representation of driver maneuvers, i.e., how the maneuver patterns are structured and formatted for DMIL model learning. The complex spatial mobility characteristics (e.g., how a turn is affected by the road curvatures and conditions), temporal signal dynamics (i.e., the order of the driver's actions during a maneuver interaction), and measurement noise (e.g., collected from the mobile IMU sensors) often make it highly challenging to capture the driver maneuver patterns that are essential for learning and identification. Conventional statistical threshold- or rule-based approaches [47, 48] often focus on feature *representations* that may be engineered, and might not effectively characterize and differentiate the noisy maneuver data, leading to unsatisfactory DMIL results. For instance, we show in Fig. 2 the confusion matrix of a conventional sequence learning model based on long short-term memory (LSTM) for DMIL. We have observed that the similar time series features make it difficult for the approach to differentiate the acceleration and brake maneuvers, yielding low accuracy as illustrated in Fig. 2. Therefore, there exist imperative needs for a *comprehensive representation learning approach* to extract and model the complex, dynamic, and heterogeneous driver maneuver patterns.

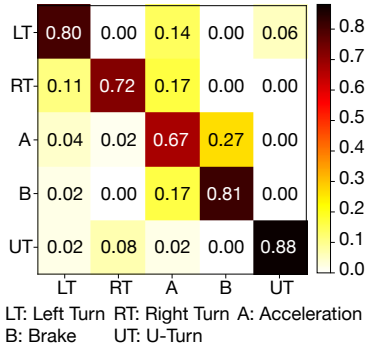


Fig. 2. The confusion matrix of a conventional sequence learning approach for DMIL.

**Challenge B — How to handle anomalous driver maneuvers and interactions for federated DMIL:** We have observed from our real-world driver data analysis that the federated DMIL model training may be prone to anomalous data (e.g., aggressive driving behaviors or styles; maneuvers that are mislabeled as different categories or classes) within the driver maneuver datasets. However, in a federated learning (FL) setting, the server that performs the federated DMIL (say, the DMIL server) cannot access the local driver maneuver records for further identification and validation. Our extensive driver data analytics also revealed that such data anomalies, if left unattended, may lead to DMIL performance degradation and slow model convergence. For instance, we illustrate an example in Fig. 3 a federated DMIL scenario (e.g., by taking the average of the harvested model weight parameters [41]) with different percentages of anomalous driver maneuver data, such as aggressive left/right turns. We can observe that the accuracy of the deep learning approach for DMIL drops with respect to the increased percentage of anomalous maneuvers, with more federated learning communication rounds needed. Such a practical challenge calls for an anomaly-aware federated DMIL design that can adaptively identify anomalous driver maneuver behaviors and subsequently enhance the federated DMIL performance.

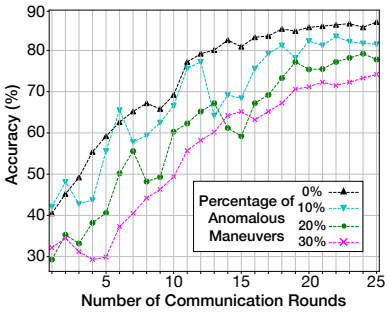


Fig. 3. Performance of conventional federated DMIL given different percentage levels of anomalous driver maneuver data.

To overcome the above-mentioned two major technical barriers, we propose AF-DMIL, an **A**nomaly-aware **F**ederated **D**river **M**aneuver **I**nteraction **L**earning system. Toward prototyping this DMIL system with the IMU data, we have made the following three technical contributions:

(i) **Heterogeneous representation attention network for complex DMIL:** In order to cope with the complexities and dynamics within the driver maneuver data (Challenge A), we have designed a novel DMIL network named `HetRANet`, i.e., the **Het**erogeneous **R**epresentation **A**ttention **Net**work. Specifically, we have conducted extensive real-world IMU sensor data analytics, and taken into account three heterogeneous representations from the accelerometer and gyroscope, i.e., spectral, time series, and statistical features. Such heterogeneous representations aim at comprehensively characterizing the complex driver maneuver behaviors and mitigating the ambiguity across different maneuver classes. To further capture the complex spectral representations derived from the continuous wavelet transform (CWT), we have designed a spectral attention module within `HetRANet`, along with densely-connected convolutional neural networks. The densely-connected convolutional neural networks within `HetRANet` consist of dense blocks and transition blocks, which respectively enforce the complex feature extraction and enhance the computational efficiency of `HetRANet`. Via the spectral attention, temporal sequence attention, and statistical learning modules, our `HetRANet` differentiates the heterogeneous representations, and yields high DMIL accuracy in our experimental studies.

(ii) **Anomaly-aware federated learning for decentralized DMIL model training:** To overcome the challenges on ubiquitous anomaly-awareness (Challenge B), we have designed a novel anomaly-aware federated learning mechanism within `AF-DMIL`. Such a mechanism adapts our core DMIL model and mitigates the impacts of anomalous driver maneuver records. Specifically, to enable the anomaly awareness in `AF-DMIL`, we project the high-dimensional embeddings of each driver's maneuver data into a low-dimensional space, through an embedding projection network that is adapted from `HetRANet`. `AF-DMIL` derives the exemplars that represent the maneuver patterns of the drivers, measures the mutual differences of the exemplars across the client devices, and determines each driver's participation weights in contributing to the federated DMIL. With these weights, `AF-DMIL` adaptively finds the relative importance of different driver maneuver embeddings, and mitigates the impacts of anomalous inputs upon the DMIL model training. Given the increasing pervasiveness of the DMIL-enabled applications, our proposed designs can be further integrated with the ubiquitous vehicle crowdsensing [29] and crowdsourcing settings for large-scale driver behavior analysis applications [60] (e.g., driving safety [33], driver authentication [27], and drivers' in-vehicle well-being [4]).

(iii) **Extensive data analytics and data-driven model studies and experimental evaluations:** In order to validate our proposed design, we have performed extensive driver maneuver data analytics to derive the representation learning designs of `AF-DMIL`. We have studied two open-source datasets and one collected on our own, which consists of a total of 6,698 driver maneuver records. We have designed various experimental settings in order to evaluate `AF-DMIL`'s accuracy in identifying complex driver maneuvers and effectiveness in handling anomalous maneuvers for federated DMIL. Our extensive experimental results have demonstrated that our `AF-DMIL` achieves higher accuracy (on average 13.71% improvements in DMIL accuracy) compared with the other baseline approaches, such as [1, 18, 42, 63, 77], and faster convergence (on average 29.20% fewer communication rounds) than the existing distributed learning paradigms including [10, 25, 37, 39, 41].

The rest of the paper is organized as follows. We first review the related studies in Sec. 2. Then, we present the driver maneuver datasets studied, overview the system framework, and discuss the data preprocessing and our problem statement in Sec. 3. After that, we introduce the proposed heterogeneous representation learning for DMIL in Sec. 4, and provide the anomaly-aware federated DMIL designs in Sec. 5. We present the experimental studies in Sec. 6, discuss the deployment of the proposed system in Sec. 7, and finally conclude the paper in Sec. 8.

## 2 RELATED WORKS

We overview our related work in the following two categories.

• **Driver Maneuver Interaction Learning (DMIL):** With the advances in driver behavioral analysis and ubiquitous vehicle computing, the DMIL problem has attracted much attention recently. Conventional DMIL approaches often considered rule-based [8, 9] or traditional machine learning algorithms to identify the driver maneuvers and interaction behaviors. For instance, Chen et al. [8] proposed a data-driven threshold design based on the gyroscope readings to identify the maneuvers and lane changes [9]. Traditional machine learning approaches, including decision trees [69], hidden Markov model [78], and random forest [7], have also been applied to identify the driver maneuvers.

In order to capture more important features for classifying the driver maneuvers, recent studies have taken into account the deep learning approaches for driver maneuver interaction learning. Li et al. [36] and Taherifard et al. [63] studied the long short-term memory (LSTM), and Aboah et al. [2] considered 1-D convolutional neural networks (CNN) to extract the temporal features and sequential dependencies [49] within the driver maneuver data. Choi et al. [13] studied the recurrent neural network (RNN) to classify the normal and aggressive driving behaviors. Xie et al. [71] and Bejani et al. [5] considered the CNN model to learn and capture the statistical features within the different maneuvers. Nedorubova et al. [44] considered the spectrogram of the sensor time series [62] for human activity recognition. These conventional learning approaches largely considered feature engineering, such as forming feature vectors, without accounting for fusing and learning the heterogeneous representations for DMIL. Hence these approaches may not necessarily capture the complex driver maneuvers for accurate identification. Different from the above studies, our proposed AF-DMIL provides a novel heterogeneous representation attention network by adaptively fusing spectral, time series, and temporal features. Our novel network designs differentiate the different maneuver patterns, and therefore yield higher accuracy and effectiveness on DMIL than the existing baseline approaches [5, 7, 35, 41, 63] in our experimental studies.

• **Federated Learning for Behavior Recognition:** Federated learning (FL) enables the mobile sensing devices to collaboratively learn a shared model, while keeping all the training data locally on the clients or mobile devices [35, 59, 66, 81]. Such a learning paradigm decouples the model training from the need to store the data in the centralized server (say, cloud), which is particularly suitable for ubiquitous human activity recognition with privacy-preserving implications [41]. Existing FL approaches [35, 75] for human activity recognition largely consider the distributed learning processes with normal behavior or activity records that follow the consistent behavior data distributions, while how to cope with the anomalous ones under FL settings remains challenging. Li et al. [37] and Gu et al. [25] studied the reconstruction errors from the auto-encoder models (globally pre-trained with the normal data on the server side) to identify the clients that may be contributing anomalous data. Liu et al. [39] synthesized the human behavior data distributions based on the trained models to compare and identify the abnormal behavior data. In order to mitigate the influence of anomalous data, existing studies have taken into account the model weight parameters [80] or gradients [12] for differentiation. However, the model weight parameters and gradients of the behavior recognition model (e.g., the DMIL models in our case) may mostly capture the majority of normal behavior patterns (e.g., normal left or right turns) and aggregate them within the model weight parameters or the gradients. Therefore, their designs may not well discern the anomalous behaviors such as aggressive left or right turns that might preserve the similarity with the corresponding normal ones, making it difficult to perform the robust federated DMIL.

Different from the above studies, our AF-DMIL provides a new anomaly-aware approach in a federated DMIL setting by (a) providing an embedding projection network that projects the feature embeddings into low-dimensional feature space for efficient neighbor calculation; and (b) detecting relative importance of different drivers' maneuver data based on the neighbors of the feature embeddings. Based on the derived participation weights of the drivers, AF-DMIL provides feature-level differentiation of the normal and anomalous maneuver

data different than the above-mentioned approaches. This way, AF-DMIL enhances the accuracy, adaptivity, and robustness of the federated DMIL as validated in our experimental studies.

## 3 DMIL DATASETS, SYSTEM OVERVIEW & FEATURE EXTRACTION

We first present the datasets studied in Sec. 3.1, followed by the system framework of AF-DMIL in Sec. 3.2. Then, we present the driver maneuver data preprocessing designs, the heterogeneous feature extraction, and the problem statement of DMIL in Sec. 3.3.

### 3.1 Overview of DMIL Datasets Studied

We present the details of the three DMIL datasets as follows.

• *DMIL Dataset 1 (DS1):* We leverage an open-source driver maneuver dataset (i.e., accelerometer and gyroscope measurements) [74] harvested by an Apple iPhone 5 with a sampling rate of 50 Hz. The dataset consists of driver maneuvers from multiple metropolitan cities in the U.S. In this prototype study, we select and label maneuvers collected in Manhattan and Brooklyn in New York City as well as San Francisco and San Jose in California. In addition to the normal driver maneuver patterns such as normal acceleration and braking, we have further taken into account the aggressive acceleration and aggressive braking identified from DS1 for evaluation of our anomaly-aware federated DMIL.

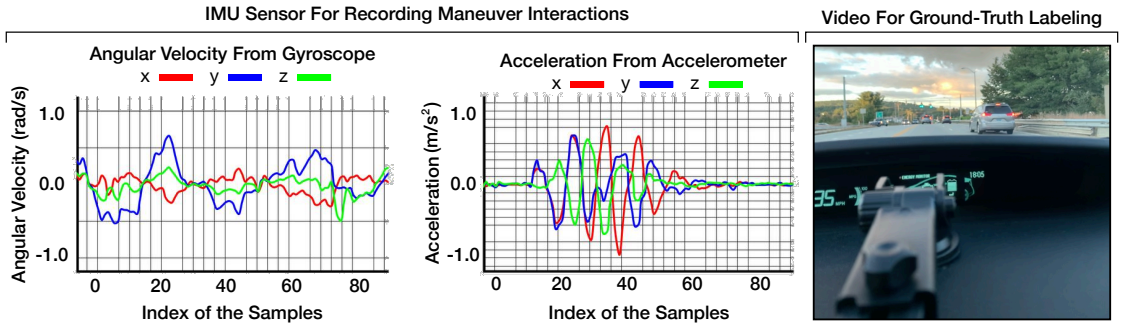


Fig. 4. Illustration of our smartphone-based prototype setup and data collection setting for DS2. During the data collection, we have enabled the main camera of the smartphone to record the maneuver behaviors for the ground-truth driver maneuver class labeling.

• *DMIL Dataset 2 (DS2):* We have collected our driver maneuver dataset when driving a 2010 Toyota Prius in our university town (situated in a rural area) in North America using an Apple iPhone XR and a Google Pixel 3. We set the sampling rate of the IMU sensors (accelerometer and gyroscope) of the Apple iPhone XR to 40Hz and the Google Pixel 3 to 240Hz. The smartphones are mounted near the dashboard of the vehicle during data collection, and we have the main camera facing the forward (front-view) direction during the driving, and the recorded videos are used for ground-truth maneuver class labeling. We leverage such a mobile platform for DS2 collection to enable system prototype development studies.

We illustrate our prototype setup and data collection settings in Fig. 4. This prototype study includes the videos recorded by the smartphone's camera, and we label the ground-truth driver maneuver class (denoted as $\mathbf{y}$) according to the scenes within the recorded videos. We note that our entire driver maneuver data collection for DS2 was performed in a naturalistic driving setting [61] with the minimum interference on the drivers and the local traffic, i.e., collected during daily commutes. Our settings follow the common driving safety measures, the local traffic rules, and the related social norms.

• *DMIL Dataset 3 (DS3):* In order to emulate the DMIL scenarios with anomalous maneuver patterns, we have further taken into account an open-source dataset from [21], which was collected with a Motorola XT1058 (with a sampling rate of 50–200 Hz) when a driver is driving and maneuvering the 2011 Honda Civic. DS3 contains the aggressive driving behaviors, such as aggressive left/right turns, acceleration, and braking, which are further leveraged for the evaluation of our anomaly-aware federated DMIL designs.

Toward the prototyping of AF-DMIL, we focus on the following driver maneuver classes, i.e., left turn (LT), aggressive left turn (ALT), right turn (RT), aggressive right turn (ART), normal acceleration (NA), aggressive acceleration (AA), normal brake (NB), and aggressive brake (AB). Table 1 summarizes the three DMIL datasets for our prototype studies. For DS1, we have labeled a total of 5,736 maneuver records collected by 100 drivers (50 from New York State and 50 from California State) from DS1, which consists of a total of 7 different maneuver classes. For DS2, we have identified a total of 824 maneuvers (5 maneuver classes). DS3 consists of 4 different maneuver classes with driving behaviors that are labeled as aggressive ones (e.g., aggressive acceleration and aggressive braking).

Table 1. Statistics of the three datasets studied for DMIL.

| Datasets | Maneuver Classes Involved | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | LT | ALT | RT | ART | UT | NA | AA | NB | AB |
| **DS1** | 729 | N/A | 454 | N/A | 778 | 2,175 | 148 | 1,298 | 154 |
| **DS2** | 128 | N/A | 64 | N/A | 178 | 249 | N/A | 205 | N/A |
| **DS3** | N/A | 33 | N/A | 33 | N/A | N/A | 36 | N/A | 36 |

In evaluating the heterogeneous representation learning designs, we focus on the five conventional maneuver classes, i.e., LT, RT, UT, NA, and NB. To emulate the federated DMIL with the anomalous maneuver patterns, we infuse the aggressive maneuvers — that is, ALT, ART, AA, and AB — within the drivers' maneuvering data of DS1 and DS3 with respect to each of the corresponding classes of normal maneuvers — that is, LT, RT, NA, and NB. For instance, we emulate the scenarios (e.g., due to mislabeling) when the aggressive left turns (ALTs) are introduced to the maneuver class of left turns (LTs) to form a mixture of anomalous and normal maneuvers, and evaluate the performance of our federated DMIL and other approaches.

## 3.2 Overview of System Framework

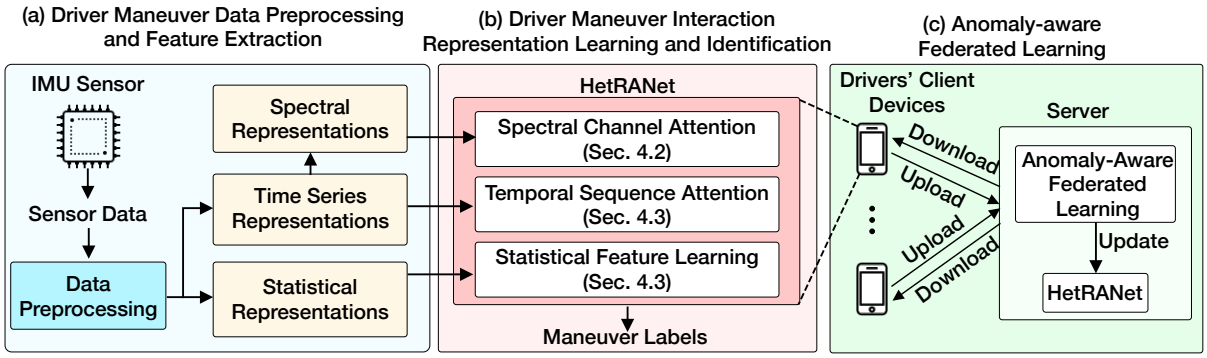


Fig. 5. Overview of our proposed AF-DMIL's system framework.

We overview the system framework of AF-DMIL in Fig. 5, which consists of the following three phases:

**(a) Driver Maneuver Data Preprocessing and Feature Extraction:** In this phase, AF-DMIL preprocesses the collected IMU sensor data and extracts the heterogeneous features for DMIL. AF-DMIL first performs the

coordinate alignment to align the IMU sensor values (the local coordinate system) with the vehicle's (earth) coordinate system. Afterwards, AF-DMIL segments the time series, filters the resulting segments based on a Kalman filter [70], and normalizes the segments into the range of [-1, 1]. Given the preprocessed data for each driver indexed by $d$, AF-DMIL extracts and generates three heterogeneous representations of the input driver maneuvers, i.e., spectral representations $\mathbf{P}^d$, time series representations $\mathbf{T}^d$, and statistical feature representations $\mathbf{S}^d$.

**(b) Driver Maneuver Interaction Representation Learning and Identification:** In this phase, AF-DMIL further takes in and learns the input representations with our proposed heterogeneous representation attention network (HetRANet), which consists of three parallel modules of spectral channel attention, temporal sequence attention, and statistical feature learning. AF-DMIL combines the outputs of the above three modules to produce the probability scores of different driver maneuver interaction classes.

**(c) Anomaly-Aware Federated Learning:** In this phase, AF-DMIL further trains the local models of HetRANet on different client devices (the drivers' smartphones). Specifically, each client device downloads the global model weight parameters of HetRANet from the server, and trains the DMIL model with the local maneuver dataset at the driver's side. We note that on the client device side, AF-DMIL projects the high-dimensional embedding features to a low-dimensional space through an embedding projection network. Per driver maneuver class, AF-DMIL retrieves the mean of the projected embeddings as the exemplar to represent the driver's maneuver patterns. Each client device uploads the resulting model weight parameters of the local HetRANet along with the sets of exemplars for the maneuver classes studied. The DMIL server then performs the maneuver anomaly detection, and derives the participation weights for the contributing drivers based on their relative importance. The participation weights help differentiate and exclude the anomalous driver maneuvers, and mitigate their impacts upon the global DMIL model training at the DMIL server.

### 3.3 Driver Maneuver Data Preprocessing, Feature Extraction, and DMIL Problem Statement

• **Driver Maneuver Data Preprocessing:** In order to perform the DMIL studies, we align the IMU sensor data collected by the smartphone, i.e., acceleration (in m/s$^2$) and angular velocity (in rad/s), to the vehicle's coordinate system. This way, we can reflect the vehicle dynamics regardless of how the smartphone is mounted. Specifically, we leverage the rotation matrix provided by Android or iOS APIs [3, 23], which fuses the accelerometer, gyroscope, and magnetometer, and transform the IMU readings from the smartphone's coordinate system ($x'$, $y'$, and $z'$ in Fig. 6) to the ones of the vehicle's coordinate system ($x$, $y$, and $z$ in Fig. 6). We denote the aligned time series of acceleration with respect to the three axes as $\mathbf{a}_x$, $\mathbf{a}_y$, and $\mathbf{a}_z$, and we similarly have the aligned angular velocity series as $\mathbf{g}_x$, $\mathbf{g}_y$, and $\mathbf{g}_z$.

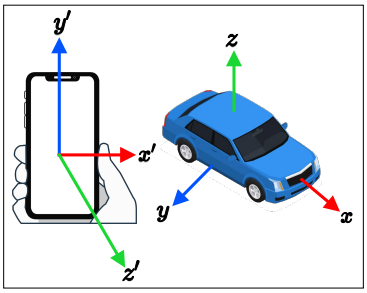

Fig. 6. Illustration of the coordinate systems of the vehicle (left) and the smartphone (right) before the coordinate alignment.

We further segment the aligned IMU sensor time series with a sliding window into segments of length $\omega$ and 50% overlap (further evaluation can be referred to Sec. 6). AF-DMIL feeds each time series segment through a Kalman filter [70] to mitigate the noise due to the inherent imperfection of smartphone IMU sensors. We normalize each time series segment with respect to each axis based on the min-max normalization into the range of $[-1, 1]$. We then pad each time series with zeros to ensure that all the time series segments have an equal length of $\omega$. We note that as the IMUs are inherently noisy and the vehicle is non-static during maneuvering, it is very unlikely to observe a long sequence of zeros from IMU data, and our DMIL model can still distinguish the padded zero sequence.

• **Driver Maneuver Feature Extraction:** Based on the processed data, in this prototype study, we will derive the following three heterogeneous representations.

(a) Time Series Representations: For each $d$-th driver ($d \in \{1, \ldots, U\}$), we preprocess and obtain a set of $N^d \times 6$ time series segments, denoted as $\mathbf{T}^d = \left\{\mathbf{t}^{d,1}, \ldots, \mathbf{t}^{d,N^d}\right\}$, where each element is given by

$$\mathbf{t}^{d,i} = \left\{\mathbf{a}_x^{d,i}, \mathbf{a}_y^{d,i}, \mathbf{a}_z^{d,i}, \mathbf{g}_x^{d,i}, \mathbf{g}_y^{d,i}, \mathbf{g}_z^{d,i}\right\}, \tag{1}$$

and $\mathbf{a}_x^{d,i}$, $\mathbf{a}_y^{d,i}$, $\mathbf{a}_z^{d,i}$, $\mathbf{g}_x^{d,i}$, $\mathbf{g}_y^{d,i}$, and $\mathbf{g}_z^{d,i} \in \mathbb{R}^{\omega}$ (each is of length $\omega$) respectively correspond to the time series values of accelerometer and gyroscope. Here, the $x$, $y$, and $z$ axes are all at the vehicle's coordinate system.

(b) Spectral Representations: Recall that as illustrated in Fig. 2, using only time series representation may not necessarily differentiate the complex driver maneuvers. We further generate the spectral representations of the driver maneuvers based on the continuous wavelet transform (CWT) [24] of the time series representations $\mathbf{t}^{d,i}$. CWT is based on the classical short-time Fourier transform, while allowing for the variable time-frequency resolutions [50]. In our DMIL studies, we adopt the Ricker wavelet function, i.e., the Mexican hat wavelet function [50], with $V$ different scales of the wavelet function (from 1 to $V = 50$ in our case), and derive the spectral representation.

In particular, for each scale $\gamma \in \{1, \ldots, V\}$ and translational value $b$ ($b = 0$ in our current study), we convolve the Ricker wavelet function at each time step $t$, denoted as $\mathcal{G}(\cdot)$, with each of the six time series segments in $\mathbf{t}^{d,i}$, denoted as $\mathcal{H}(t)$, and obtain the spectral representations $\mathbf{p}^{d,i}$. For each time step of a time series segment, $\mathcal{H}(t)$, we denote the continuous wavelet transformation (CWT) process, which is approximated by the rectangular integration in our implementation, with the scale $\gamma$ by $\texttt{CWT}(\mathcal{H}(t), \gamma)$, i.e.,

$$\texttt{CWT}(\mathcal{H}(t), \gamma) = \frac{1}{\sqrt{\gamma}} \int_{-\infty}^{\infty} \left(\mathcal{H}(t) \times \mathcal{G}\left(\frac{t-b}{\gamma}\right)\right) dt \approx \frac{1}{\sqrt{\gamma}} \sum_{t=1}^{\omega} \left(\mathcal{H}(t) \times \mathcal{G}\left(\frac{t-b}{\gamma}\right)\right). \tag{2}$$

We perform the $\texttt{CWT}(\mathcal{H}(t), \gamma)$ operation upon each time step $t \in \{1, \ldots, \omega\}$ with respect to each scale $\gamma \in \{1, \ldots, V\}$ of the Ricker wavelet function, and then obtain the corresponding 2-D spectral representations for each axis of the accelerometer and gyroscope, i.e., $\tilde{\mathbf{a}}_x^{d,i}$, $\tilde{\mathbf{a}}_y^{d,i}$, $\tilde{\mathbf{a}}_z^{d,i}$, $\tilde{\mathbf{g}}_x^{d,i}$, $\tilde{\mathbf{g}}_y^{d,i}$, and $\tilde{\mathbf{g}}_z^{d,i} \in \mathbb{R}^{V \times \omega}$, for each driver $d$. Based on the above, we then transform the time series representations of a driver $d$, denoted as $\mathbf{T}^d = \{\mathbf{t}^{d,1}, \ldots, \mathbf{t}^{d,N^d}\}$, into the spectral representations of the driver maneuvers, i.e.,

$$\mathbf{P}^d = \left\{\mathbf{p}^{d,1}, \ldots, \mathbf{p}^{d,N^d}\right\}, \quad \text{where} \quad \mathbf{p}^{d,i} = \left[\tilde{\mathbf{a}}_x^{d,i}, \tilde{\mathbf{a}}_y^{d,i}, \tilde{\mathbf{a}}_z^{d,i}, \tilde{\mathbf{g}}_x^{d,i}, \tilde{\mathbf{g}}_y^{d,i}, \tilde{\mathbf{g}}_z^{d,i}\right] \in \mathbb{R}^{V \times \omega \times 6}. \tag{3}$$

Here each element of $\mathbf{p}^{d,i}$ corresponds to the wavelet coefficient (of dimension $V \times \omega$) for each axis of the IMU sensors studied.

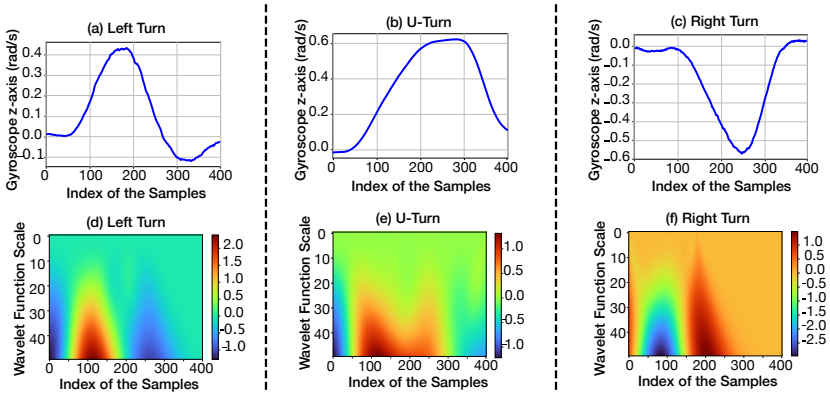


Fig. 7. Time series and spectral representations of three driver maneuver samples: (a) left turn; (b) U-turn; and (c) right turn.

We illustrate three examples of time series (angular velocity with respect to the $z$-axis), left turn, U-turn, and right turn, as well as the corresponding spectral representations in Figs. 7(a), (b), and (c). Specifically, we can observe that the left turn and U-turn exhibit similar shapes in terms of time series representation, as illustrated in Figs. 7(a) and (b). With the derived spectral representations (see Figs. 7(d) and (e)), we may further differentiate the complex driver maneuvers and enable more accurate DMIL.

(c) Statistical Representations: To derive a comprehensive representation of the driver maneuvers, we form the statistical representation, denoted as $\mathbf{S}^d = \{\mathbf{s}^{d,1}, \ldots, \mathbf{s}^{d,N^d}\}$, by extracting statistical representations — that is, minimum, maximum, and average values of each time series segment — from the time series of the driver $d$, denoted as $\mathbf{T}^d$, i.e.,

$$\mathbf{s}^{d,i} = \left[\min\left(\mathbf{a}_x^{d,i}\right), \quad \max\left(\mathbf{a}_x^{d,i}\right), \quad \text{average}\left(\mathbf{a}_x^{d,i}\right), \quad \ldots, \quad \min\left(\mathbf{g}_z^{d,i}\right), \quad \max\left(\mathbf{g}_z^{d,i}\right), \quad \text{average}\left(\mathbf{g}_z^{d,i}\right)\right] \in \mathbb{R}^{\omega \times 18}.$$

• **Problem Statement for DMIL:** The problem formulation of AF-DMIL is to take in the heterogeneous representations in terms of spectral, time series, and statistical features, i.e., $\mathbf{M}^d = \left\{\mathbf{P}^d, \mathbf{T}^d, \mathbf{S}^d\right\}$, and returns the estimated labels of the driver maneuvers or classes for each driver $d$ ($d \in \{1, \ldots, U\}$).

## 4 HETEROGENEOUS REPRESENTATION LEARNING FOR DMIL

We first overview the heterogeneous representation learning architecture of AF-DMIL (Sec. 4.1), followed by the detailed designs of spectral channel attention (Sec. 4.2), as well as temporal sequence attention and statistical feature learning (Sec. 4.3).

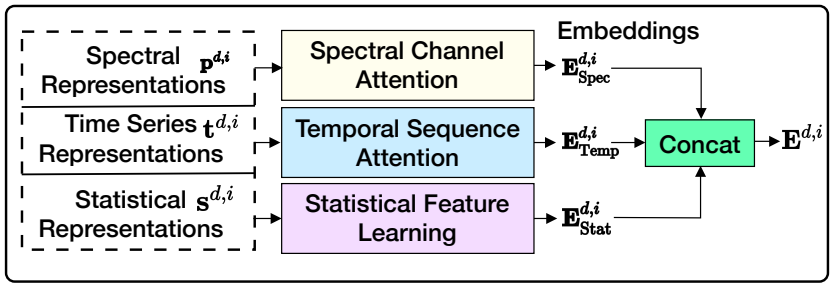


Fig. 8. Overview of the HetRANet network architecture.

### 4.1 Overview of HetRANet for DMIL

We overview the architecture of our proposed heterogeneous representation attention network (HetRANet) in Fig. 8. In particular, given the spectral, time series, and statistical representations for the $i$-th driver maneuver interaction, HetRANet processes them through the following three modules: (a) a spectral channel attention module takes in the spectral representations $\mathbf{p}^{d,i} \in \mathbf{P}^d$ and generates the embedding $\mathbf{E}^{d,i}_{\text{Spec}} \in \mathbb{R}^{q_1}$; (b) a temporal sequence attention module extracts temporal information (such as the order of the driving actions) from the time series representations, $\mathbf{t}^{d,i} \in \mathbf{T}^d$, and generates the embedding $\mathbf{E}^{d,i}_{\text{Temp}} \in \mathbb{R}^{q_2}$; and (c) a statistical feature learning module takes in the statistical representations, $\mathbf{s}^{d,i} \in \mathbf{S}^d$, and generates the embedding $\mathbf{E}^{d,i}_{\text{Stat}} \in \mathbb{R}^{q_3}$. At the final estimation, HetRANet concatenates the resulting three embeddings, denoted as $\mathbf{E}^{d,i}_{\text{Spec}}$, $\mathbf{E}^{d,i}_{\text{Temp}}$, and $\mathbf{E}^{d,i}_{\text{Stat}}$, and further returns the driver maneuver class label.

### 4.2 Spectral Channel Attention Module

• **Design Motivations.** Through the feature extraction in Sec. 3.3, AF-DMIL derives the spectral representations from the multiple axes of the IMU sensors. We note from our data analytics that different axes of the IMU sensors

may contribute differently in identifying different maneuvers. For instance, we have observed that the angular velocity around the $z$-axis of the gyroscope, denoted as $\mathbf{g}_z^{d,i}$, may particularly help differentiate the turning maneuvers (e.g., left turn). Therefore, the key idea of our spectral channel attention is to account for each IMU sensor axis as a channel and leverage a *channel attention* mechanism to assist the DMIL model in distinguishing them. In addition to the channel attention, we have further designed the densely-connected convolution blocks to reuse and fine-grain the features across different layers. This way, AF-DMIL further identifies the most important channels in the spectral representations with respect to different maneuver classes, refines the feature extraction, and yields high DMIL accuracy.

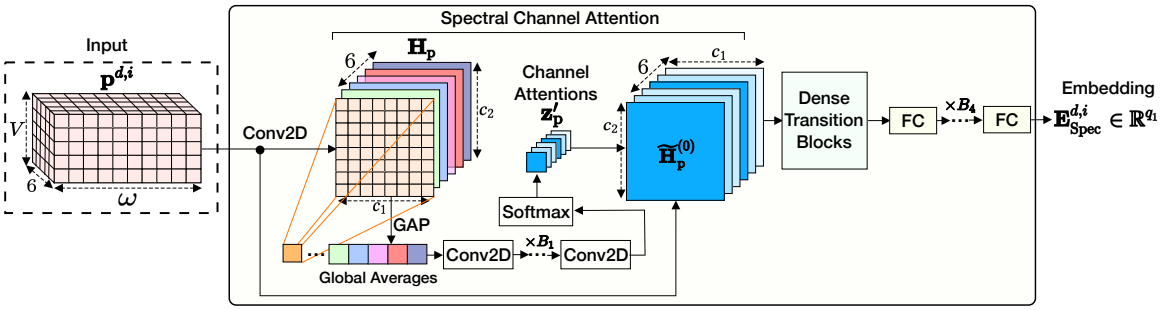


Fig. 9. Illustration of the designs of our proposed spectral channel attention module in HetRANet.

• **Detailed Designs.** We illustrate in Fig. 9 the processing steps of the spectral representations by the spectral channel attention module, which consists of (a) channel attention and (b) dense transition.

(a) Channel Attention: Specifically, given the spectral representations of the $i$-th maneuver of the driver $d$, denoted as $\mathbf{p}^{d,i} \in \mathbb{R}^{V \times \omega \times 6}$, we first process it with a 2-D convolution (Conv2D) layer with 6 filters (equivalent to the number of the sensor axes) to obtain the latent feature maps $\mathbf{H}_\text{p} \in \mathbb{R}^{c_1 \times c_2 \times 6}$, where $c_1$ and $c_2$ represent the width and the height of each latent feature map after the convolution layer, respectively.

We then further have a global average pooling (GAP) [38], a more generalizable alternative to flattening and fully connected layers, to convert the convolutional output feature maps to a vector, which helps mitigate the model over-fitting issues. The global average pooling aggregates the channel-wise information upon the latent feature maps $\mathbf{H}_\text{p}$, and forms the aggregated feature map as $\mathbf{H}'_\text{p} \in \mathbb{R}^{1 \times 1 \times 6}$.

To further differentiate the channel-wise aggregated global information from $\mathbf{H}'_\text{p}$, AF-DMIL integrates a total of $B_1$ consecutive Conv2D layers with a kernel size $1 \times 1$, which extracts the features along the channel dimension to obtain the latent channel features, denoted as $\mathbf{z}_\text{p} \in \mathbb{R}^{1 \times 1 \times 6}$.

Given the above, we find the spectral attention scores, denoted as $\mathbf{z}'_\text{p} \in \mathbb{R}^6$, based on a softmax operation upon $\mathbf{z}_\text{p}$ along its last dimension that corresponds to the channels, i.e.,

$$\mathbf{z}'_\text{p} = \texttt{softmax}\left(\mathbf{z}_\text{p}[j]\right) = \frac{\exp(\mathbf{z}_\text{p}[j])}{\sum_j \exp(\mathbf{z}_\text{p}[j])}, \tag{4}$$

where $j \in \{1, \ldots, 6\}$ represents each of the six IMU sensor axes in our current DMIL designs, i.e., $\left\{\mathbf{a}_x^{d,i}, \mathbf{a}_y^{d,i}, \mathbf{a}_z^{d,i}, \mathbf{g}_x^{d,i}, \mathbf{g}_y^{d,i}, \mathbf{g}_z^{d,i}\right\}$. Afterwards, we impose the channel-wise attention scores $\mathbf{z}'_\text{p}$ upon $\mathbf{H}_\text{p}$ by multiplying them along the channel dimension. We hence obtain the output of our spectral channel attention, denoted as $\widetilde{\mathbf{H}}_\text{p}^{(0)}$, i.e.,

$$\widetilde{\mathbf{H}}_\text{p}^{(0)} = \mathbf{H}_\text{p} \odot \mathbf{z}'_\text{p} \in \mathbb{R}^{c_1 \times c_2 \times 6}, \tag{5}$$

where the operator $\odot$ represents the Hadamard product, i.e., the element-wise multiplication.

(b) Dense Transition: Given the output of the channel attention, we further design the densely-connected convolutional neural networks for AF-DMIL to reuse and refine the features extracted. We present the designs of

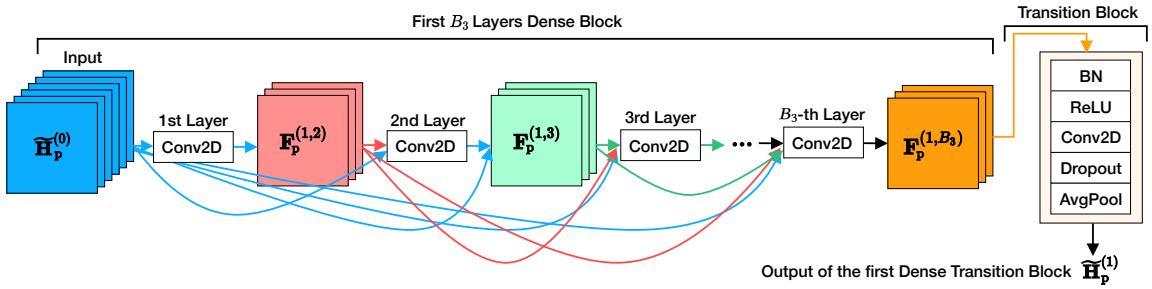


Fig. 10. Details of the first dense transition (DT) block, where BN represents the batch normalization and AvgPool represents the average pooling.

the dense transition (DT) blocks within our densely-connected convolutional neural networks in Fig. 10. Each DT block consists of a dense block and a transition block. The dense block enforces the dense connections across its layers, i.e., the input to each layer is the output of all the preceding layers. Such connections improve the gradient flow beyond the conventional convolutional neural networks (CNNs) by mitigating the gradient vanishing issues [31]. On the other hand, the role of transition blocks is to further reduce the feature dimensions after each dense block, enabling a lightweight and computationally-efficient architecture.

We present the designs of the dense block and the transition block as follows.

— Dense Block: At each dense block $l$ ($l \in \{1, \ldots, B_2\}$), we pass the hidden feature map $\widetilde{\mathbf{H}}_{\mathrm{p}}^{(l-1)}$ through a dense block and obtain the feature map of $\widetilde{\mathbf{H}}_{\mathrm{p}}^{(l)}$. Specifically, each dense block consists of a total of $B_3$ 2-D convolution neural network (Conv2D) layers. For each dense block $l$, we let $\mathbf{F}_{\mathrm{p}}^{(l,o)}$ be the output of the $o$-th Conv2D layer. Conventional convolutional neural networks stack the Conv2D layers and each layer takes in the output only from its preceding layer, which may overlook the important dependencies across these layers and result in the under-utilization of the extracted features. Instead of simple stacking, at each Conv2D layer indexed by $(o+1)$, we concatenate and take in the outputs from all the preceding Conv2D layers, i.e.,

$$\mathbf{F}_{\mathrm{p}}^{(l,o+1)} = \mathtt{Conv2D}\left(\mathtt{ReLU}\left(\mathtt{BN}\left(\mathtt{concat}\left[\mathbf{F}_{\mathrm{p}}^{(l,1)}, \ldots, \mathbf{F}_{\mathrm{p}}^{(l,o)}\right]\right)\right)\right), \tag{6}$$

and we have $\mathbf{F}_{\mathrm{p}}^{(l,1)} = \mathtt{Conv2D}\left(\mathtt{ReLU}\left(\mathtt{BN}\left(\widetilde{\mathbf{H}}_{\mathrm{p}}^{(l-1)}\right)\right)\right)$. Based on Eq. (5), we initially set the input of the first Conv2D layer at the first dense block as $\mathbf{F}_{\mathrm{p}}^{(1,1)} = \widetilde{\mathbf{H}}_{\mathrm{p}}^{(0)}$.

— Transition Block: After the dense block, as illustrated in Fig. 10, we then further process the feature map $\mathbf{F}_{\mathrm{p}}^{(l,B_3)}$ by a transition block. Specifically, we feed the output of the dense block through the batch normalization operation (with the ReLU activation function), and a Conv2D layer, and obtain the feature map $\overline{\mathbf{H}}_{\mathrm{p}}^{(l)}$, i.e.,

$$\overline{\mathbf{H}}_{\mathrm{p}}^{(l)} = \mathtt{Conv2D}\left(\mathtt{ReLU}\left(\mathtt{BN}\left(\mathbf{F}_{\mathrm{p}}^{(l,B_3)}\right)\right)\right). \tag{7}$$

Then, AF-DMIL regularizes the transition block by a dropout layer (Dropout) and compresses the feature map $\overline{\mathbf{H}}_{\mathrm{p}}^{(l)}$ with an average pooling (AvgPool) layer. This way, we obtain the output, denoted as $\widetilde{\mathbf{H}}_{\mathrm{p}}^{(l)}$, of the $l$-th dense transition layer, i.e.,

$$\widetilde{\mathbf{H}}_{\mathrm{p}}^{(l)} = \mathtt{AvgPool}\left(\mathtt{Dropout}\left(\overline{\mathbf{H}}_{\mathrm{p}}^{(l)}\right)\right), \tag{8}$$

which will be further fed to the $(l+1)$-th dense block.

We summarize all the operations for the $l$-th DT layer in Eqs. (6), (7), and (8), as the function $\mathtt{DT}(\cdot)$, i.e.,

$$\widetilde{\mathbf{H}}_{\mathrm{p}}^{(l)} = \mathtt{DT}\left(\widetilde{\mathbf{H}}_{\mathrm{p}}^{(l-1)}\right). \tag{9}$$

To summarize, we stack a total of $B_2$ dense transition (DT) blocks to take in $\widetilde{\mathbf{H}}_{\mathrm{p}}^{(0)}$ and obtain $\widetilde{\mathbf{H}}_{\mathrm{p}}^{(B_2)}$. Then, we flatten the output of the dense transition blocks, denoted as $\widetilde{\mathbf{H}}_{\mathrm{p}}^{(B_2)}$, to reshape it into a vector, and further process it through a total of $B_4$ fully connected (FC) layers (with the ReLU activation function) and obtain $\mathbf{E}_{\mathrm{Spec}}^{d,i} \in \mathbb{R}^{q_1}$, where each of the first ($B_4$-1) FC layers has $c_4$ neurons while the last one has $q_1$ neurons.

### 4.3 Temporal Sequence Attention, Statistical Feature Learning, and Model Training

• **Design Motivations.** In addition to the spectral attention, our AF-DMIL further leverages the temporal sequence attention and statistical feature learning modules to further extract the interdependencies of driver maneuvers with respect to the time series and statistical feature representations. This way, AF-DMIL further learns and captures the heterogeneous representations of the driver maneuvers, and yields accurate DMIL results.

• **Detailed Designs.** We present the detailed designs of the two modules, (a) temporal sequence attention and (b) statistical feature learning, as follows.

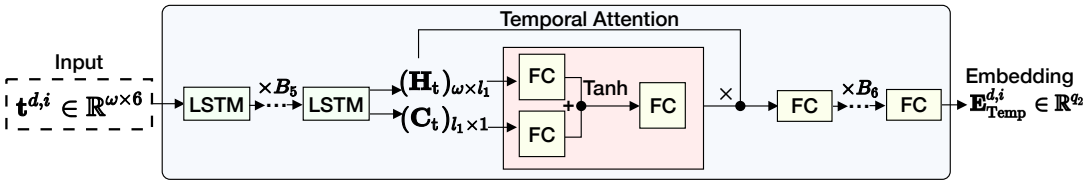


Fig. 11. Network structure of the temporal sequence attention module in HetRANet.

(a) Temporal Sequence Attention Module: We illustrate in Fig. 11 the temporal sequence attention module design for the $i$-th time series representation of the driver $d$, denoted as $\mathbf{t}^{d,i}$. In particular, we first process it with a mask layer to discard the padded zero values, and the output is further processed through a total of $B_5$ long short-term memory (LSTM) layers (with a total of $l_1$ units), and obtain the hidden state $\mathbf{H}_{\mathrm{t}} \in \mathbb{R}^{\omega \times l_1}$ and cell state $\mathbf{C}_{\mathrm{t}} \in \mathbb{R}^{l_1 \times 1}$. Then, we take into account a linear layer with hyperparameters, i.e., $\mathbf{W}_1 \in \mathbb{R}^{l_1 \times l_1}$, $\mathbf{W}_2 \in \mathbb{R}^{l_1 \times l_1}$, and $\mathbf{W}_3 \in \mathbb{R}^{l_1 \times 1}$, to fuse the information within the hidden state and the cell state, and obtain $\mathbf{z}_{\mathrm{t}}$, i.e.,

$$\mathbf{z}_{\mathrm{t}} = \tanh\left(\mathbf{H}_{\mathrm{t}}\mathbf{W}_1 + \mathbf{W}_2\mathbf{C}_{\mathrm{t}}\right) \cdot \mathbf{W}_3. \tag{10}$$

We further find the temporal sequence attention scores, denoted as $\mathbf{z}'_{\mathrm{t}} \in \mathbb{R}^{\omega}$, i.e.,

$$\mathbf{z}'_{\mathrm{t}} = \frac{\exp(\mathbf{z}_{\mathrm{t}}[j])}{\sum_j^{\omega} \exp(\mathbf{z}_{\mathrm{t}}[j])}, \tag{11}$$

where $\omega$ represents the segment length of the hidden state $\mathbf{H}_{\mathrm{t}}$. We further multiply the temporal sequence attention scores $\mathbf{z}'_{\mathrm{t}}$ with the hidden state embeddings $\mathbf{H}_{\mathrm{t}}$ obtained from the LSTM layers for the weighted feature embeddings $\widetilde{\mathbf{H}}_{\mathrm{t}}$, i.e., $\widetilde{\mathbf{H}}_{\mathrm{t}} = \mathbf{H}_{\mathrm{t}} \odot \mathbf{z}'_{\mathrm{t}}$, based on the Hadamard product (element-wise multiplication).

Afterwards, we further process the hidden features $\widetilde{\mathbf{H}}_{\mathrm{t}}$ with a total of $B_6$ FC layers (with the ReLU activation function) followed by a dropout layer, and obtain the final embeddings of the time series representation, denoted as $\mathbf{E}_{\mathrm{Temp}}^{d,i} \in \mathbb{R}^{q_2}$, where each of the first ($B_6$-1) FC layers has a total of $c_5$ neurons while the last one has a total of $q_2$ neurons.

(b) Statistical Feature Learning Module: In particular, we feed the statistical representations of the $i$-th maneuver interaction for the driver $d$, denoted as $\mathbf{s}^{d,i}$, through a total of $B_7$ consecutive FC layers (with the ReLU activation function) followed by a dropout layer. Each of the first ($B_7$-1) FC layers has $c_6$ neurons and the last one has $q_3$ neurons. Then, the statistical feature learning module returns the statistical feature embeddings, denoted as $\mathbf{E}_{\mathrm{Stat}}^{d,i} \in \mathbb{R}^{q_3}$.

• **Model Training Design.** The proposed HetRANet processes the heterogeneous representations, and finally obtains and concatenates the embeddings as

$$\mathbf{E}^{d,i} = \text{concat}\left(\left[\mathbf{E}^{d,i}_{\text{Spec}}, \mathbf{E}^{d,i}_{\text{Temp}}, \mathbf{E}^{d,i}_{\text{Stat}}\right]\right) \in \mathbb{R}^{(q_1+q_2+q_3)}. \tag{12}$$

To identify the driver maneuver class, we feed the embeddings $\mathbf{E}^{d,i}$ through an FC layer, and obtain a vector $\mathbf{v}^{d,i} \in \mathbb{R}^K$ that corresponds to the $K$ driver maneuver classes. We then obtain the probability score through a softmax function, i.e.,

$$\varphi^{d,i}[k] = \frac{\exp\left(\mathbf{v}^{d,i}[k]\right)}{\sum_{k=1}^{K} \exp\left(\mathbf{v}^{d,i}[k]\right)}, \tag{13}$$

where $\mathbf{v}^{d,i} \in \mathbb{R}^K$. Let $\mathbf{y}^{d,i} = \left[y^{d,i}[1], \ldots, y^{d,i}[K]\right]$ be the ground-truth label vector (in one-hot encoding) for the $i$-th maneuver of the driver $d$. We train the model of HetRANet based on cross-entropy loss between the ground-truths $\mathbf{y}^{d,i}$ and the probability scores $\varphi^{d,i}$ (see Appendix).

## 5 ANOMALY-AWARE FEDERATED LEARNING FOR DMIL

We first overview the anomaly-aware federated DMIL designs in Sec. 5.1, and then present the details of the anomaly-aware parameter filtering in Sec. 5.2.

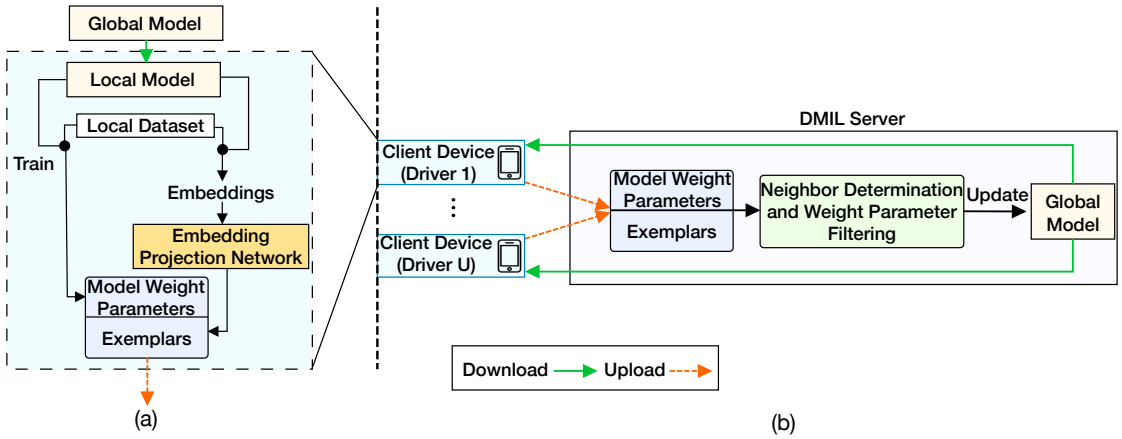


Fig. 12. Overview of our proposed anomaly-aware federated DMIL: (a) the local model training at the driver's client device; and (b) the global model learning at the DMIL server.

### 5.1 Overview of Anomaly-Aware Federated DMIL

We illustrate the overall workflow of our proposed anomaly-aware federated DMIL in Fig. 12, which consists of (a) the local model training and (b) the global model training.

(a) Local Model Training: At each communication round, the DMIL server randomly selects $U$ drivers' client devices for local DMIL model training. Each client device downloads the latest global model weight parameters of HetRANet — that is, aggregated model weight parameters from the previous communication round — from the DMIL server to train the local HetRANet model using its local dataset. In the meantime, the client's device obtains the low-dimensional feature projection of the output embeddings of HetRANet, and derives the exemplars of projected embeddings. Both the exemplars and the model weight parameters are uploaded at each communication round to the DMIL server for global model training.

(b) Global Model Training: After collecting the model weight parameters and exemplars from the client devices, the DMIL server leverages the low-dimensional exemplars to determine the participation weight of each

client device with respect to each driver maneuver class. Then, the DMIL server filters away the model weight parameters of those client devices that are estimated to have low participation weights at the current round. After that, the DMIL server leverages the remaining model weight parameters for the global model training. Here we follow the paradigm of federated averaging (FedAvg) [41] to aggregate the model weight parameters, and the resulting global model of HetRANet is used for future communication rounds of federated DMIL. AF-DMIL performs the multiple updates of the HetRANet model at the client device (i.e., passing multiple rounds of the local dataset) before transmitting the updated local model weight parameters to the DMIL server for the global model update. Multiple steps of local model updates can help overcome the limitations of FedSGD [10] (which performs one step) when the frequent communication of a single update step per client device may not be feasible under dynamic DMIL.

## 5.2 Anomaly-Aware Parameter Filtering Designs

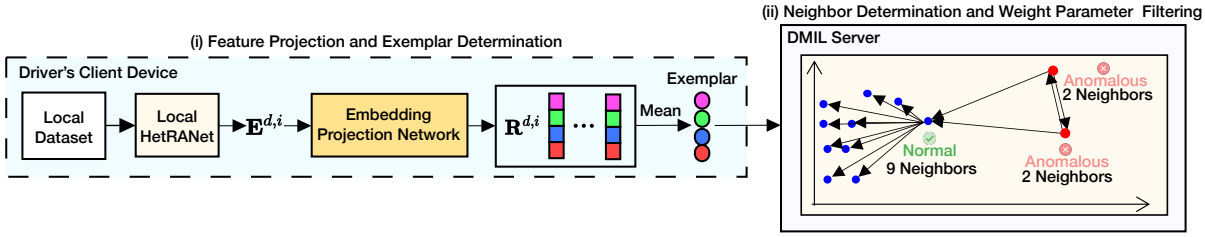


Fig. 13. Illustration of the anomaly-aware mechanism in AF-DMIL. We note that the steps (i) and (ii) are respectively performed at the client device and the DMIL server sides.

• **Design Motivations.** Our proposed anomaly-aware parameter filtering aims to determine the relative importance of the drivers' maneuver patterns and identify the anomalies that have the significant deviation in terms of maneuver patterns. Then AF-DMIL filters away the corresponding model weight parameters before the federated DMIL is performed in the global model training. The key idea of AF-DMIL is to project the maneuver feature embeddings from the high-dimensional space to a low-dimensional one that preserves the maneuver patterns. Such projection helps characterize the key aspects of maneuver patterns that can further discern the anomalous maneuvers. AF-DMIL leverages their mutual closeness to determine the anomalous ones and mitigate their impacts in the federated DMIL. AF-DMIL determines the participation weight of the client device (corresponding to a participating driver) with respect to each maneuver class $k \in \{1, \ldots, K\}$, gains the fine-grained understanding regarding each driver's contributions to federated DMIL, and filters away the model weight parameters that correspond to the anomalous maneuver patterns.

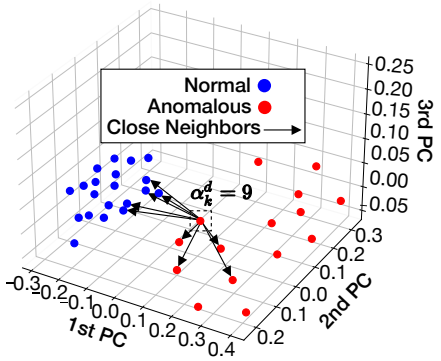


Fig. 14. Illustration of our participation weight determination. For instance, the data point highlighted has $\alpha_k^d = 9$ close neighbors and has a low density in its neighborhood.

• **Detailed Designs.** To this end, we present the detailed designs of our anomaly-aware parameter filtering in Fig. 13, which consists of the following two major steps: (i) feature projection and exemplar determination, and (ii) neighbor determination and parameter filtering.

(i) Feature Projection and Exemplar Determination: As illustrated in Fig. 13*(i)*, for each client device of driver $d$, we project the feature embeddings output from the local HetRANet model (at the client device side) through an embedding projection network based on FC layers, and obtain their low-dimensional projections. Such a

projection from the high-dimensional feature space of the driver maneuvers to a low-dimensional one helps enable the driver maneuver pattern upload to the DMIL server with privacy-preserving implications [19, 45]. Furthermore, AF-DMIL finds the neighbors of each client device with respect to each maneuver class in order to form fine-grained characterization regarding each driver's contributions to the federated DMIL.

Specifically, for each maneuver class $k$, the embedding projection network integrates a total of $B_8$ consecutive FC layers (with the ReLU activation) to project the resulting feature embeddings from the local HetRANet at the driver $d$, denoted as $\mathbf{E}^{d,i} \in \mathbb{R}^{(q_1+q_2+q_3)}$ ($i \in \{1, \ldots, N^d\}$) in Eq. (12). Each of the first ($B_8$-1) FC layers has $c_7$ neurons and the last one has $q_4$ neurons. This way, we process $\mathbf{E}^{d,i}$ and obtain the projected low-dimensional features as $\mathbf{R}^{d,i} \in \mathbb{R}^{q_4}$ ($q_4 < q_1 + q_2 + q_3$) for all $N^d$ maneuvers of a driver $d$. We then adopt the softmax function (similar to Eq. (13)) upon the projected low-dimensional features $\mathbf{R}^{d,i}$, and further train the embedding projection network with ground-truth maneuver classes with cross-entropy loss function (Eq. (17)) to ensure that the low-dimensional projected features maintain the essential maneuver patterns from the high-dimensional data.

Then, for each maneuver class $k$ studied, AF-DMIL finds the mean of the projected features ($\mathbf{R}^{d,i}$)'s with respect to all the $N^d$ maneuvers as the exemplar of the maneuver class for a driver $d$, and uploads all the exemplars as the maneuver patterns of the driver to the DMIL server. For each driver $d$, we denote the $k$-th exemplar as $\mathbf{e}_k^d$ ($k \in \{1, \ldots, K\}$), and we let $\mathbf{\Gamma}^d = \left\{\mathbf{e}_1^d, \ldots, \mathbf{e}_K^d\right\}$ represent the set of a total of derived $K$ exemplars (corresponding to the number of the maneuver classes).

(ii) Neighbor Determination and Weight Parameter Filtering: Given the collected model weight parameters and the exemplars, the DMIL server further leverages the exemplars to determine the neighbors of the client devices. Then the DMIL server identifies the client devices whose maneuver patterns deviate significantly from others', and determines the anomalous maneuver patterns. Our proposed design aims to examine the neighborhood of each exemplar that represents the driver maneuver patterns, and determine those that are significantly far away from others by quantifying their mutual closeness relative to the entire dataset.

Specifically, to adaptively determine the relative closeness of different exemplars, AF-DMIL first finds the average distance of all exemplar pairs for the same maneuver class $k$ across all the $U$ participating drivers in a communication round of federated DMIL, i.e.,

$$\Delta_k = \frac{2}{U\,(U-1)} \left( \sum_{d \neq d'} \xi\left(\mathbf{e}_k^d, \mathbf{e}_k^{d'}\right) \right), \tag{14}$$

where $\mathbf{e}_k^d \in \Gamma^d$, $\mathbf{e}_k^{d'} \in \Gamma^{d'}$, and $\xi(\mathbf{e}_k^d, \mathbf{e}_k^{d'})$ represents the Euclidean distance between the exemplars $\mathbf{e}_k^d$ and $\mathbf{e}_k^{d'}$. Then, for any given pair of exemplars $\mathbf{e}_k^d$ and $\mathbf{e}_k^{d'}$, the DMIL server decides that they are the close neighbors with each other if their mutual Euclidean distance is no greater than the average distance of all the exemplar pairs at the server, i.e.,

$$\psi\left(\mathbf{e}_k^d, \mathbf{e}_k^{d'}\right) = \begin{cases} 1, & \text{if } \xi\left(\mathbf{e}_k^d, \mathbf{e}_k^{d'}\right) \leq \Delta_k; \\ 0, & \text{otherwise.} \end{cases} \tag{15}$$

Given above, the DMIL server counts the total number of the close neighbors of driver $d$'s exemplars ($\psi\left(\mathbf{e}_k^d, \mathbf{e}_k^{d'}\right) = 1$) in each communication round of the federated DMIL, and finds the participation weight of driver $d$ (regarding the $k$-th maneuver class) based on the number of close neighbors, i.e.,

$$\alpha_k^d = \sum_{d' \neq d}^{U} \psi\left(\mathbf{e}_k^d, \mathbf{e}_k^{d'}\right). \tag{16}$$

Then, given the total close neighbors of an exemplar for a driver $d$, the DMIL server identifies it as an anomalous maneuver pattern if it does not have enough close neighbors, i.e., $\alpha_k^d < \delta \cdot U$ ($\delta = 0.8$ in our study and $U$ is the number of the client devices involved per communication round), and the model weight parameters of

this client device are filtered and excluded from the federated DMIL at this communication round. Then AF-DMIL follows the paradigm of federated averaging [41] and aggregates the remaining model weight parameters for the global model training.

We illustrate an example of the projected embeddings for the left turn maneuver class (visualized through principle component analysis) in Fig. 14 for multiple drivers in a communication round. Here we project the embeddings into a 3-D space based on the first three principal components (PCs). We can observe that the determination of the close neighbors helps AF-DMIL identify the anomalous maneuver patterns and reduce the impacts from the corresponding client devices in the federated DMIL.

## 6 EXPERIMENTAL STUDIES

We first present the experimental settings including the baseline approaches in Sec. 6.1. Then, we present our experimental results in terms of maneuver identification and federated DMIL in Sec. 6.2.

### 6.1 Experimental Settings

• **Baseline Comparison:** To evaluate the DMIL performance, we compare the performance of AF-DMIL with the following models.

- ResNet-P and ResNet-T: which leverage the residual networks (ResNet) [26] to process the spectral and time series representations, respectively.
- ALSTM: which adapts two LSTM layers with 64 hidden units with the attention mechanism [63] and processes the time series representations.
- Conv2D-P and Conv2D-T: which leverage three Conv2D layers with kernels of sizes $3 \times 3$, $5 \times 5$, and $7 \times 7$, and 32 hidden units to process the spectral and time series representations, respectively.
- Conv1D-T: which implements three 1-dimension convolutional (Conv1D) layers to process the time series representations.
- LSTM and GRU: which respectively process the time series representations with LSTM and gated recurrent unit (GRU).
- ICGN [18]: which adapts the inception network [58], convolutional neural networks, and gated recurrent unit (GRU) layers to process the time series, spectral, and statistical feature representations.
- TED [77]: which adapts the transformer encoder architecture to process the time series, spectral, and statistical feature representations.
- ECN [42]: which implements an ensemble convolutional network based on 1-D convolutional neural network, recurrent network, and auto-encoder neural network to process the time series representations.
- ST-HAR [1]: which forms the spatio-temporal neural network by combining LSTM layers with convolutional residual layers to process the time series representations.
- MWConv [71]: which implements a multi-sliding window fusion mechanism based on convolutional layers to process the statistical features extracted from time series representations. We use three sliding windows of size 10, 8, and 6 seconds to extract the features.
- TDBC [67]: which adopts the transformer encoder-decoder architecture to process the time series representations.
- SVM, RF, and GBDT: Support vector machine (SVM), random forest (RF) [7], and gradient boosting decision tree (GBDT) [69] are further considered to process the statistical feature representations.

In addition to conventional federated stochastic gradient descent (FedSGD) [10] and federated averaging (FedAvg) [41], we have also compared the performance of our proposed anomaly-aware federated DMIL with the following distributed learning mechanisms:

- AER [25, 37]: which first pre-trains an instance of HetRANet model upon the DMIL server with publicly available normal data (a portion of DS1 in our setting). Then, AER processes the drivers' client devices' local features with the pre-trained model to identify the anomalous maneuvers.
- DIST [39]: which synthesizes the behavioral data distributions based on the local DMIL models (harvested from the drivers' client devices) to identify the anomalous driver maneuvers.

• **Model Parameter Settings:** Unless otherwise stated, we use the following parameters at each phase of DMIL by default.

(a) Driver Maneuver Data Preprocessing and Feature Extraction: We set the size of the sliding window $\omega$ to 10 seconds (400 samples on average) with an overlap of 50% to segment the sensor time series. As different datasets may have different sampling frequencies in the driver maneuver data collection, we have performed the IMU sensor data resampling during the preprocessing and empirically set the resampling as 40Hz.

(b) Driver Maneuver Interaction Representation Learning and Identification: We set our model parameters of HetRANet as follows. For the spectral channel attention module, we adopt the $3 \times 3$ kernel and 6 filters of the Conv2D layer (Fig. 9). In addition, in order to calculate the channel attention scores (Fig. 9), we use $B_1 = 2$ Conv2D layers which have three and six $1 \times 1$ kernel filters, respectively. After the channel attention, we incorporate a total of $B_2 = 3$ DT blocks, each of which consists of $B_3 = 2$ dense blocks (Fig. 10). For the temporal sequence attention module, we use $B_5 = 1$ LSTM layer with $l_1 = 8$ units (Fig. 11). Each of the spectral attention, time sequence attention, and statistical feature learning modules uses three FC layers, i.e., $B_4 = B_6 = B_7 = 3$ for $\mathbf{E}^{d,i}_{\text{Spec}}$, $\mathbf{E}^{d,i}_{\text{Temp}}$, and $\mathbf{E}^{d,i}_{\text{Stat}}$. Among them, the first two FC layers have 32 neurons, i.e., $c_4 = c_5 = c_6 = 32$, while the last layer has 64 neurons, i.e., $q_1 = q_2 = q_3 = 64$. We use a dropout rate of 10% for all the dropout layers.

In evaluating the designs of HetRANet, we have further augmented the driver maneuver datasets to create balanced maneuver classes for DMIL evaluation. Specifically, we have performed data augmentation on DS1 and DS2 to generate a total of 2,500 and 1,000 records for 5 maneuver classes (LT, RT, UT, NA, and NB), i.e., 500 and 200 records per maneuver class for DS1 and DS2, respectively. In evaluating the HetRANet, we leverage 80% of the datasets to cross-validate the model (i.e., five-fold cross-validation) in a centralized DMIL setting. We note that our model evaluation is performed in a user-independent setting (say, driver-independent). In other words, a driver's maneuver data will be either inside the training data or the validation data, and therefore the DMIL model will not capture the driver-dependent correlation during the cross-validation.

(c) Anomaly-aware Federated Learning: At each client device, AF-DMIL iterates the local model update for five times (with the local minibatch size as 32) before communicating with the DMIL server for global model updates. For the embedding projection network, we use three FC layers, $B_8 = 3$. Among them, the first two layers have $c_7 = 32$ neurons, while the last one has $q_4 = 16$ neurons. We note that unlike the evaluation of HetRANet in (b), our evaluation of the anomaly-aware federated learning does not use the augmented datasets, and aims to emulate the real-world scenarios of federated DMIL when different numbers of maneuver classes or types may be uploaded. We demonstrate the accuracy of AF-DMIL and other DMIL baseline approaches after 20 communication rounds. At each communication round, the DMIL server selects $U = 25$ (i.e., the global batch size) client devices (drivers) to participate. We use the Adam optimizer for the model training, and the learning rates at the client device and the DMIL server are both set as 0.01.

In our experimental studies with the DS1 and DS3 (see Sec. 3.1), we consider by default the local datasets at 15% of the participating client devices contain the anomalous maneuvers (i.e., the aggressive ones from DS1 and DS3). For each driver that is selected to have anomalous maneuver patterns in a communication round, we infuse the aggressive maneuvers from the DS1 and DS3 into the normal maneuvers that have the same types of maneuvering, and therefore each maneuver class of the local dataset becomes a mixture of aggressive and normal maneuvers (say, aggressive and normal left turns).

• **Experimental Settings:** We have performed all the experiments on a deep learning HPC with four GPUs of GeForce RTX 3090 (with 24GB GDDR5 memory), an AMD Ryzen Threadripper 3960X 24-Core CPU, and 128GB RAM. We have implemented all the models studied with TensorFlow 2.6.2 and Python 3.6.9. In terms of computation overhead, our data preprocessing, `HetRANet` training, and DMIL prediction respectively takes 26.92ms, 3.72ms, and 0.23ms per sample. We have also evaluated the performance of `AF-DMIL` on Google Pixel 3, where the model training takes 11.71ms per sample, and extracting and recognizing each maneuver takes 47.20ms on average, which suffices to support efficient and ubiquitous DMIL in practice. We also evaluate the performance of DMIL under the conventional federated learning based on federated stochastic gradient descent (`FedSGD` [10]) and federated averaging (`FedAvg` [41]) to validate our proposed anomaly-aware federated DMIL designs.

• **Performance Metrics:** We evaluate the accuracy of DMIL, i.e., the number of correct predictions versus the total number of driver maneuvers, as the performance metric for DMIL. In addition, by considering the normal and anomalous maneuvers as positive and negative cases, respectively, we have included the precision, recall, F1 score, and false negative rate (FNR) of our proposed anomaly-aware federated learning in identifying the anomalous maneuver patterns.

## 6.2 Experimental Results

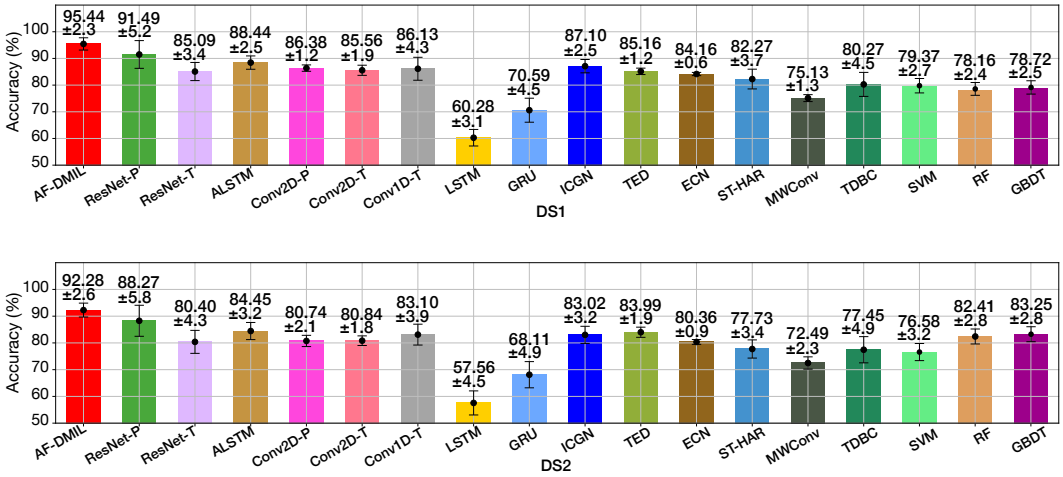


Fig. 15. DMIL performance of different approaches (%) with respect to DS1 and DS2.

• **Overall Performance:** We first compare the performance (in terms of mean accuracies and standard deviations) of `AF-DMIL` in identifying the driver maneuvers in Fig. 15. We can observe in Fig. 15 that `AF-DMIL` outperforms the other baseline approaches on average by 14.01% and 13.41% on DS1 and DS2, respectively. Specifically, we can see that the deep learning-based approaches, such as `ResNet-T`, `ICGN`, `ST-HAR`, and `TDBC` cannot provide comprehensive representation learning of the heterogeneous maneuver patterns upon the complex driver maneuvers, thus yielding lower DMIL accuracy and fewer performance variations. Traditional machine learning methods, such as `SVM`, `RF`, and `GBDT`, cannot effectively characterize the high-dimensional driver maneuver patterns, and hence achieve lower DMIL accuracy. In addition, we have also observed from the performance of the baseline approaches that it is overall more challenging to distinguish the normal acceleration (NA) from the normal braking (NB). This is mainly due to the similar sensor readings along different axes despite the forward direction, and the baseline approaches did not further differentiate the important axes.

Unlike the above-mentioned approaches, our `HetRANet` within `AF-DMIL` extracts and differentiates the complex driver maneuver patterns and thus outperforms the other baselines in the DMIL (including `ICGN` and `TED` that

take in the three different representations). Furthermore, thanks to the densely-connected convolutional neural networks, our HetRANet captures the spectral representations that are related to the driver's behaviors and interactions when steering the vehicle, and learns the heterogeneous representations for enhanced DMIL accuracy.

• **Model Ablation Studies:** We have conducted extensive model ablation studies to evaluate the different design components in AF-DMIL. Specifically, we compare the complete model of AF-DMIL (labeled as (1)) with the following variations: AF-DMIL without (2) spectral attention, (3) time sequence attention, (4) statistical feature learning modules, and (5) AF-DMIL without dense transition, as well as with (6) spectral representations only, (7) time series only, and (8) statistical features only.

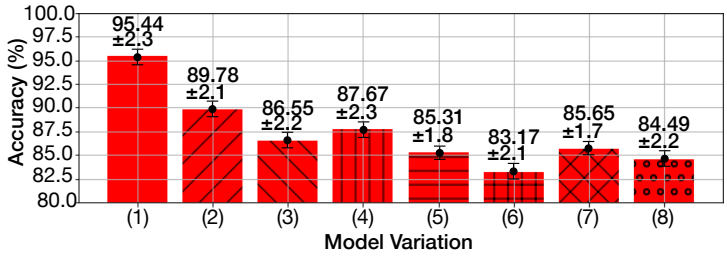


Fig. 16. Model ablation studies for AF-DMIL.

We demonstrate the results of the ablation studies in Fig. 16. We can observe that the heterogeneous representations captured by the three modules in HetRANet are important for accurate DMIL, and hence their removals yield results of lower DMIL accuracy compared with (1). Furthermore, the DMIL performance degradation from that of (1) to that of (5) has demonstrated the importance of our proposed densely-connected convolutional neural networks. Thanks to the designs of dense blocks and transition blocks, our HetRANet can extract and learn the important maneuver patterns for accurate DMIL. We can observe that the time series representation only (labeled as (7)) tend to have the lowest performance drop from (1), which is mainly because the time series representation serves as the direct indicator of the maneuver. However, all three heterogeneous representations are important for the DMIL, and simple reliance upon only one of them (variations (6)–(8)) may not necessarily capture driver maneuver patterns and therefore yield lower accuracy.

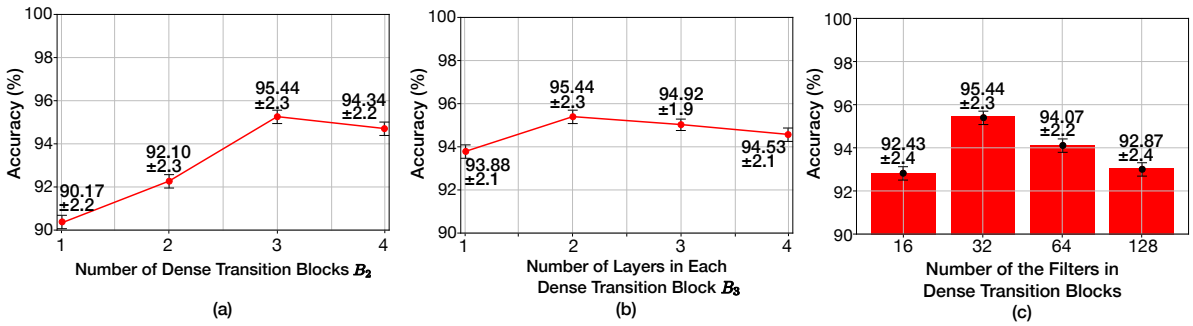


Fig. 17. Sensitivity studies of AF-DMIL on the model parameter settings (a) the number of DT blocks; (b) number of Conv2D layers per DT block; and (c) the number of filters within the Conv2D layers of the DT block.

• **Model Parameter and Data Processing Sensitivity Studies:** We have conducted sensitivity studies on the important model parameters of AF-DMIL in Fig. 17. Specifically, we show the impacts of the number of dense transition blocks and Conv2D layers per DT block in Figs. 17(a) and 17(b), respectively. We can observe that when the number of DT blocks ($B_2$ in Fig. 10) and that of Conv2D layers per DT block ($B_3$ in Eq. (7)) are small, increasing their numbers can help improve the performance of AF-DMIL due to the efficient gradient flow within the dense

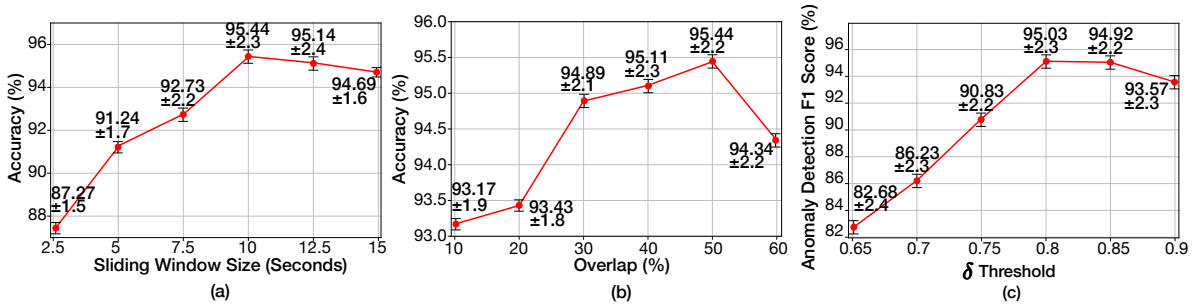


Fig. 18. Sensitivity studies of AF-DMIL regarding the data processing parameters of (a) the size of sliding windows; (b) size of segment overlap; and (c) threshold $\delta$ in the neighbor determination and parameter filtering.

connections [31]. However, as their numbers further increase, we can observe that the performance of AF-DMIL starts to deteriorate as the over-complicated HetRANet model may start to fit upon the noise within the driver maneuver patterns and yield under-generalization in DMIL and lower accuracy. Therefore, we set $B_2 = 3$ and $B_3 = 2$ by default. Similarly, we have conducted the model sensitivity studies upon the number of filters within the Conv2D layers in the DT blocks in Fig. 17(c). We can observe that as the number of filters in the DT blocks increases, the accuracy of AF-DMIL first improves due to extraction of more important spectral features, and then degrades likely owing to the over-complicated structures of DT blocks. In this prototype study, we thus set the number of filters (Conv2D layers) in the DT blocks as 32.

We have further performed the sensitivity studies regarding (a) the sizes of sliding windows, (b) the sizes of the segment overlap (for the sliding windows), and (c) threshold $\delta$ (relative percentage) for the neighbor determination and parameter filtering. Specifically, from Fig. 18(a) we can observe that a small sliding window (say, 2.5s in our studies) may result in the incomplete characterization of the driver maneuver, and therefore the DMIL performance drops. On the other hand, a large sliding window (e.g., larger than 10s) may not necessarily improve the performance, since it may involve multiple maneuvers within one segment, rendering it hard to perform accurate DMIL. Regarding the sizes of overlap, we can see from Fig. 18(b) that a large overlap (say, more than 50%) may result in segments containing multiple maneuvers and low DMIL accuracy, while a small overlap size may not necessarily characterize a single maneuver and hence lead to performance degradation. Furthermore, as illustrated in Fig. 18(c), a threshold $\delta$ that is too large or too small may lead to over-sensitive or under-sensitive detection of the anomalous driver patterns. Based on the above, we set by default the window size as 10s, 50% overlap, and threshold $\delta = 0.8$ in our experimental studies.

• **Studies on Federated DMIL:** We further evaluate our designs in anomaly-aware federated DMIL.

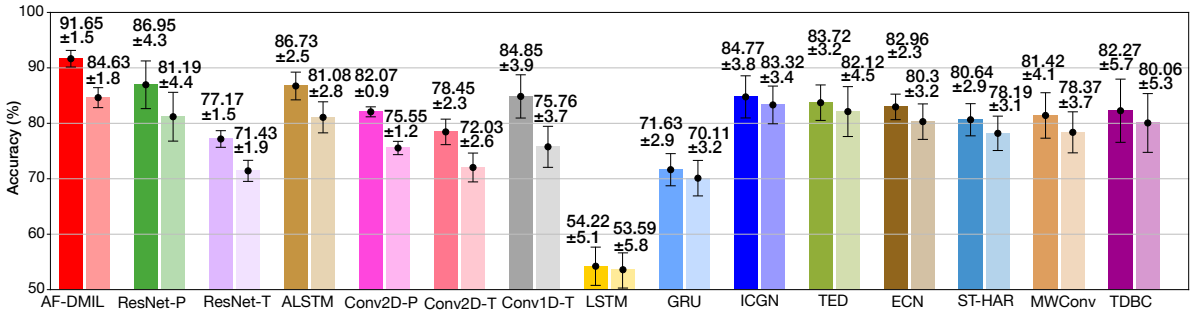


Fig. 19. DMIL performance of different approaches with our anomaly-aware federated learning algorithm (indicated by darker colors) and FedAvg [41] method (%).

– *Improved DMIL Performance*: As illustrated in Fig. 19, we evaluate AF-DMIL and the baseline models with the presence of anomalous behaviors within the maneuver data. Specifically, we leverage our proposed anomaly-aware federated learning algorithm and FedAvg [41] to train the models. We can observe that compared to the FedAvg algorithm, our proposed anomaly-aware federated learning enhances the DMIL performance by identifying and filtering the model weight parameters that correspond to the potentially anomalous driver maneuver patterns. On average, our proposed anomaly-aware federated DMIL further enhances the mean DMIL accuracy by 7.02% for all schemes evaluated in Fig. 19 compared with the other anomaly-agnostic designs.

– *Performance in Identifying Anomalous Maneuvers*: We evaluate the performance of the proposed anomaly-aware federated learning within AF-DMIL to validate its efficiency, effectiveness, and robustness. We further illustrate the precision, recall, F1 measure, and false negative rate (FNR) of our proposed anomaly-aware federated DMIL given different percentages of anomalous driver maneuvers in Fig. 20(a). We also further evaluate the performance of the other distributed learning mechanisms, i.e., AER and DIST, in Figs. 20(b) and 20(c). We can observe that as the percentage of anomalous maneuvers increases, the precision, F1 measure, and recall in identifying the anomalous maneuvers start to decrease, while the FNR increases. Despite these, our proposed designs within AF-DMIL still achieve a more robust performance (say, in general with precision/F1 measure/recall over 90% and with FNR below 10%) than the other two approaches. We can see that AF-DMIL achieves on average 15.68%, 13.10%, and 14.26% higher in terms of precision, recall, and F1 score, and 13.35% lower FNR, compared with the other distributed learning mechanisms evaluated.

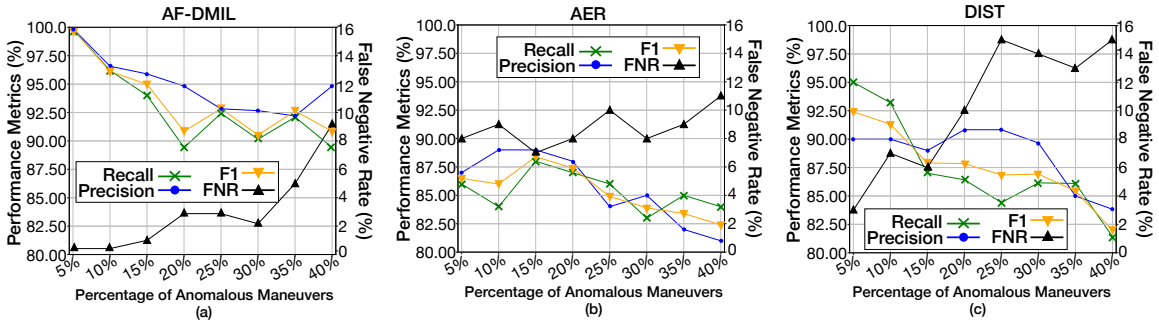


Fig. 20. Precision, recall, F1 score, and false negative rate (FNR) of AF-DMIL compared with other mechanisms.

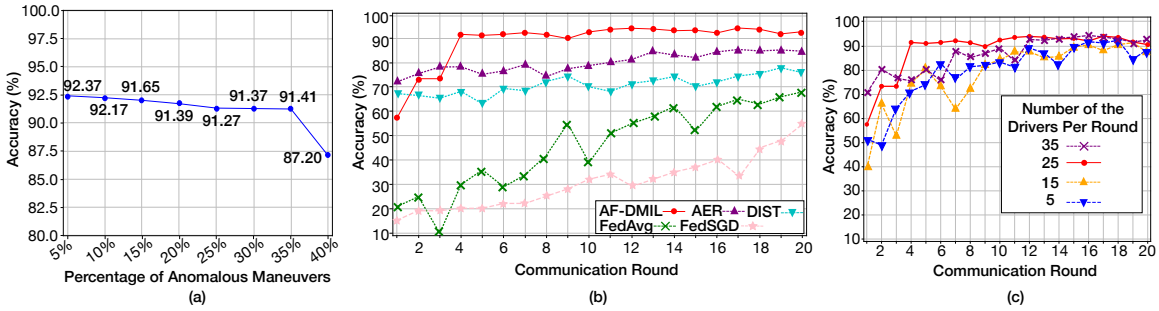


Fig. 21. (a) Accuracy (%) of AF-DMIL vs. percentage of anomalous maneuver patterns for AF-DMIL. (b) Performance convergence studies of different model training approaches. (c) Performance convergence studies given different numbers of participating drivers for AF-DMIL.

– *Dynamic Federated DMIL Evaluation*: In addition, we further show in Fig. 21 the federated DMIL performance of AF-DMIL in terms of (a) the percentage of anomalous maneuver patterns within the datasets, (b) DMIL performance convergence, and (c) number of participating drivers.

We first demonstrate in Fig. 21(a) the performance of AF-DMIL given different proportions of anomalous maneuver patterns within the training data. Thanks to the augmented anomaly-awareness, we can observe that even with a large percentage of the anomalous maneuvers included, AF-DMIL can still achieve overall high DMIL.

We then show in Fig. 21(b) the performance convergence of AF-DMIL with respect to communication rounds. Our proposed anomaly-aware federated DMIL in AF-DMIL identifies and filters away the anomalous maneuver patterns, and hence with the filtered maneuver patterns AF-DMIL converges much faster (i.e., with fewer communication rounds) than other learning designs such as FedAvg and FedSGD which do not account for anomalous maneuver patterns, as well as the anomaly detection designs in AER and DIST. We note that both AER and DIST assume the known normal distributions of the input behavior patterns, which, however, may not adapt to the complex and dynamic driver maneuvers, thus yielding low DMIL accuracy. With the anomaly-aware designs, our proposed DMIL achieves on average 29.20% fewer communication rounds than the above-mentioned approaches before converging to a reasonable DMIL accuracy (say, 85%).

We show in Fig. 21(c) the performance convergence of AF-DMIL under different numbers of participating client devices (drivers). We can observe that in general a small number of client devices (per communication round) may slow down the global HetRANet model training process, while enrollment of more client devices might make it faster for AF-DMIL to converge to high DMIL accuracy, at the potential costs of more communication overheads between the client devices and the DMIL server. Given the ubiquitous computing scenarios such as large-scale DMIL, we can observe that our proposed AF-DMIL can achieve in general high accuracy and scalable system deployment.

## 7 DEPLOYMENT DISCUSSION

We discuss the deployment of AF-DMIL as follows.

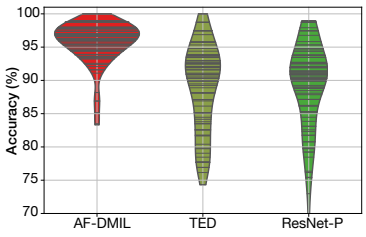


Fig. 22. Performance under leave-one-user-out settings.

– *Settings in Federated DMIL*: Our proposed anomaly-aware federated DMIL designs provide the DMIL insights, including the embedding projection and mutual closeness of exemplars, from an honest-majority setting (i.e., the majority of the participating drivers are contributing normal maneuver data) [22, 53]. Further defense measures for handling the data poisoning attacks and other data privacy and security designs are outside the scope of our proposed studies, and can be referred to [6]. In addition, our current studies take into account the aggressive maneuvers as the anomalous maneuvers for evaluating the robustness [20, 55] of anomaly-aware federated DMIL, which is considered more challenging than noisy or incorrect maneuver labeling [20] based on our experimental observations. We have evaluated the scenarios with maneuver class mislabeling such as the case that a left turn is mislabeled as a right turn, and observed that the maneuver class mislabeling (e.g., due to noisy measurements, improper segmentation, and human errors) can be more detectable than the aggressive ones by the anomaly-federated DMIL. This is mainly due to the discernible differences across the different maneuver types. Therefore, we focus on detecting the anomalous maneuvers (e.g., aggressive), which is important for other ubiquitous computing (e.g., ADAS) applications such as driving style monitoring and feedback for more accountable driving [14, 55].

– *Labeling Driver Maneuver Classes*: In our current DMIL settings, similar to other federated human activity recognition studies [35, 54, 66], we consider the driver maneuver class labels are provided and fed to the DMIL after the driver maneuver data collection is performed. We note that the driver maneuver class labeling can be performed through explicit (say, through recorded videos segmentation [79] or GPS trajectories [76]) or implicit (say, via model self-training [64] or semi-supervised learning [16]) mechanisms, which is outside the scope of our current studies and is part of our future work.

– *Supporting Large-scale Anomaly-Aware Federated DMIL*: We note that in practice the comparison operation required for the anomaly detection could be facilitated through parallelism (e.g., GPU and existing factorization optimization [40]) or hierarchical computation designs. This way, the overall computation time can be significantly reduced. For instance, the anomaly-aware parameter filtering can be performed in a hierarchical manner [73], i.e., partitioning the harvested maneuver data in terms of space (say, different districts of a city), time (e.g., different periods of a day), and other contextual factors, to reduce the comparison operations and support metropolitan-scale DMIL. We also note that distribution shifts – that is, the training and testing datasets may not follow the same distribution – generally exist due to different cars, data collection settings, and participating drivers, etc. We have evaluated `AF-DMIL` under the leave-one-user-out setting (i.e., training the model upon other drivers' data and validate on a single driver) in Fig. 22, and show the violin plot of the accuracy values of one hundred drivers (compared with `TED` and `ResNet-P`), where the wider regions indicate higher probability densities. We have observed that `AF-DMIL` captures the important maneuver patterns in characterizing the heterogeneous representations and mitigates the impacts (e.g., individual differences). In the interest of space, and our current focus, further adaptation, such as meta model adaptation [35], will be considered in our future studies.

– *Extension to Broader DMIL Problems*: Despite our current focus on DMIL based on the IMU sensors, our model designs of `HetRANet` can be also extended to asynchronous federated DMIL [11], other maneuver classes (e.g., left/right lane changes), as well as additional sensing modalities (e.g., LiDAR) that are increasingly pervasive in the emerging ADAS and CAV techniques [4, 27, 33].

## 8 CONCLUSION

We have designed `AF-DMIL`, a novel anomaly-aware federated driver maneuver interaction learning system, toward ubiquitous driver maneuver interaction learning. Leveraging the smartphone IMU sensor data, we have designed and derived three heterogeneous representations for `AF-DMIL` regarding spectral, time series, and statistical features. In order to capture and identify the complex patterns within driver maneuver behaviors, we have designed a heterogeneous representation attention network (`HetRANet`) based on spectral channel attention learning, temporal sequence attention learning, and statistical feature learning. In addition, we have integrated a novel anomaly-aware federated learning algorithm to identify the anomalous maneuvers (e.g., aggressive maneuvers), and mitigate their negative impacts upon the federated DMIL. We have conducted extensive real-world driver data analytics and experimental studies on three real-world datasets (one is collected on our own) to evaluate the prototype of `AF-DMIL`, demonstrating `AF-DMIL`'s accuracy, adaptivity, and robustness compared to the state-of-the-art DMIL.

## ACKNOWLEDGMENTS

We thank the editors and anonymous reviewers for the constructive feedback. This project is supported, in part, by the National Science Foundation (NSF) under Grant 2239897, Google Research Scholar Program Award (2021—2022), NVIDIA Applied Research Accelerator Program Award (2021—2022), and the Connecticut Transportation Institute (CTI).

## REFERENCES

[1] Mohamed Abdel-Basset, Hossam Hawash, Ripon K Chakrabortty, Michael Ryan, Mohamed Elhoseny, and Houbing Song. 2020. ST-DeepHAR: Deep learning model for human activity recognition in IoHT applications. *IEEE IoT-J* 8, 6 (2020), 4969–4979.

[2] Armstrong Aboah, Yaw Adu-Gyamfi, Senem Velipasalar Gursoy, Jennifer Merickel, Matt Rizzo, and Anuj Sharma. 2023. Driver Maneuver Detection and Analysis Using time series Segmentation and Classification. *Journal of Transportation Engineering, Part A: Systems* 149, 3 (2023), 04022157.

[3] Apple. 2021. Apple Developer Documents - Rotation Matrix. Retrieved October 4, 2021 from https://developer.apple.com/documentation/coremotion/cmattitude/1616139-rotationmatrix

[4] Stephanie Balters, Matthew L Mauriello, So Yeon Park, James A Landay, and Pablo E Paredes. 2020. Calm commute: Guided slow breathing for daily stress management in drivers. *Proc. ACM IMWUT* 4, 1 (2020), 1–19.

[5] Mohammad Mahdi Bejani and Mehdi Ghatee. 2019. Convolutional neural network with adaptive regularization to classify driving styles on smartphones. *IEEE T-ITS* 21, 2 (2019), 543–552.

[6] Arjun Nitin Bhagoji, Supriyo Chakraborty, Prateek Mittal, and Seraphin Calo. 2019. Analyzing federated learning through an adversarial lens. In *Proc. ICML*. PMLR, 634–643.

[7] Manuel Ricardo Carlos, Luis C González, Johan Wahlström, Graciela Ramírez, Fernando Martínez, and George Runger. 2019. How smartphone accelerometers reveal aggressive driving behavior?—The key is the representation. *IEEE T-ITS* 21, 8 (2019), 3377–3387.

[8] Dongyao Chen, Kyong-Tak Cho, Sihui Han, Zhizhuo Jin, and Kang G Shin. 2015. Invisible sensing of vehicle steering with smartphones. In *Proc. ACM MobiSys*. 1–13.

[9] Dongyao Chen and Kang G Shin. 2019. TurnsMap: enhancing driving safety at intersections with mobile crowdsensing and deep learning. *Proc. ACM IMWUT* 3, 3 (2019), 1–22.

[10] Jianmin Chen, Xinghao Pan, Rajat Monga, Samy Bengio, and Rafal Jozefowicz. 2016. Revisiting Distributed Synchronous SGD. *arXiv preprint arXiv:1604.00981* (2016).

[11] Yujing Chen, Yue Ning, Martin Slawski, and Huzefa Rangwala. 2020. Asynchronous online federated learning for edge devices with non-iid data. In *Proc. IEEE Big Data*. IEEE, 15–24.

[12] Zheng Chen, Chung-Hsuan Hu, and Erik G Larsson. 2021. Anomaly-Aware Federated Learning with Heterogeneous Data. In *IEEE Proc. ICAS*. IEEE, 1–5.

[13] Young Ah Choi, Kyung Ho Park, Eunji Park, and Huy Kang Kim. 2021. Unsupervised driver behavior profiling leveraging recurrent neural networks. In *Information Security Applications: 22nd International Conference, WISA 2021, Jeju Island, South Korea, August 11–13, 2021, Revised Selected Papers 22*. Springer, 28–38.

[14] Sara Coleman. 2022. Road Rage Statistics 2022. https://www.bankrate.com/insurance/car/road-rage-statistics/.

[15] Yuepeng Cui, Jianqing Wu, Hao Xu, and Aobo Wang. 2020. Lane change identification and prediction with roadside LiDAR data. *Optics & Laser Technology* 123 (2020), 105934.

[16] Enmao Diao, Jie Ding, and Vahid Tarokh. 2022. SemiFL: Semi-supervised federated learning for unlabeled clients with alternate training. *Proc. NeurIPS* 35 (2022), 17871–17884.

[17] DriversEd. 2020. *2020 'Reckless Ridesharing' Report: An Alarming Number of Rideshare Drivers Are Engaging in Dangerous Behind-the-Wheel Behaviors*. https://driversed.com/trending/reckless-ridesharing-report-2020,

[18] Nidhi Dua, Shiva Nand Singh, Vijay Bhaskar Semwal, and Sravan Kumar Challa. 2023. Inception inspired CNN-GRU hybrid network for human activity recognition. *Multimedia Tools and Applications* 82, 4 (2023), 5369–5403.

[19] Cynthia Dwork, Kunal Talwar, Abhradeep Thakurta, and Li Zhang. 2014. Analyze Gauss: optimal bounds for privacy-preserving principal component analysis. In *Proc. ACM STOC*. 11–20.

[20] Xiuwen Fang and Mang Ye. 2022. Robust federated learning with noisy and heterogeneous clients. In *Proc. IEEE/CVF CVPR*. 10072–10081.

[21] Jair Ferreira, Eduardo Carvalho, Bruno V Ferreira, Cleidson de Souza, Yoshihiko Suhara, Alex Pentland, and Gustavo Pessin. 2017. Driver behavior profiling: An investigation with different smartphone sensors and machine learning. *PLoS one* 12, 4 (2017), e0174959.

[22] Clement Fung, Chris JM Yoon, and Ivan Beschastnikh. 2018. Mitigating sybils in federated learning poisoning. *arXiv preprint arXiv:1808.04866* (2018).

[23] Google. 2021. Google Developer Documents - Rotation Matrix. Retrieved October 4, 2021 from https://developer.android.com/guide/topics/sensors/sensors_position

[24] A Grossmann, Richard Kronland-Martinet, and J Morlet. 1990. Reading and understanding continuous wavelet transforms. In *Wavelets*. Springer, 2–20.

[25] Zhipin Gu and Yuexiang Yang. 2021. Detecting malicious model updates from federated learning on conditional variational autoencoder. In *IEEE IPDPS*. IEEE, 671–680.

[26] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *Proc. IEEE CVPR*. 770–778.

[27] Liang He, Yuanchao Shu, Youngmoon Lee, Dongyao Chen, and Kang G Shin. 2020. Authenticating Drivers Using Automotive Batteries. *Proc. ACM IMWUT* 4, 4 (2020), 1–27.

[28] Suining He and Kang G. Shin. 2019. Spatio-Temporal Adaptive Pricing for Balancing Mobility-on-Demand Networks. *ACM TIST* 10, 4, Article 39 (jul 2019), 28 pages.

[29] Suining He and Kang G. Shin. 2019. Spatio-Temporal Capsule-Based Reinforcement Learning for Mobility-on-Demand Network Coordination. In *Proc. WWW*. 2806–2813.

[30] Suining He and Kang G. Shin. 2022. Spatio-Temporal Capsule-Based Reinforcement Learning for Mobility-on-Demand Coordination. *IEEE TKDE* 34, 3 (2022), 1446–1461.

[31] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. 2017. Densely connected convolutional networks. In *Proc. IEEE CVPR*. 4700–4708.

[32] Hongbo Jiang, Jingyang Hu, Daibo Liu, Jie Xiong, and Mingjie Cai. 2021. DriverSonar: Fine-Grained Dangerous Driving Detection Using Active Sonar. *Proc. ACM IMWUT* 5, 3 (2021), 1–22.
[33] Landu Jiang, Xinye Lin, Xue Liu, Chongguang Bi, and Guoliang Xing. 2018. SafeDrive: Detecting distracted driving behaviors using wrist-worn devices. *Proc. ACM IMWUT* 1, 4 (2018), 1–22.
[34] Alexey Kashevnik, Igor Lashkov, and Andrei Gurtov. 2019. Methodology and Mobile Application for Driver Behavior Analysis and Accident Prevention. *IEEE T-ITS* 21, 6 (2019), 2427–2436.
[35] Chenglin Li, Di Niu, Bei Jiang, Xiao Zuo, and Jianming Yang. 2021. Meta-HAR: Federated representation learning for human activity recognition. In *Proc. The Web Conference*. 912–922.
[36] Pei Li, Mohamed Abdel-Aty, Qing Cai, and Zubayer Islam. 2020. A deep learning approach to detect real-time vehicle maneuvers based on smartphone sensors. *IEEE T-ITS* (2020).
[37] Suyi Li, Yong Cheng, Yang Liu, Wei Wang, and Tianjian Chen. 2019. Abnormal client behavior detection in federated learning. *arXiv preprint arXiv:1910.09933* (2019).
[38] Min Lin, Qiang Chen, and Shuicheng Yan. 2013. Network in network. *arXiv preprint arXiv:1312.4400* (2013).
[39] Bingyan Liu, Yifeng Cai, Ziqi Zhang, Yuanchun Li, Leye Wang, Ding Li, Yao Guo, and Xiangqun Chen. 2021. DistFL: Distribution-aware Federated Learning for Mobile Scenarios. *arXiv preprint arXiv:2110.11619* (2021).
[40] Duhu Man, Kenji Uda, Yasuaki Ito, and Koji Nakano. 2011. A GPU implementation of computing euclidean distance map with efficient memory access. In *Proc. IEEE ICNC*. IEEE, 68–76.
[41] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*. PMLR, 1273–1282.
[42] Debadyuti Mukherjee, Riktim Mondal, Pawan Kumar Singh, Ram Sarkar, and Debotosh Bhattacharjee. 2020. EnsemConvNet: a deep learning approach for human activity recognition using smartphone sensors for healthcare applications. *Multimedia Tools and Applications* 79 (2020), 31663–31690.
[43] Sashank Narain, Triet D Vo-Huu, Kenneth Block, and Guevara Noubir. 2016. Inferring user routes and locations using zero-permission mobile sensors. In *Proc. IEEE S&P*. IEEE, 397–413.
[44] Anna Nedorubova, Alena Kadyrova, and Aleksey Khlyupin. 2021. Human Activity Recognition using Continuous Wavelet Transform and Convolutional Neural Networks. *arXiv preprint arXiv:2106.12666* (2021).
[45] Hung Nguyen, Di Zhuang, Pei-Yuan Wu, and Morris Chang. 2020. AutoGAN-based Dimension Reduction for Privacy Preservation. *Neurocomputing* 384 (2020), 94–103.
[46] NHTSA. 2010. *2020 Fatality Data Show Increased Traffic Fatalities During Pandemic.* https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/811366
[47] Zhenchao Ouyang, Jianwei Niu, and Mohsen Guizani. 2017. Improved vehicle steering pattern recognition by using selected sensor data. *IEEE TMC* 17, 6 (2017), 1383–1396.
[48] Zhenchao Ouyang, Jianwei Niu, Yu Liu, and Xue Liu. 2019. An ensemble learning-based vehicle steering detector using smartphones. *IEEE T-ITS* 21, 5 (2019), 1964–1975.
[49] Salah-Eddine Ramah, Afaf Bouhoute, Karim Boubouh, and Ismail Berrada. 2021. One Step Further Towards Real-Time Driving Maneuver Recognition Using Phone Sensors. *IEEE T-ITS* (2021).
[50] Yi Ren, Chenxu Hu, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu. 2020. Fastspeech 2: Fast and high-quality end-to-end text to speech. *arXiv preprint arXiv:2006.04558* (2020).
[51] Market Resaerh. 2021. *Autonomous Vehicle Market Size, Share & Trends Analysis Report By Application (Transportation, Defense), By Region (North America, Europe, Asia Pacific, South America, MEA), And Segment Forecasts, 2021 - 2030.* https://www.marketresearch.com/Grand-View-Research-v4060/Autonomous-Vehicle-Size-Share-Trends-13145844/
[52] Ankur Sarker and Haiying Shen. 2018. A data-driven misbehavior detection system for connected autonomous vehicles. *Proc. ACM IMWUT* 2, 4 (2018), 1–21.
[53] Muhammad Shayan, Clement Fung, Chris JM Yoon, and Ivan Beschastnikh. 2020. Biscotti: A blockchain system for private and secure federated learning. *IEEE TPDS* 32, 7 (2020), 1513–1525.
[54] Qiang Shen, Haotian Feng, Rui Song, Stefano Teso, Fausto Giunchiglia, Hao Xu, et al. 2022. Federated Multi-Task Attention for Cross-Individual Human Activity Recognition. In *IJCAI*. IJCAI, 3423–3429.
[55] Harpreet Singh and Ankit Kathuria. 2021. Profiling drivers to assess safe and eco-driving behavior–A systematic review of naturalistic driving studies. *Accident Analysis & Prevention* 161 (2021), 106349.
[56] Jun Sun, Tianyi Chen, Georgios B Giannakis, Qinmin Yang, and Zaiyue Yang. 2020. Lazily aggregated quantized gradient innovation for communication-efficient federated learning. *IEEE TPAMI* (2020).
[57] Wei Sun and Kannan Srinivasan. 2022. On the Feasibility of Securing Vehicle-Pavement Interaction. *Proc. ACM IMWUT* 6, 1 (2022), 1–24.
[58] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *Proc. IEEE CVPR*. 1–9.

[59] Mahan Tabatabaie and Suining He. 2023. Naturalistic E-Scooter Maneuver Recognition with Federated Contrastive Rider Interaction Learning. *Proc. ACM IMWUT* 6, 4 (2023), 1–27.
[60] Mahan Tabatabaie, Suining He, and Kang G. Shin. 2023. Cross-Modality Graph-Based Language and Sensor Data Co-Learning of Human-Mobility Interaction. *Proc. ACM IMWUT* 7, 3, Article 125 (Sep 2023), 25 pages.
[61] Mahan Tabatabaie, Suining He, and Xi Yang. 2021. Reinforced Feature Extraction and Multi-Resolution Learning for Driver Mobility Fingerprint Identification. In *Proc. ACM SIGSPATIAL*. New York, NY, USA, 69–80.
[62] Mahan Tabatabaie, Suining He, and Xi Yang. 2022. Driver maneuver identification with multi-representation learning and meta model update designs. *Proc. ACM IMWUT* 6, 2 (2022), 1–23.
[63] Nima Taherifard, Murat Simsek, Charles Lascelles, and Burak Kantarci. 2020. Attention-based event characterization for scarce vehicular sensing data. *IEEE Open Journal of Vehicular Technology* 1 (2020), 317–330.
[64] Chi Ian Tang, Ignacio Perez-Pozuelo, Dimitris Spathis, Soren Brage, Nick Wareham, and Cecilia Mascolo. 2021. SelfHAR: Improving Human Activity Recognition through Self-training with Unlabeled Data. *Proc. ACM IMWUT* 5, 1 (2021), 1–30.
[65] New York Times. 2021. *America Has a GPS Problem.* https://www.nytimes.com/2021/01/23/opinion/gps-vulnerable-alternatives-navigation-critical-infrastructure.html
[66] Linlin Tu, Xiaomin Ouyang, Jiayu Zhou, Yuze He, and Guoliang Xing. 2021. FedDL: Federated Learning via Dynamic Layer Sharing for Human Activity Recognition. In *Proc. ACM SenSys*. 15–28.
[67] Jayant Vyas, Nishit Bhardwaj, Debasis Das, et al. 2022. TransDBC: Transformer for Multivariate Time-Series based Driver Behavior Classification. In *Proc. IEEE IJCNN*. IEEE, 1–8.
[68] Johan Wahlström, Isaac Skog, and Peter Händel. 2017. Smartphone-Based Vehicle Telematics: A Ten-Year Anniversaryy. *IEEE T-ITS* 18, 10 (2017), 2802–2825.
[69] Yan Wang, Tianming Zhao, Fatemeh Tahmasbi, Jerry Cheng, Yingying Chen, and Jiadi Yu. 2020. Driver Identification Leveraging Single-turn Behaviors via Mobile Devices. In *Proc. IEEE ICCCN*. IEEE, 1–9.
[70] Greg Welch, Gary Bishop, et al. 1995. An introduction to the Kalman filter. (1995).
[71] Jie Xie, Kai Hu, Guofa Li, and Ya Guo. 2021. CNN-based driving maneuver classification using multi-sliding window fusion. *Expert Systems with Applications* 169 (2021), 114442.
[72] Chuang-Wen You, Nicholas D Lane, Fanglin Chen, Rui Wang, Zhenyu Chen, Thomas J Bao, Martha Montes-de Oca, Yuting Cheng, Mu Lin, Lorenzo Torresani, et al. 2013. Carsafe app: Alerting drowsy and distracted drivers using dual cameras on smartphones. In *Proc. ACM MobiSys*. 13–26.
[73] Chung-Hsien Yu, Wei Ding, Melissa Morabito, and Ping Chen. 2015. Hierarchical spatio-temporal pattern discovery and predictive modeling. *IEEE TKDE* 28, 4 (2015), 979–993.
[74] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. 2020. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proc. IEEE/CVF CVPR*. 2636–2645.
[75] Hongzheng Yu, Zekai Chen, Xiao Zhang, Xu Chen, Fuzhen Zhuang, Hui Xiong, and Xiuzhen Cheng. 2021. FedHAR: Semi-supervised online learning for personalized federated human activity recognition. *IEEE TMC* (2021).
[76] Stephen A Zekany, Ronald G Dreslinski, and Thomas F Wenisch. 2019. Classifying ego-vehicle road maneuvers from dashcam video. In *Proc. IEEE ITSC*. IEEE, 1204–1210.
[77] George Zerveas, Srideepika Jayaraman, Dhaval Patel, Anuradha Bhamidipaty, and Carsten Eickhoff. 2021. A transformer-based framework for multivariate time series representation learning. In *Proc. ACM SIGKDD*. 2114–2124.
[78] Yang Zheng and John HL Hansen. 2017. Lane-change detection from steering signal using spectral segmentation and learning-based classification. *IEEE TIV* 2, 1 (2017), 14–24.
[79] Yang Zheng, Amardeep Sathyanarayana, and John HL Hansen. 2014. Threshold based decision-tree for automatic driving maneuver recognition using CAN-Bus signal. In *Proc. IEEE ITSC*. IEEE, 2834–2839.
[80] Yujie Zhou, Ruyan Wang, Xingyue Mo, Zhidu Li, and Tong Tang. 2022. Robust Hierarchical Federated Learning with Anomaly Detection in Cloud-Edge-End Cooperation Networks. *Electronics* 12, 1 (2022), 112.
[81] Yuanshao Zhu, Yi Liu, JQ James, and Xingliang Yuan. 2021. Semi-Supervised Federated Learning for Travel Mode Identification From GPS Trajectories. *IEEE T-ITS* (2021).

## APPENDIX

We list the major symbols used in AF-DMIL formulation in Table 2.

Table 2. Major symbols in the designs of AF-DMIL.

| Notations | Definitions | Notations | Definitions |
|---|---|---|---|
| $N^d$ | Total segments for each driver $d$. | $\mathbf{\Gamma}^d$ | The set of all the exemplars for each driver $d$. |
| $\mathbf{p}^{d,i}$, $\mathbf{t}^{d,i}$, $\mathbf{s}^{d,i}$ | The $i$-th spectral, time series, and | $U$ | Total drivers in a communication round. |
| | statistical representations for each driver $d$. | $\mathbf{e}_k^d$ | The $k$-th exemplar of each driver $d$ |
| $\mathbf{z}'_{\mathrm{p}}$, $\mathbf{z}'_{\mathrm{t}}$ | Spectral channel and temporal sequence attention scores. | $\mathbf{R}_k^d$ | The $k$-th feature projections for each driver $d$. |
| $\mathbf{E}^{d,i}$ | Embeddings of the $i$-th driver maneuver interaction. | $K$ | Total number of the maneuver interaction classes. |
| $\varphi^{d,i}[k]$ | Output probability of the $i$-th maneuver record. | $\mathbf{y}^{d,i}$ | Ground-truth label for the $i$-th maneuver of each driver $d$. |
| | for the $d$-th driver and $k$-th maneuver class. | $\alpha_k^d$ | $k$-th participation weight for each driver $d$. |

The cross-entropy loss for model training of HetRANet is formally given by

$$\texttt{loss} \triangleq -\sum_{d=1}^{U}\sum_{i=1}^{N^d}\sum_{k=1}^{K}\left(y^{d,i}[k]\cdot\log\varphi^{d,i}[k]\right), \tag{17}$$

where $N^d$ is the total number of maneuvers for a driver $d \in \{1,\ldots,U\}$.