- **1 Team Name**

  Fighting_zsn

- **2 Provide short summary of the approach, the methods, techniques for processing the data.**

  Title: Road Network Enhanced Transportation Mode Recognition with an Integrated Machine Learning Model

  Participants of the fifth edition of SHL recognition challenge 2023 aim to recognize eight locomotion or transportation modes in a user-independent manner based on motion and GPS sensor data. The "Fighting_zsn" team proposes an integrated machine learning model based on road networks, which is experimentally shown to significantly improve model performance. Initially, time-domain and frequency-domain features are extracted from the provided SHL datasets. Additionally, external knowledge is incorporated to obtain road network features, enriching the dataset. To capture context information, changing and window-based features are derived from the three types of features mentioned above. To address label distribution imbalance, the original training datasets are augmented, ensuring a balanced label distribution for improved training. The original validation set is then utilized for testing the model's performance. After evaluating multiple machine learning models, the integrated model based on XGBOOST, LIGHTGBM, and RF emerges as the most effective choice. This model successfully achieves efficient and accurate recognition of activity modes, attaining a weighted F1 score of 0.80 and an averaged precision score of 0.82.

- **3 Provide your performance for the training dataset (e.g. F1 score).**

  We understand that the requirement is to provide the model's performance on the training set. However, in our case, training the model on the entire original training set and evaluating it on the same set would lead to severe overfitting. As a result, the weighted F1 scores would be exceedingly high, exceeding 98%. We believe that reporting the model's performance on the training set would be meaningless in this context.Instead, we have opted to present the model's results on the test set, which is composed of the combined original validation set. This test set allows us to assess the model's generalization ability and provide a more meaningful evaluation of its performance on unseen data.

  To provide a more accurate measure of the model's performance, we employed an alternative workflow. Firstly, we balance the distribution of labels in the original training set, which comprised Bag_train, Hips_train, Torso_train, and Hand_train, through data augmentation techniques. This resulted in a new training set called New_Bag_train, New_Hips_train, New_Torso_train, and New_Hand_train. Subsequently, we utilize the original validation set, Bag_valid, Hips_valid, Torso_valid, and Hand_valid as test datasets to evaluate the model's generalization ability. We conducted **21** experiments using the designed model, and the weighted F1 scores obtained from these experiments are presented in Table 1.

Table 1. Weighted F1 Scores of 21 Experimental Sets: Training vs. Testing

| Test Set / Train Set | Bag_valid | Hand_valid | Hips_valid | Torso_valid | ALL_four_valid |
|---|---|---|---|---|---|
| New_Bag_train | 0.86 | 0.83 | 0.71 | 0.67 | \ |
| New_Hand_train | 0.74 | 0.79 | 0.69 | 0.66 | \ |
| New_Hips_train | 0.62 | 0.75 | 0.75 | 0.60 | \ |
| New_Torso_train | 0.75 | 0.76 | 0.75 | 0.75 | \ |
| ALL_four_train | 0.83 | 0.85 | 0.76 | 0.75 | 0.80 |

- **4 Based on the training and validation dataset, please predict the performance of your method for the testing dataset (e.g. F1 score).**

  Based on the answer 3, we observed that training with four augmented training sets (ALL_four_train) and testing on a single validation set yielded better results compared to training with a single training set and testing on a single validation set. Therefore, our final model combines the four augmented training sets with the four original validation sets to obtain the best-performing model. Subsequently, we utilize this model to generate the final test results on the original test set.

  The weighted F1 scores of the (ALL_four_train) model on Bag_valid, Hand_valid, Hips_valid, Torso_valid, and ALL_four_valid are 0.83, 0.85, 0.76, 0.75, and 0.80, respectively. It's important to note that the test set and the original validation set are identical, both comprising a combination of data from USER2 and USER3. Therefore, we expect the optimal model to perform similarly on the test set as it did on the original validation set. Based on these observations, we anticipate that our model's performance on the test set would fall within the range of 0.75 to 0.85.
- **5 What kind of computing power did you use? For example: CPU 2-core@2.2GHz, RAM-25G; GPU RTX 3080**

  The experiments are measured on a 32-Core Intel Xeon(R) E52620 v4 CPU@2.10GHz server with 173 GB memory.
- **6 What is the size (KB, MB) of the trained model (e.g., if you save it on the hard disc)?**

  Our model uses an integrated machine learning model, which is used instantly during the training process and does not save the model, so we do not provide a file size for the model.
- **7 How much time does it take to train the model (roughly in minutes, hours, days)?**

  Training with (All_four_train), the training model time is as follows.
  training set: (All_four_train)
  X_train: (4202104, 176)
  Y_train: 4202104
  Time elapsed for training rf: 1058.8 s
  Time elapsed for training LightGBM: 392.15 s
  Time elapsed for training XGBoost: 3307.48 s
  The total training time for the integrated machine learning model is 4758.43s, which is about 79.30 minutes.
- **8 How much time does it take evaluate the test dataset (roughly in seconds, minutes, hours)?**

**8.1 Train with (ALL_four_train), test with (All_four_valid), the test time on the (All_four_valid) is as follows.**

training set : (ALL_four_train) ; testing set : (All_four_valid)
X_val : (575840, 176)
Y_val : 575840
Time elapsed for testing rf 4.04 s
Time elapsed for testing lgb: 4.68 s
Time elapsed for testing xgb: 8.99 s
The total time we spent testing this dataset (All_four_valid) using this integrated machine learning model was 17.71 s.

**8.2 Train with (ALL_four_train) , test with (Torso_valid) ,the test time on the (Torso_valid) is as follows.**

training set : (ALL_four_train) ; testing set : (Torso_valid)
X_val : (143960, 176)
Y_val : 143960
Time elapsed for testing rf 1.0 s
Time elapsed for testing lgb: 1.27 s
Time elapsed for testing xgb: 2.42 s
The total time we spent testing this dataset (Torso_valid) using this integrated machine learning model was 4.69 s.

**8.3 Train with (ALL_four_train) , test with (Bag_valid) ,the test time on the (Bag_valid) is as follows.**

training set : (ALL_four_train) ; testing set : (Bag_valid)
X_val : (143960, 176)
Y_val : 143960
Time elapsed for testing rf 1.03 s
Time elapsed for testing lgb: 1.2 s
Time elapsed for testing xgb: 2.3 s
The total time we spent testing this dataset (Bag_valid)using this integrated machine learning model was 4.53 s.

**8.4 Train with (ALL_four_train) , test with (Hand_valid) ,the test time on the (Hand_valid) is as follows.**

training set : (ALL_four_train) ; testing set : (Hand_valid);
X_val : (143960, 176)
Y_val : 143960
Time elapsed for testing rf 1.02 s
Time elapsed for testing lgb: 1.26 s
Time elapsed for testing xgb: 2.39 s
The total time we spent testing this dataset (Hand_valid) using this integrated machine learning model was 4.67 s.

**8.5 Train with (ALL_four_train) , test with (Hips_valid) ,the test time on the(Hips_valid) is as follows.**

training set : (ALL_four_train); testing set : (Hips_valid)
X_val : (143960, 176)

Y_val : 143960
Time elapsed for testing rf 0.98 s
Time elapsed for testing lgb: 1.13 s
Time elapsed for testing xgb: 2.34 s
The total time we spent testing this dataset (Hips_valid) using this integrated machine learning model was 4.45 s.

- **9 Which sensor modality are you using?**

  We utilize all the motion sensors data, including the accelerometer, gyroscope, and magnetometer, along with GPS sensor data such as GPS coordinates and location information. By leveraging these diverse sensor inputs, we generate a multitude of new features.

- **10 Which classifier are you using? For example, XGBoost, LSTM, CNN etc.**

  We conducted tests on various models including CATBOOST, Random Forest (RF), XGBOOST, LIGHTGBM, Bagging, probability summation models, as well as voting models. After thorough evaluation, we decided to adopt an integrated model based on probability summation, utilizing predictions from RF, XGBOOST, and LIGHTGBM.

- **11 What is your decision window size of the classifier?**

  We utilize the GYR, MAG, and ACC data to construct new aggregated data points based on every 100 raw data points. The timestamp of the new data point is determined by calculating the average timestamp of the 100 data points. The aggregation process involves combining the features of the original data points and deriving overall frequency features. To achieve this, we select a non-repeating window of size 100 for the GYR, MAG, and ACC data.

  Additionally, we directly map the LOCATION and GPS information to the new aggregated data points. In the model prediction phase, we consider the labels of the 100 data points within the same window to be identical, and we only need to predict a single label for that window. After performing calculations, we observe that over 99.99% of the original data labels remain the same within the same window. Hence, it is reasonable to process the data in this manner, transforming the predicted 100 raw data points into a single new data point.

- **12 What is your fusion scheme? For example, early fusion, intermediate fusion, late fusion etc.**

  We employ both early fusion and late fusion techniques in our model. Early fusion involves combining features from different types or sources during the input phase of the model. This means that we merge and integrate the various feature representations at the initial stage of the model, allowing for a unified representation that incorporates information from multiple sources or types.

  On the other hand, late fusion occurs in the output phase of the model. In this approach, we aggregate the output values produced by different machine learning models, combining them to generate an overall output. This can be considered as a form of late fusion, as the fusion of predictions occurs at a later stage in the model pipeline.

- **13 Which type of features are you using? For example, hand-crafted features, raw data, etc.**

  We incorporate features from two distinct sources into our model. The first source is the original SHL data, from which we select meaningful raw data and extract hidden time-domain features, such as speed, as well as compute frequency-domain features like average frequency. These features provide insights into the temporal characteristics of the data.

The second source is an external knowledge base, where we primarily calculate road-network features such as the distance to various Points of Interest (POIs). These features offer contextual information related to the geographical surroundings.

Furthermore, we generate additional changing features and window-based features based on these initial features. The changing features capture the dynamic nature of the data, allowing us to observe how the values evolve over time. The window-based features are derived by aggregating information within specific windows, enabling the model to capture contextual information and patterns over a defined period.

By incorporating a combination of original SHL data features and external road-network features, along with dynamic and contextual features, we aim to capture a comprehensive representation of the data, considering both the temporal aspects and the surrounding environment. This approach enhances the model's ability to capture relevant information and make accurate predictions.

- **14 Do you have any post processing scheme?**

Yes, we apply smoothing techniques to improve the accuracy of our predictions. While individual point predictions may not be entirely accurate, smoothing them with predictions from surrounding points can enhance overall accuracy. We experiment with different smoothing window sizes, such as 80s, 120s, 150s, 180s, 200s, and 300s. Through these experiments, we determine that our model achieves the best performance with a smoothing window of 150s.

In practical application, for each timestamp, the original point predictions are replaced by the mean values within the 150s window. This smoothing operation helps to reduce the impact of individual predictions and provide a more consistent and stable prediction over a larger time frame.

- **15 What software are you using? For example, Matlab, Python, Java etc.**

We use python to do data pre-processing, feature extraction, model building, result prediction, visualization, etc.

- **16 Which library are you using? For example, Matlab Deep Learning, Scikit-learn, Pytorch, etc.**

A number of Python libraries were used throughout the competition, here is a list of some of the libraries used in two ways

**16.1 for data_process and feature extraction**

Pandas, numpy, matplotlib, sklearn.metrics, seaborn, sklearn.metrics, scipy, time, collections, utm, typing, re, tqdm,math, rtree, network, geopandas, pickle, os, pickle

**16.2 for model**

sklearn.model_selection, sklearn.ensemble, sklearn.tree, lightgbm, xgboost, catboost

- **17 If there is a reference paper that you would like to highlight for your algorithm, please cite it here**

[1] Ashqar, H. I., Almannaa, M. H., Elhenawy, M., Rakha, H. A., & House, L. (2018). Smartphone transportation mode recognition using a hierarchical machine learning classifier and pooled features from time and frequency domains. *IEEE Transactions on Intelligent Transportation Systems*, *20*(1), 244-252.

[2] Balabka, D., & Shkliarenko, D. (2021, September). Human activity recognition with AutoML using smartphone radio data. In *Adjunct Proceedings of the 2021 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2021 ACM International Symposium on Wearable Computers* (pp. 346-352).

[3] Breiman, L. (2001). Random forests. *Machine learning*, *45*, 5-32.

[4] Breiman, L. (2001). Using iterated bagging to debias regressions. *Machine Learning*, *45*, 261-277.

[5] Chen, T., & Guestrin, C. (2016, August). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785-794). Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., ... & Liu, T. Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, *30*.

[6] Chen, Y., Zhang, H., Sun, W., & Zheng, B. (2022). RNTrajRec: Road Network Enhanced Trajectory Recovery with Spatial-Temporal Transformer. *arXiv preprint arXiv:2211.13234*.

[7] Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2018). CatBoost: unbiased boosting with categorical features. *Advances in neural information processing systems*, *31*.

[8] Ren, Y. (2021, September). Multiple tree model integration for transportation mode recognition. In *Adjunct Proceedings of the 2021 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2021 ACM International Symposium on Wearable Computers* (pp. 385-389).

[9] Siraj, M. S., Faisal, M. A. A., Shahid, O., Abir, F. F., Hossain, T., Inoue, S., & Ahad, M. A. R. (2020, September). UPIC: user and position independent classical approach for locomotion and transportation modes recognition. In *Adjunct Proceedings of the 2020 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2020 ACM International Symposium on Wearable Computers* (pp. 340-345).

[10] Wang, L., Ciliberto, M., Gjoreski, H., Lago, P., Murao, K., Okita, T., & Roggen, D. (2021, September). Locomotion and transportation Mode Recognition from GPS and radio signals: Summary of SHL Challenge 2021. In *Adjunct Proceedings of the 2021 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2021 ACM International Symposium on Wearable Computers* (pp. 412-422).

[11] Wang, L., Gjoreski, H., Ciliberto, M., Lago, P., Murao, K., Okita, T., & Roggen, D. (2020, September). Summary of the Sussex-Huawei locomotion-transportation recognition challenge 2020. In *Adjunct Proceedings of the 2020 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2020 ACM International Symposium on Wearable Computers* (pp. 351-358).