POPAyI: Muscling Ordinal Patterns for low-complex and usability-aware transportation mode detection

Isadora Cardoso-Pereira, João B. Borges, Aline C. Viana, Antonio A. F. Loureiro, **Heitor S. Ramos** - heitor@dcc.ufmg.br

Federal University of Minas Gerais (UFMG), Brazil, Federal University of Rio Grande do Norte (UFRN), Brazil, Inria, France

Introduction

- Transportation Mode Detection (TMD) involves classifying mobility traces to identify the corresponding transport mode. It can provide valuable insights into the mobility preferences, aiding in epidemic prevention, traffic management, and improving quality of life

Introduction

- Transportation Mode Detection (TMD) involves classifying mobility traces to identify the corresponding transport mode. It can provide valuable insights into the mobility preferences, aiding in epidemic prevention, traffic management, and improving quality of life
- Most TMD literature relies on hand-crafted features (reaching hundreds of features) or computationally- and data-intensive DL methods, which pose challenges for resource-constrained Internet of Things (IoT) scenarios

Introduction

- Transportation Mode Detection (TMD) involves classifying mobility traces to identify the corresponding transport mode. It can provide valuable insights into the mobility preferences, aiding in epidemic prevention, traffic management, and improving quality of life
- Most TMD literature relies on hand-crafted features (reaching hundreds of features) or computationally- and data-intensive DL methods, which pose challenges for resource-constrained Internet of Things (IoT) scenarios
- We present POPAyI, an innovative strategy based on Ordinal Pattern (OP) transformation applied to mobility-related time series that is computationally efficient, scalable, and provides accurate transportation mode detection.

Ordinal Patterns

A symbolic representation of time series dynamics, capturing intrinsic features without relying on predefined models or heavy computational resources. It needs two parameters:

- Embedding dimension $D \in \mathbb{N}$ to determine the length of the patterns

Ordinal Patterns

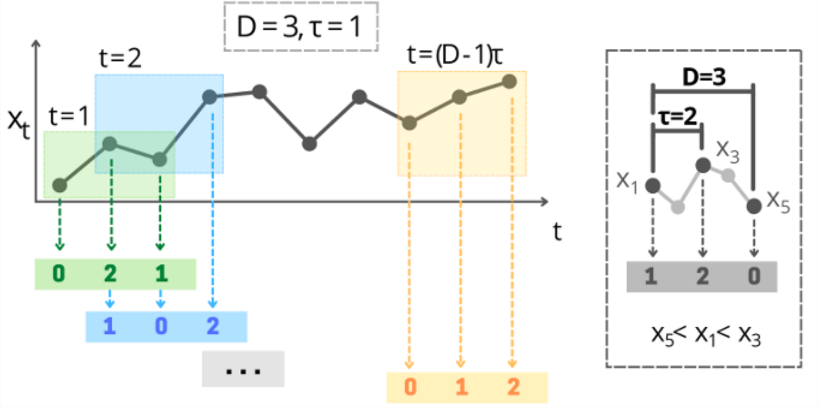
A symbolic representation of time series dynamics, capturing intrinsic features without relying on predefined models or heavy computational resources. It needs two parameters:

- Embedding dimension $D \in \mathbb{N}$ to determine the length of the patterns
- Embedding delay $\tau \in \mathbb{N}$ to define the interval between consecutive data points

Ordinal Patterns

A lightweight solution well-suited for edge computing applications, enabling on-device analytics while preserving privacy

Ordinal Patterns



Ordinal Patterns Limitations

Traditional OPs face limitations when dealing with multidimensional data (e.g., mobility)

1. Absence of amplitude information

Ordinal Patterns Limitations

Traditional OPs face limitations when dealing with multidimensional data (e.g., mobility)

1. Absence of amplitude information

- Be invariant to non-linear monotonic transformation is helpful against noise, but it can be harmful when amplitude contains essential information

Ordinal Patterns Limitations

Traditional OPs face limitations when dealing with multidimensional data (e.g., mobility)

1. Absence of amplitude information

- Be invariant to non-linear monotonic transformation is helpful against noise, but it can be harmful when amplitude contains essential information
- Previous studies have incorporated amplitude into features extracted from OP, but its application to other OP features and representations remains limited

Ordinal Patterns Limitations

Traditional OPs face limitations when dealing with multidimensional data (e.g., mobility)

1. Absence of amplitude information
   - Be invariant to non-linear monotonic transformation is helpful against noise, but it can be harmful when amplitude contains essential information
   - Previous studies have incorporated amplitude into features extracted from OP, but its application to other OP features and representations remains limited
2. Originally defined for univariate time series

Ordinal Patterns Limitations

Traditional OPs face limitations when dealing with multidimensional data (e.g., mobility)

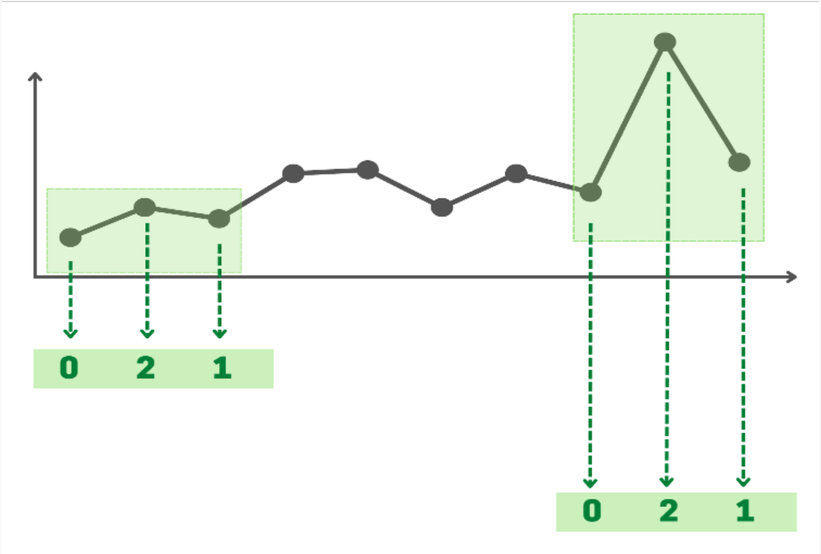
1. Absence of amplitude information

- Be invariant to non-linear monotonic transformation is helpful against noise, but it can be harmful when amplitude contains essential information

- Previous studies have incorporated amplitude into features extracted from OP, but its application to other OP features and representations remains limited

2. Originally defined for univariate time series
    - Previous studies have attempted to extend OP to multivariate data, but faced limitations such as increased computational complexity or loss of interpretability

Ordinal Patterns Limitations



Our approach: PoPayI

POPAyI (Polar Ordinal Patterns with Amplitude Information) is the first work to leverage amplitude information in a multivariate OP approach.

How POPAyI works?

1. Converts 2D spatial data into symbolic patterns based on ordinal relationships between points

Our approach: PoPayI

POPAyI (Polar Ordinal Patterns with Amplitude Information) is the first work to leverage amplitude information in a multivariate OP approach.

How POPAyI works?

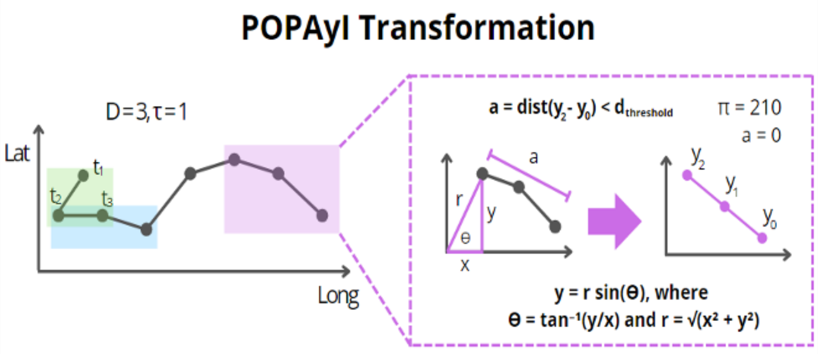
1. Converts 2D spatial data into symbolic patterns based on ordinal relationships between points
2. Uses polar coordinates to handle 2D mobility data (latitude and longitude)

POPAyI (Polar Ordinal Patterns with Amplitude Information) is the first work to leverage amplitude information in a multivariate OP approach.

How POPAyI works?

1. Converts 2D spatial data into symbolic patterns based on ordinal relationships between points
2. Uses polar coordinates to handle 2D mobility data (latitude and longitude)
3. Introduces amplitude information to distinguish variations in movement magnitude (e.g., slow vs. fast-moving vehicles)

How POPAyl works?



Methodology

A) Data Preprocessing:

Methodology

A) Data Preprocessing:

- Segmentation: Trajectories are divided into segments based on stationary points or the end of the trajectory

Methodology

A) Data Preprocessing:

- Segmentation: Trajectories are divided into segments based on stationary points or the end of the trajectory
- Data Handling: Removal of low-quality data, such as out-of-range GPS points and trajectories with fewer than 10 points.

Methodology

B) POPAyI Transformation:

Methodology

B) POPAyI Transformation:

- Ordinal Pattern Transformation: Converts time series into ordinal patterns using a sliding window.

Methodology

B) POPAyI Transformation:

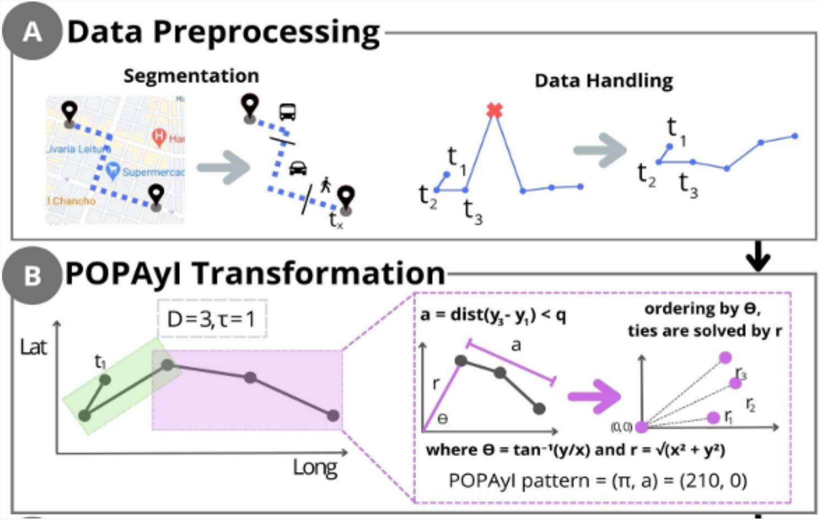
- Ordinal Pattern Transformation: Converts time series into ordinal patterns using a sliding window.
- Polar Coordinates: Patterns are based on the angle and distance of 2D points, capturing non-linear movement dynamics.

Methodology

B) POPAyI Transformation:

- Ordinal Pattern Transformation: Converts time series into ordinal patterns using a sliding window.
- Polar Coordinates: Patterns are based on the angle and distance of 2D points, capturing non-linear movement dynamics.
- Amplitude Information: Calculates the displacement (distance) between the first and last points in the window, allowing for better distinction between movement types (e.g., slow vs. fast)

Methodology



Methodology

C) Representations Derived from POPAyI:

Methodology

C) Representations Derived from POPAyI:

- Probability Distribution: counts the occurrences of each ordinal pattern

Methodology

C) Representations Derived from POPAyI:

- Probability Distribution: counts the occurrences of each ordinal pattern
  - Used to compute features like Shannon Permutation Entropy (PE), which measures the randomness of the time series, and Statistical Complexity (SC), which quantifies the level of structure in the data

Methodology

C) Representations Derived from POPAyI:

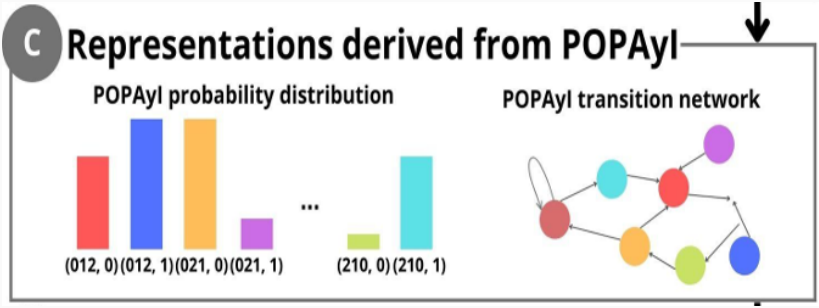
- Probability Distribution: counts the occurrences of each ordinal pattern
  - Used to compute features like Shannon Permutation Entropy (PE), which measures the randomness of the time series, and Statistical Complexity (SC), which quantifies the level of structure in the data
- Transition Network: A graph representation where nodes represent ordinal patterns, and edges represent transitions between patterns

Methodology

C) Representations Derived from POPAyI:

- Probability Distribution: counts the occurrences of each ordinal pattern
  - Used to compute features like Shannon Permutation Entropy (PE), which measures the randomness of the time series, and Statistical Complexity (SC), which quantifies the level of structure in the data
- Transition Network: A graph representation where nodes represent ordinal patterns, and edges represent transitions between patterns
  - Extract features such as Probability of Self-Transition (how likely a pattern repeats itself), Number of Nodes and Edges, and Average Edge Weights, indicating how dynamic or stable the movement is

Methodology



D) Feature Extraction:

Methodology

D) Feature Extraction:

- Features extracted from both the Probability Distribution (e.g., PE, SC) and the Transition Network (e.g., node/edge counts, self-transitions)

D) Feature Extraction:

- Features extracted from both the Probability Distribution (e.g., PE, SC) and the Transition Network (e.g., node/edge counts, self-transitions)
- Additional statistical features from the mobility data itself, such as mean, variance, max, and min values for distance and speed

Methodology

D) Feature Extraction:

- Features extracted from both the Probability Distribution (e.g., PE, SC) and the Transition Network (e.g., node/edge counts, self-transitions)
- Additional statistical features from the mobility data itself, such as mean, variance, max, and min values for distance and speed

E) Classification:

Methodology

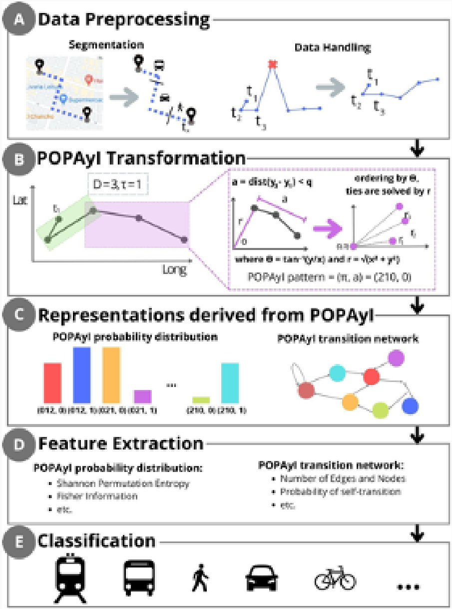
D) Feature Extraction:

- Features extracted from both the Probability Distribution (e.g., PE, SC) and the Transition Network (e.g., node/edge counts, self-transitions)
- Additional statistical features from the mobility data itself, such as mean, variance, max, and min values for distance and speed

E) Classification:

- Features are fed into an XGBoost classifier

Methodology



POPAyI advantages to TMD i

- Lightweight Approach
  - Time complexity is O(n) due to small and constant embedding dimension (D)
  - Ideal for resource-limited environments, such as IoT
- Scalability
  - Computational efficiency with linear time complexity
  - Works well with large time series and requires minimal space
- Resilience to Trajectory Size
  - Handles trajectories of varying sizes (differently from Deep Learning methods)
  - Adapts to real-world mobility patterns, regardless of trajectory length
- Robustness

POPAyI advantages to TMD ii

  - Resilient to GPS errors and dynamic noise
  - Ensures reliable detection in noisy environments.
- Enhanced Representation
  - Captures 2D temporal dynamics and incorporates amplitude.
  - Improves accuracy in distinguishing transportation modes.

Experiments

Amplitude benefits from POPAyI

Goal: Evaluate how incorporating amplitude improves detection of vehicle speed variations

Setup:

- We used Cologne dataset, which contains high-resolution vehicle trajectories with a temporal resolution of one second

Experiments

Amplitude benefits from POPAyI

Goal: Evaluate how incorporating amplitude improves detection of vehicle speed variations

Setup:

- We used Cologne dataset, which contains high-resolution vehicle trajectories with a temporal resolution of one second
- Separated into two groups: slow (2-5 m/s) and fast (6-30 m/s) vehicles

Amplitude benefits from POPAyI

Goal: Evaluate how incorporating amplitude improves detection of vehicle speed variations

Setup:

- We used Cologne dataset, which contains high-resolution vehicle trajectories with a temporal resolution of one second
- Separated into two groups: slow (2-5 m/s) and fast (6-30 m/s) vehicles
- Compare results with and without amplitude.

Experiments

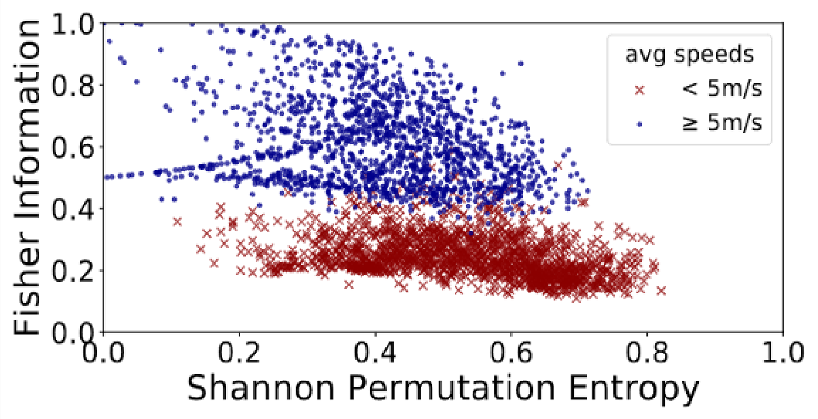


Figure 1: With amplitude: Better separation of vehicle behaviors (clear clusters)

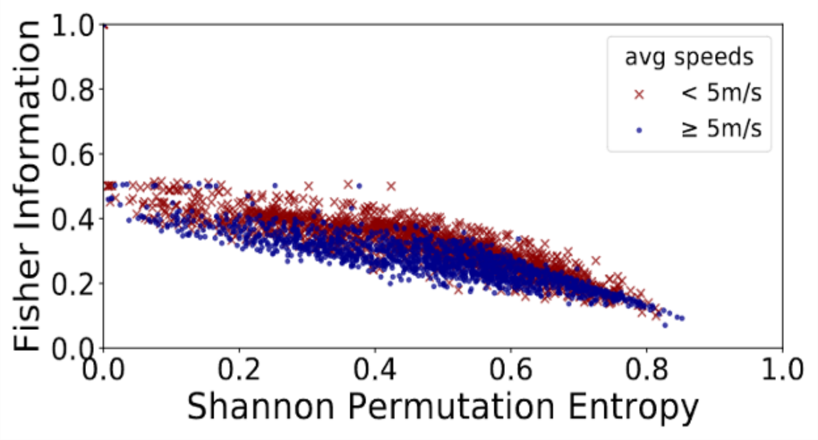


Figure 2: Without amplitude: Poor distinction between slow and fast vehicles

Conclusion: Amplitude information enhances the ability to capture mobility dynamics, improving detection accuracy.

Experiments

Transportation Mode Detection

Goals:

- Predict the transportation mode based on spatiotemporal trajectory data.
- Evaluate the effectiveness of POPAyI in classifying different transportation modes.
- Compare POPAyI to state-of-the-art Machine Learning (ML) and Deep Learning (DL) methods.

Experiments

Transportation Mode Detection

Setup:

- We used GeoLife dataset, which contains 182 users' GPS trajectories over 5 years and various transportation modes: walking, biking, car, bus, subway, train.
- Fixed-length segments of 500 points to ensure comparability across all models
- Classification using cross-validation with 5 folds and stratified classes

Hyperparameters selection

- Explored values:
  - Embedding Dimension: D = 3, 4, 5
  - Embedding Delay: $\;= 1, 2$
  - Amplitude Threshold: q = 0.0005 to 3 (0.5 m to 3 km)
- Selection Process:
  - Used Successive Halving: progressively eliminated weaker configurations based on F1-score
  - $\tau = 2$ consistently achieving the best results suggests that non-consecutive points capture more meaningful transport dynamics
  - Difficulty of establishing a straightforward correlation between hyperparameter values

Final choice: D = 3, $\;= 2$, q = 0.0005

Experiments

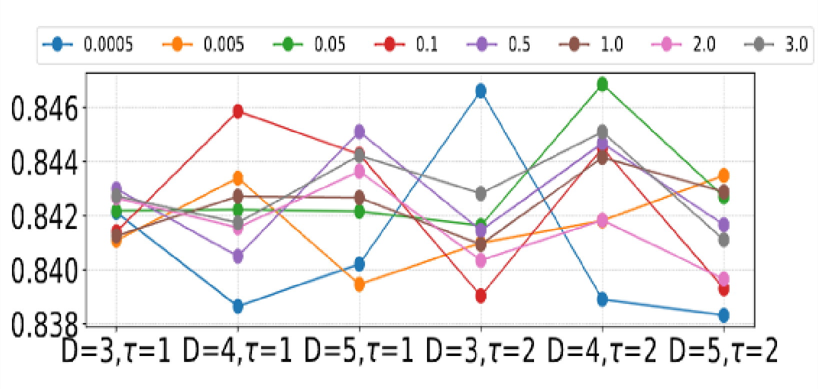


TABLE I: Comparison for different ML methods. $D = 3$, $\tau = 2$, and $q = 0.0005$.

	Accuracy	F1 score	Precision	Recall
SVM	37.36% (± 0.02)	15.62% (± 0.03)	20.35% (± 0.02)	33.82% (± 0.32)
DT	78.96% (± 0.03)	74.67% (± 0.05)	74.77% (± 0.05)	74.65% (± 0.05)
RF	86.35% (± 0.03)	82.53% (± 0.04)	81.64% (± 0.04)	83.72% (± 0.04)
XGBoost	87.12% (± 0.03)	84.02% (± 0.04)	83.28% (± 0.04)	85.90% (± 1.71)

Experiments i

Transportation Mode Detection - POPAyI vs. 1D OP and POP

1D OP:

- Separates feature extraction for latitude and longitude.
- Captures ordinal relationships in 1D space but misses non-linear movement dynamics in 2D mobility data.

POP:

- Polar Ordinal Patterns (POP) use polar coordinates to represent 2D spatial relationships.
- Better suited to capturing complex movement behaviors like turns and changes in direction.

POPAyI:

- Incorporates amplitude information into POP, capturing both direction and magnitude of movement (e.g., speed changes).

Experiments

	D	F1 score	Recall	Precision	No. of Params.
***POPAyI* q = 0.0005**	**3**	84.02% (± 0.04)	83.28% (± 0.04)	85.90% (± 1.71)	7.8×10^1
	4	84.04% (± 0.04)	83.40% (± 0.04)	84.81% (± 0.04)	1.2×10^3
	5	83.49% (± 0.04)	82.80% (± 0.04)	84.32% (± 0.04)	2.8×10^4
POP	**3**	82.24% (± 0.04)	81.23% (± 0.04)	83.47% (± 0.04)	2.1×10^1
	4	82.31% (± 0.03)	81.45% (± 0.04)	83.32% (± 0.04)	3.0×10^2
	5	81.13% (± 0.04)	79.39% (± 0.04)	81.99% (± 0.04)	7.3×10^3
1D OP lat. and long. separately	**3**	78.96% (± 0.05)	78.49% (± 0.05)	78.04% (± 0.05)	2.1×10^1
	4	78.40% (± 0.05)	77.81% (± 0.05)	79.31% (± 0.05)	3.0×10^2
	5	80.81% (± 0.03)	80.16% (± 0.05)	81.63% (± 0.04)	7.3×10^3

Experiments

TABLE III: Quantitative comparison. *POPAyI* with $D = 3$, $\tau = 2$, and $q = 0.0005$.

Transportation modes sets	Methods	Reported F1 score	Reported Accuracy	Reported Recall	Reported Precision	Data size training	No. of feats.	No. of params.
walk, bike car&taxi, bus	DT [1]	74.77%	76.20%	76.37%	76.92%	10-8000	13	-
	POPAyI	86.34% ($\pm$ 0.04)	90.21% ($\pm$ 0.05)	86.56% ($\pm$ 0.0.05)	86.16% ($\pm$ 0.06)	10-8000	19	7.8×10^1
	Light. CNN [2]	87.10% ($\pm$ 1.1)	-	-	-	500	500	1.1×10^4
	POPAyI	86.86% ($\pm$ 0.04)	89.73% ($\pm$ 0.03)	86.74% ($\pm$ 0.04)	87.02% ($\pm$ 0.04)	500	19	7.8×10^1
	POPAyI	87.41% ($\pm$ 0.04)	90.88% ($\pm$ 0.03)	87.41% ($\pm$ 0.05)	87.50% ($\pm$ 0.04)	1000	19	7.8×10^1
walk, bike, car, train bus&taxi, subway	XGBoost [3]	87.40%	90.77%	90.84%	86.46%	10-39120	111	-
	POPAyI	82.40% ($\pm$ 0.07)	86.55% ($\pm$ 0.04)	80.67% ($\pm$ 0.09)	84.57% ($\pm$ 0.07)	10-39120	19	7.8×10^1
walk, bike, car&taxi, bus, train	Best CNN [5]	74.80%	79.80%	-	-	200	200	2.6×10^6
	7 CNNs [5]	83.90%	84.80%	82.42%	86.30%	200	200	1.8×10^7
	LSTM [21]	91.90%	92.70%	91.84%	92.00%	200	200	8.1×10^6
	AE + CNN [6]	88.28%	89.47%	86.99%	89.85%	200	200	4.1×10^7
	Light. CNN [2]	83.90% ($\pm$ 1.10)	-	-	-	500	500	1.1×10^4
	POPAyI	85.68% ($\pm$ 0.03)	87.42% ($\pm$ 0.03)	85.02% ($\pm$ 0.03)	86.50% ($\pm$ 0.03)	500	19	7.8×10^1
	POPAyI	85.77% ($\pm$ 0.03)	89.48% ($\pm$ 0.05)	84.77% ($\pm$ 0.05)	86.99% ($\pm$ 0.04)	1000	19	7.8×10^1
	POPAyI	86.71% ($\pm$ 0.05)	89.76% ($\pm$ 0.03)	85.92% ($\pm$ 0.05)	87.61% ($\pm$ 0.05)	10-17000	19	7.8×10^1
walk, bike, car&taxi, bus, subway, train	Light. CNN [2]	81.80% ($\pm$ 1.10)	-	-	-	500	500	1.1×10^4
	POPAyI	84.02% ($\pm$ 0.04)	87.12% ($\pm$ 0.04)	83.28% ($\pm$ 0.04)	85.90% ($\pm$ 1.71)	500	19	7.8×10^1
	POPAyI	85.77% ($\pm$ 0.03)	89.48% ($\pm$ 0.04)	84.77% ($\pm$ 0.05)	86.99% ($\pm$ 0.05)	1000	19	7.8×10^1

Confusion Matrix

- Imbalanced classes might significantly impact classification

Confusion Matrix

- Imbalanced classes might significantly impact classification
- Certain transportation modes may misclassify due to shared traffic characteristics and similar temporal dynamics

Confusion Matrix

- Imbalanced classes might significantly impact classification
- Certain transportation modes may misclassify due to shared traffic characteristics and similar temporal dynamics
- Classifications are clearer between motor and non-motor transports due to distinct speed and distance behaviors

Confusion Matrix

- Imbalanced classes might significantly impact classification
- Certain transportation modes may misclassify due to shared traffic characteristics and similar temporal dynamics
- Classifications are clearer between motor and non-motor transports due to distinct speed and distance behaviors
- Some misclassifications result from data quality issues

Confusion Matrix

- Imbalanced classes might significantly impact classification
- Certain transportation modes may misclassify due to shared traffic characteristics and similar temporal dynamics
- Classifications are clearer between motor and non-motor transports due to distinct speed and distance behaviors
- Some misclassifications result from data quality issues
- Impact of False Positives/Negatives varies across ITS applications

Confusion Matrix

- Imbalanced classes might significantly impact classification
- Certain transportation modes may misclassify due to shared traffic characteristics and similar temporal dynamics
- Classifications are clearer between motor and non-motor transports due to distinct speed and distance behaviors
- Some misclassifications result from data quality issues
- Impact of False Positives/Negatives varies across ITS applications
  - Acceptable mixing of transport types in urban planning.

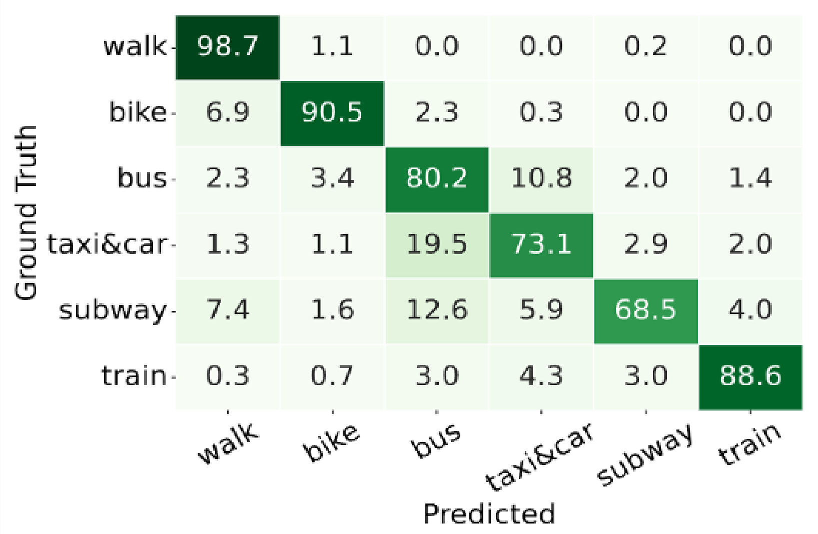
Confusion Matrix

- Imbalanced classes might significantly impact classification
- Certain transportation modes may misclassify due to shared traffic characteristics and similar temporal dynamics
- Classifications are clearer between motor and non-motor transports due to distinct speed and distance behaviors
- Some misclassifications result from data quality issues
- Impact of False Positives/Negatives varies across ITS applications
    - Acceptable mixing of transport types in urban planning.
    - Critical applications require minimal false negatives for safety.

Confusion Matrix

- Imbalanced classes might significantly impact classification
- Certain transportation modes may misclassify due to shared traffic characteristics and similar temporal dynamics
- Classifications are clearer between motor and non-motor transports due to distinct speed and distance behaviors
- Some misclassifications result from data quality issues
- Impact of False Positives/Negatives varies across ITS applications
    - Acceptable mixing of transport types in urban planning.
    - Critical applications require minimal false negatives for safety.
    - POPAyI Advantage: Fewer false negatives between motorized and non-motorized transports enhance safety in critical applications

Confusion Matrix



Conclusions

- POPAyI demonstrates effective TMD by leveraging multivariate ordinal patterns and amplitude information

Conclusions

- POPAyI demonstrates effective TMD by leveraging multivariate ordinal patterns and amplitude information
- Achieves high accuracy with minimal computational resources, making it suitable for resource-constrained environments (e.g., IoT)

Conclusions

- POPAyI demonstrates effective TMD by leveraging multivariate ordinal patterns and amplitude information
- Achieves high accuracy with minimal computational resources, making it suitable for resource-constrained environments (e.g., IoT)
- Outperforms traditional methods and DL approaches while maintaining a lightweight framework

Future Directions

- Expand Dataset Applications: Test POPAyI on more diverse mobility datasets to validate generalizability.

Future Directions

- Expand Dataset Applications: Test POPAyI on more diverse mobility datasets to validate generalizability.
- Enhance Real-Time Adaptability: Improve processing algorithms for better performance in real-time applications.

Future Directions

- Expand Dataset Applications: Test POPAyI on more diverse mobility datasets to validate generalizability.
- Enhance Real-Time Adaptability: Improve processing algorithms for better performance in real-time applications.
- Integrate with Smart City Infrastructure: Explore the use of POPAyI in smart city projects for traffic management and urban planning.

Future Directions

- Expand Dataset Applications: Test POPAyI on more diverse mobility datasets to validate generalizability.
- Enhance Real-Time Adaptability: Improve processing algorithms for better performance in real-time applications.
- Integrate with Smart City Infrastructure: Explore the use of POPAyI in smart city projects for traffic management and urban planning.
- Address Misclassifications: Investigate methods to further reduce misclassifications, particularly between similar transportation modes.

Future Directions

- Expand Dataset Applications: Test POPAyI on more diverse mobility datasets to validate generalizability.
- Enhance Real-Time Adaptability: Improve processing algorithms for better performance in real-time applications.
- Integrate with Smart City Infrastructure: Explore the use of POPAyI in smart city projects for traffic management and urban planning.
- Address Misclassifications: Investigate methods to further reduce misclassifications, particularly between similar transportation modes.
- Incorporate Additional Features: Enhance the model by integrating more features from mobility data, such as environmental factors affecting transport dynamics.