

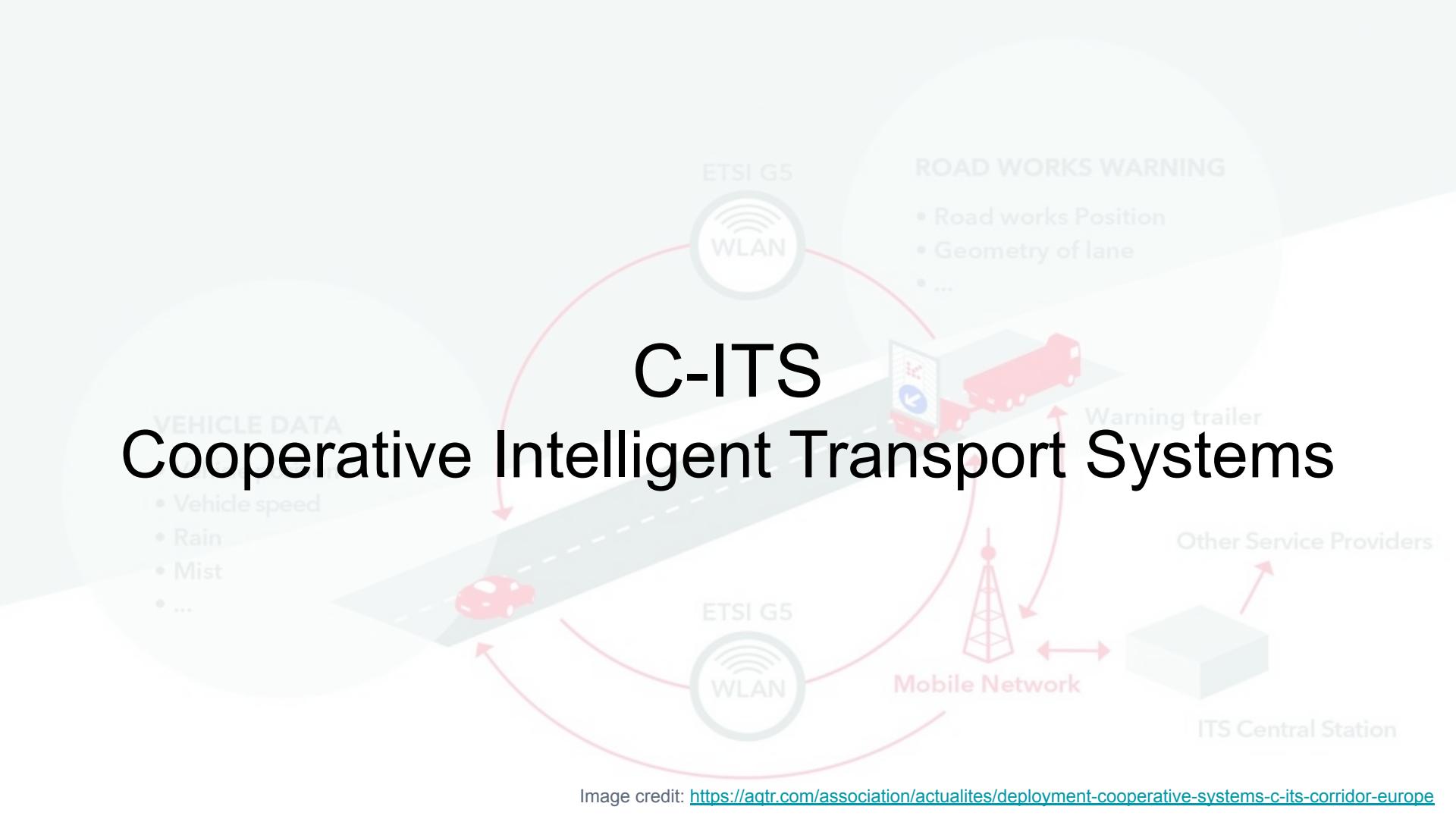
Traitement des données : Application aux C-ITS

Télécommunication
Réseaux, Sécurité et Objets Connectés
ENSEIRB-MATMECA

Secil Ercan
sercan@bordeaux-inp.fr
https://github.com/ercanse/DataProcessing_CITS

C-ITS

Cooperative Intelligent Transport Systems

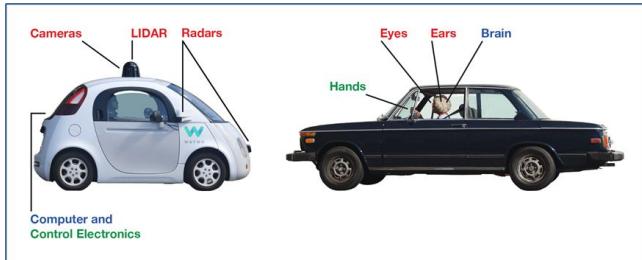


Main Objectives in ITS

- Road safety
- Traffic efficiency
- Comfort
- Sustainability

Moreover in C-ITS:

- Being prepared for future challenges in automation



ITS

WITHOUT C-ITS



Real Time

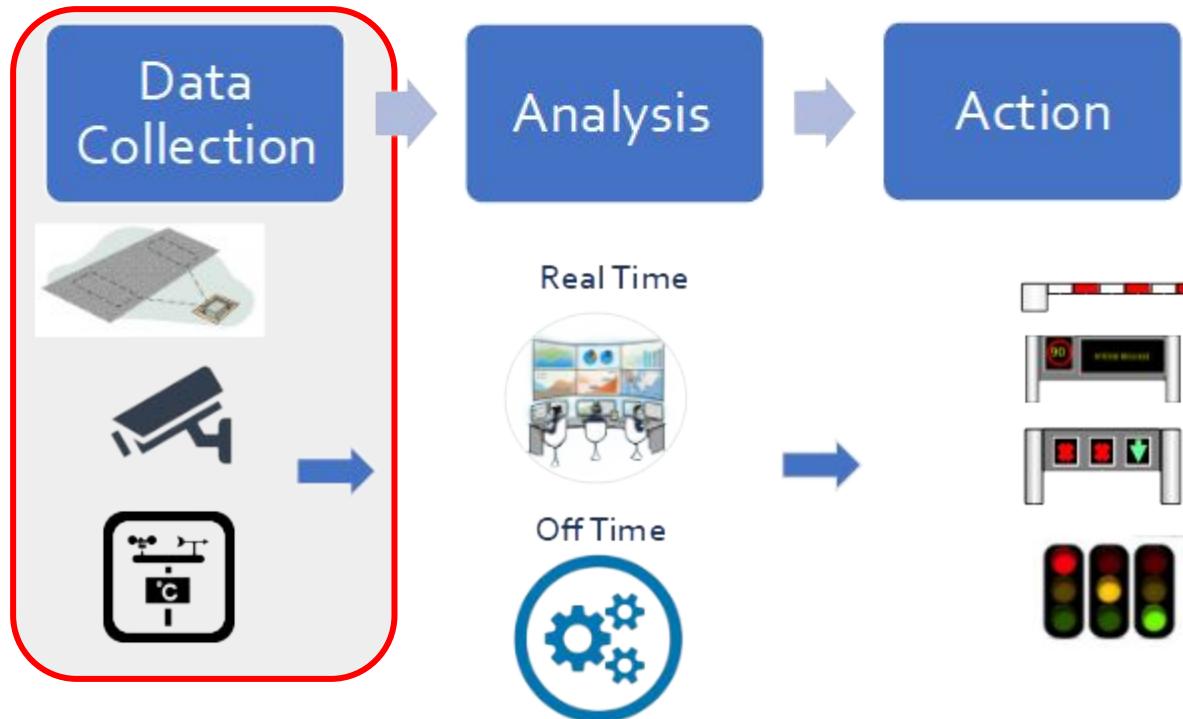


Off Time

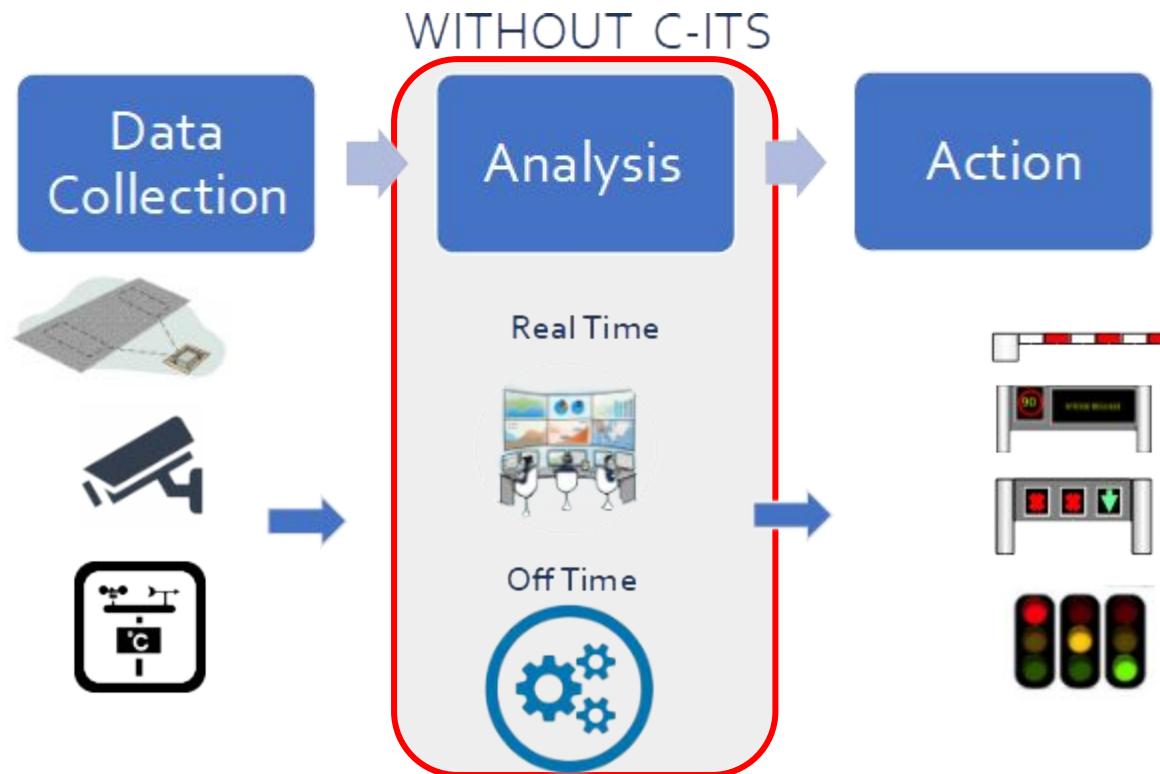


ITS

WITHOUT C-ITS

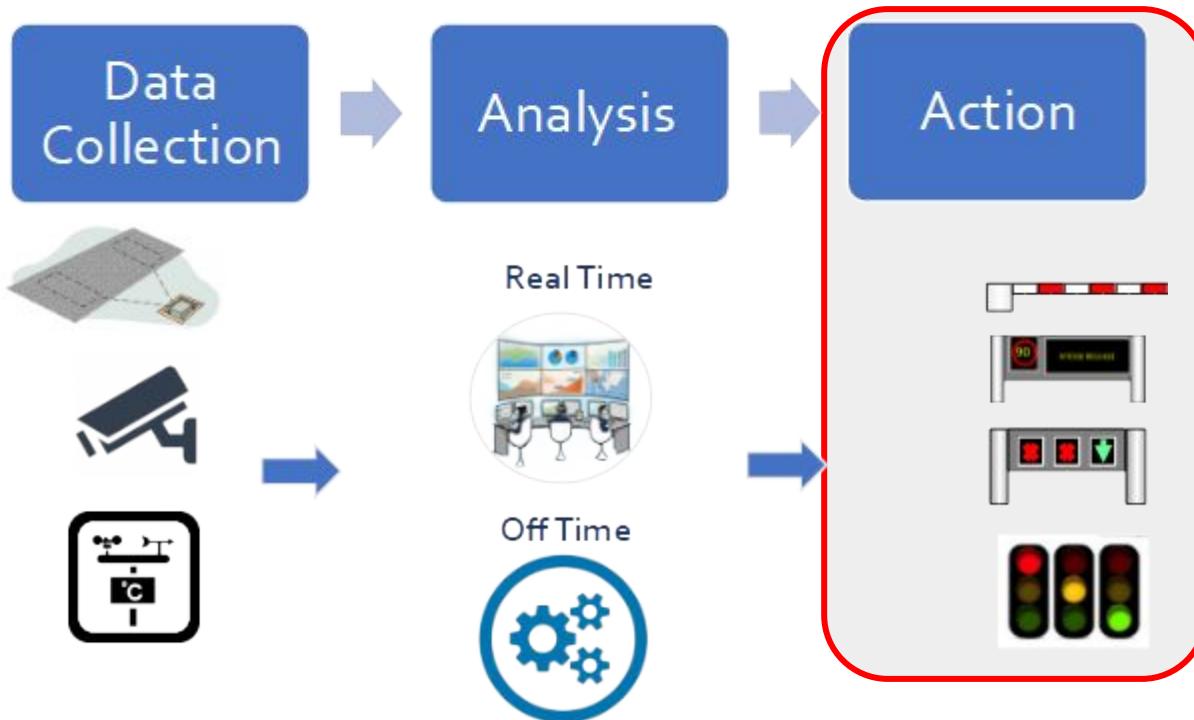


ITS



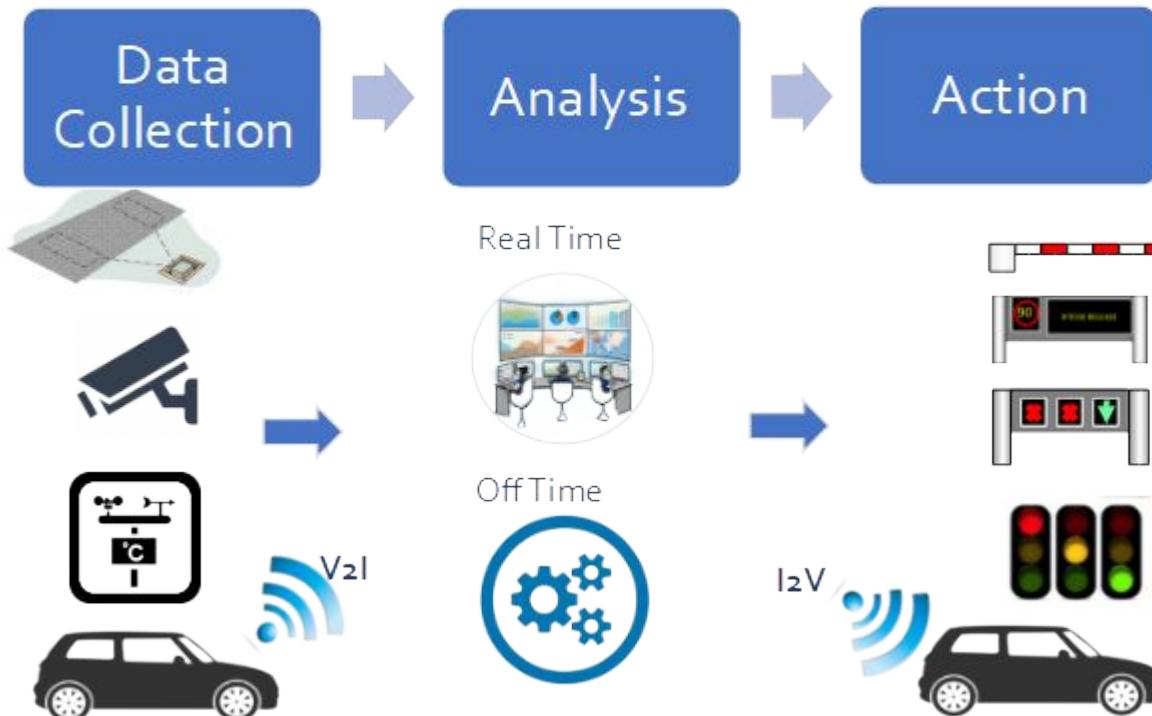
ITS

WITHOUT C-ITS

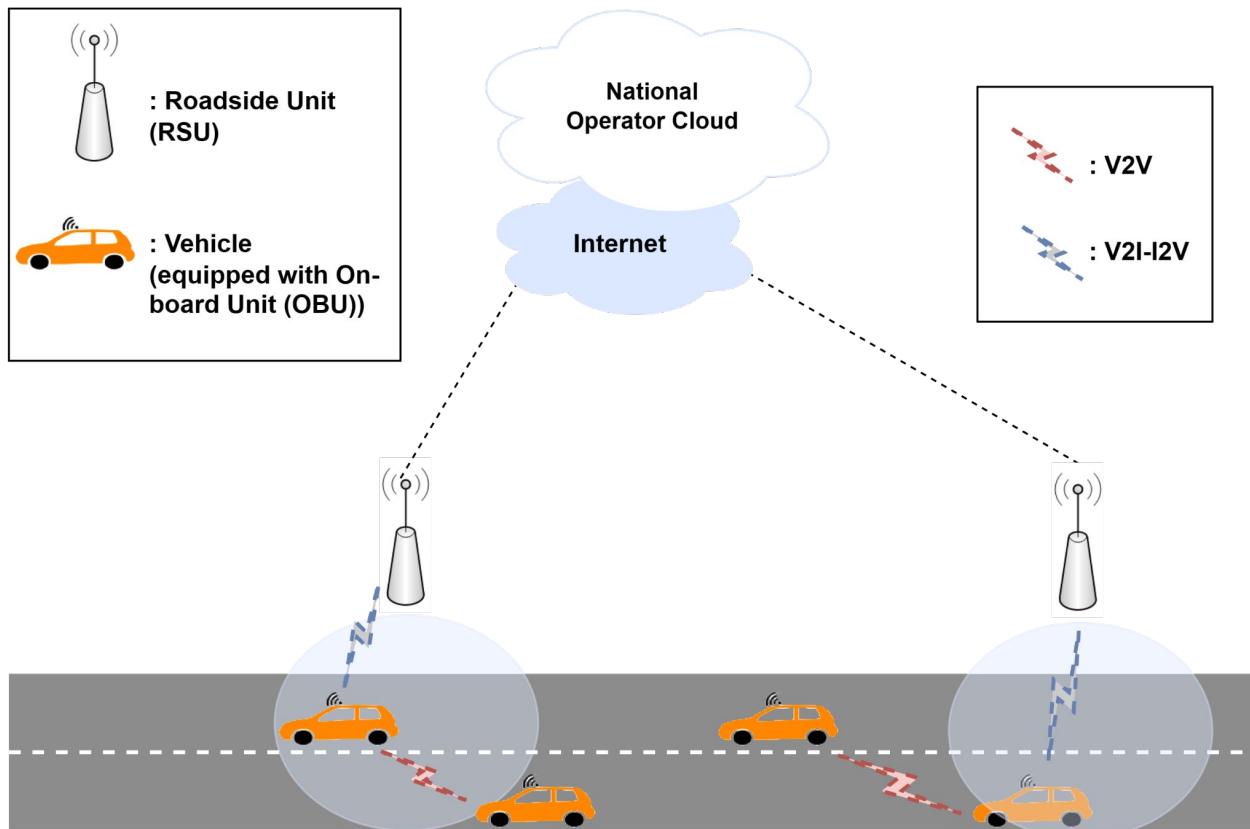


C-ITS

WITH C-ITS



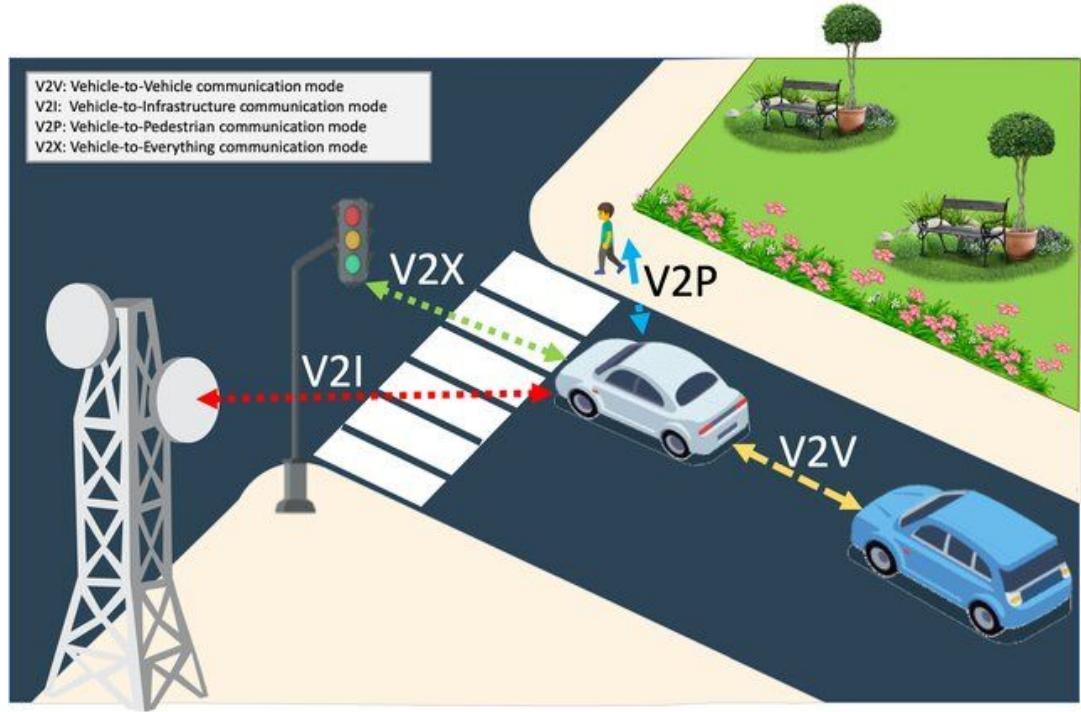
C-ITS



C-ITS

Different communication modes:

- V2V (Vehicle-to-Vehicle)
- V2I (Vehicle-to-Infrastructure)
- I2V (Infrastructure-to-Vehicle)
- V2P (Vehicle-to-Pedestrian)
- V2X (Vehicle-to-Everything), etc.



C-ITS

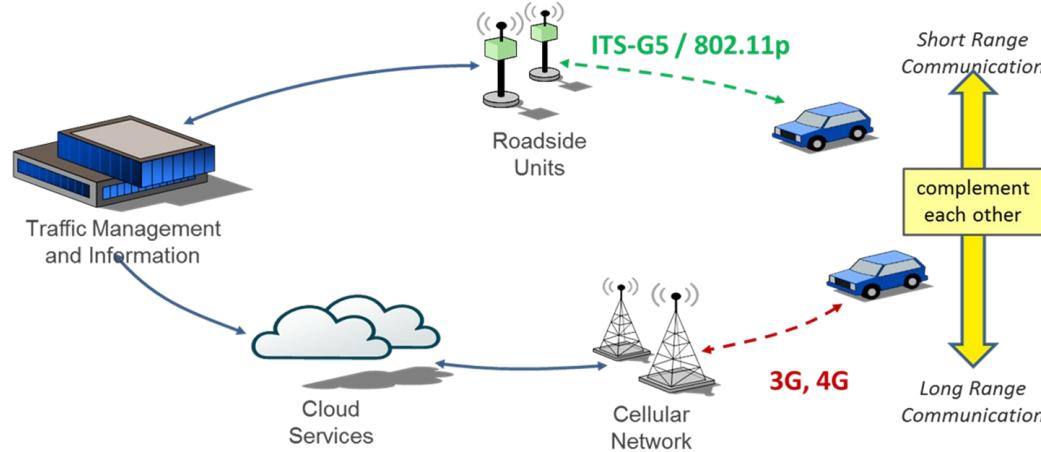
Different types of messages:

- CAM (Cooperative Awareness Message)
- DENM (Decentralized Environmental Notification Message)
- IVI (In-Vehicle Information)
- CPM (Collective Perception Message)
- TLM (Traffic Light Maneuver), etc.

C-ITS

Different Radio Access Technologies (RAT):

- ITS-G5 (IEEE 802.11p),
- Cellular (3G/4G/5G, etc.), etc.



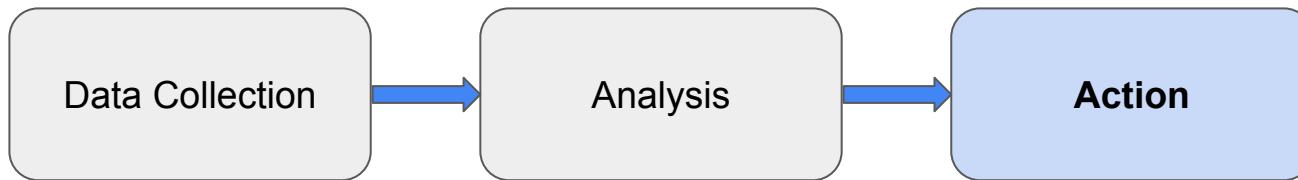
C-ITS



Examples of services for data collection - V2I

- Basic PVD (Probe Vehicle Data): Speed, position, heading, etc.
- Extended PVD: Information on road events, etc.

C-ITS



Examples of services for action - I2V

- GLOSA (Green Light Optimal Speed Advisory)
- RWW (Road Works Warning)
- IVSL (In vehicle speed limit)
- IVS (In vehicle signage - virtual DMS)

C-ITS

Example:

RWW (Road Works
Warning) message



Coopits - Interactive Map

<https://coopits.developpement-durable.gouv.fr/carte-interactive-des-services/>

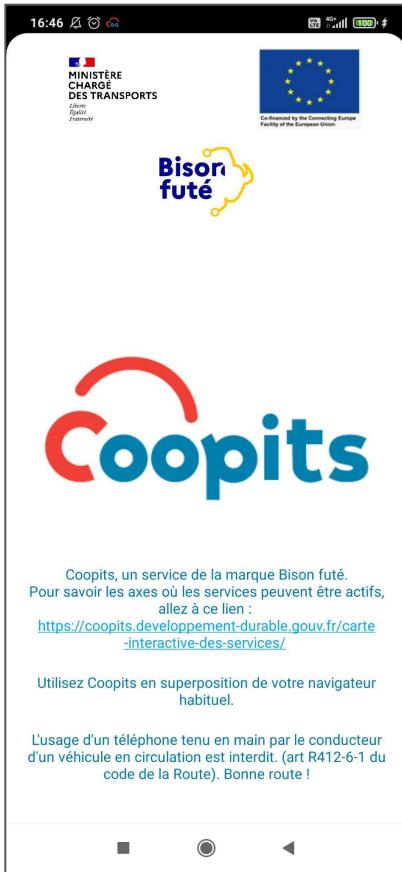
Different actors

Different services

“Coopits est une application innovante et sécurisante qui s'appuie sur les outils numériques dernière génération pour vous aider à mieux conduire : aide à la décision, écoconduite, etc.”



Coopits - Mobile Application



C-ITS Evaluation

- Latency
- Message size
- Message volume (mean by hour/day/week/month/weekend)
- Events triggered by smartphones or road operators
- Validity duration of event messages
- Message reception
- Message display
- Speed limit
- Slow down/stop after displaying an event
- Maneuver analysis

C-ITS Evaluation

CAM

log_timestamp
log_stationid
position_latitude
position_longitude
speedvalue
headingvalue

DENM

log_timestamp
log_stationid
originatingstationid
sequencenumber
referencetime
event_latitude
event_longitude
causecode

DENMEVENT

log_timestamp
log_stationid
originatingstationid
sequencenumber
eventid
referencetime
causecode

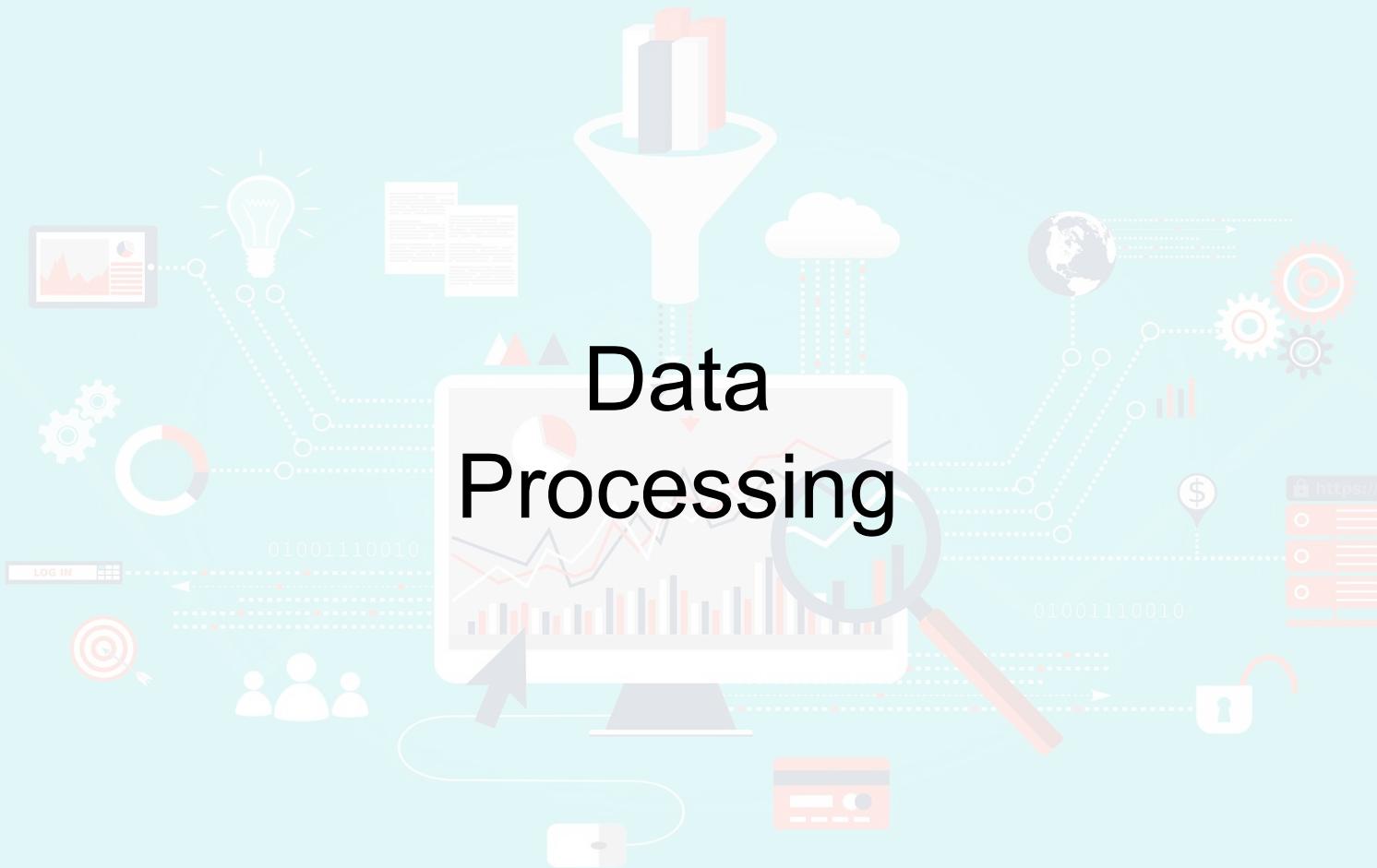
DENMACTION

log_timestamp
log_stationid
eventid
eventmodelid
eventactionid

C-ITS Problems

- Traffic prediction
- Intrusion detection
- Driver behavior detection
- Accident prediction/detection
- Obstacle detection
- Vehicle movement prediction
- Routing protocol
- Advanced toll system
- Smart parking
- Spectrum sensing
- Energy efficient
- Lane changing assessment
- Entertainment
- Advertising
- etc.

Data Processing



Data Types

- Unstructured data
- Semi-structured data
- Structured data

Unstructured data

The university has 5600 students.
John's ID is number 1, he is 18 years old and already holds a B.Sc. degree.
David's ID is number 2, he is 31 years old and holds a Ph.D. degree. Robert's ID is number 3, he is 51 years old and also holds the same degree as David, a Ph.D. degree.

Semi-structured data

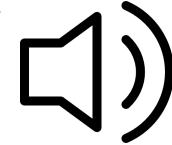
```
<University>
  <Student ID="1">
    <Name>John</Name>
    <Age>18</Age>
    <Degree>B.Sc.</Degree>
  </Student>
  <Student ID="2">
    <Name>David</Name>
    <Age>31</Age>
    <Degree>Ph.D. </Degree>
  </Student>
  ...
</University>
```

Structured data

ID	Name	Age	Degree
1	John	18	B.Sc.
2	David	31	Ph.D.
3	Robert	51	Ph.D.
4	Rick	26	M.Sc.
5	Michael	19	B.Sc.

- Numeric data
- Video/Image data
- Text data
- Audio data
- etc.

12	780,8	178
567	110,6	109
22	120,5	120
125	143,6	107



Data Processing Steps

- Collection
 - JSON, CSV, XML, PNG, etc.
- Preparation
- Input
- Processing
- Output
 - Performance indicators
 - Display results
- Storage

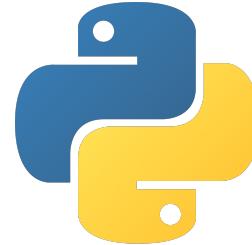


Data Processing Methods

- Single user programming
- Multiple programming
- Real-time processing
- On-line processing
- Time sharing processing
- Distributed processing
- etc.

Data Processing Tools

- Python - Pandas/ Sklearn/ TensorFlow/ Keras
 - Spark
 - Hadoop
 - etc.
- Anaconda
 - Jupyter notebook
 - Python



Data in C-ITS

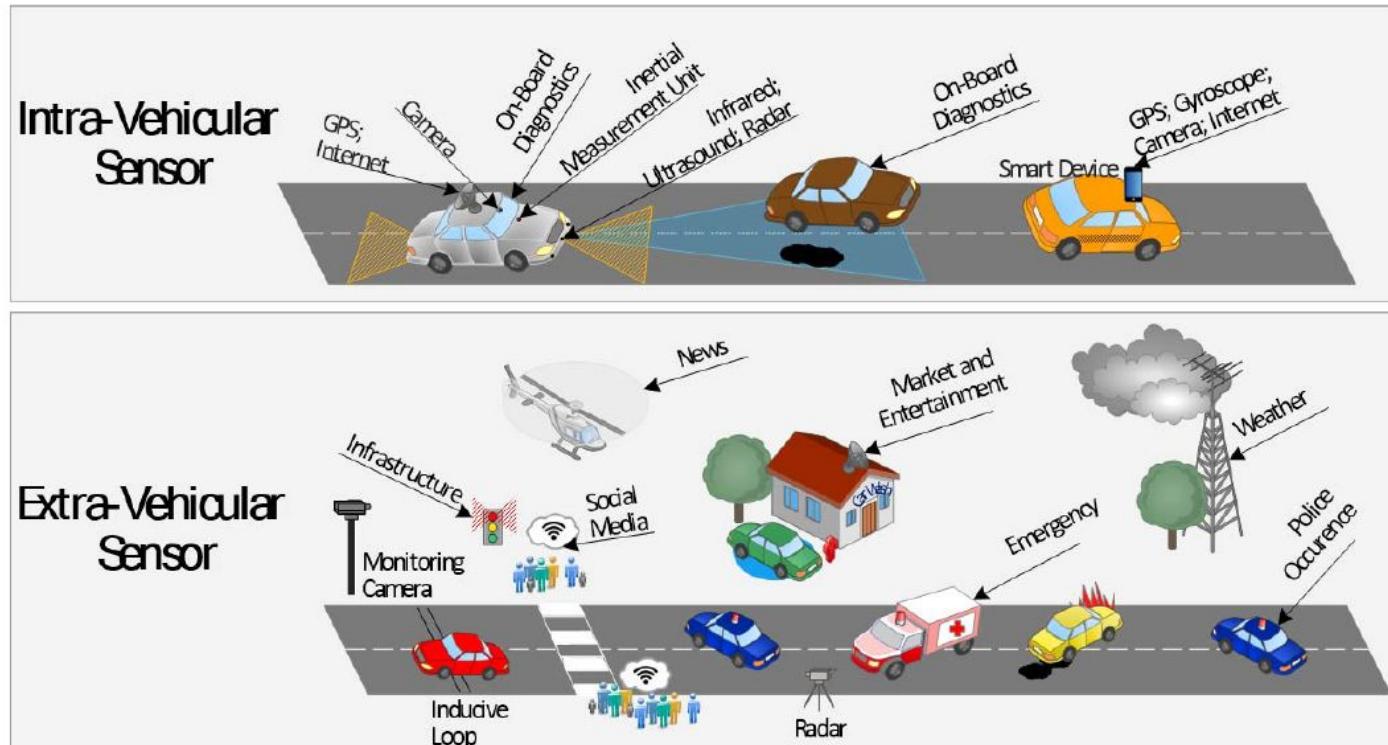


Image credit: P. H. Rettore, G. Maia, L. A. Villas and A. A. F. Loureiro, "Vehicular Data Space: The Data Point of View," in *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2392-2418, thirdquarter 2019, doi: 10.1109/COMST.2019.2911906.

Data collection in C-ITS

Sensors:

- LiDARs
- Cameras
- Radars
- Onboard sensors
- etc.

Collection of sensor data:

- from vehicles
- from infrastructures

Data collection in C-ITS

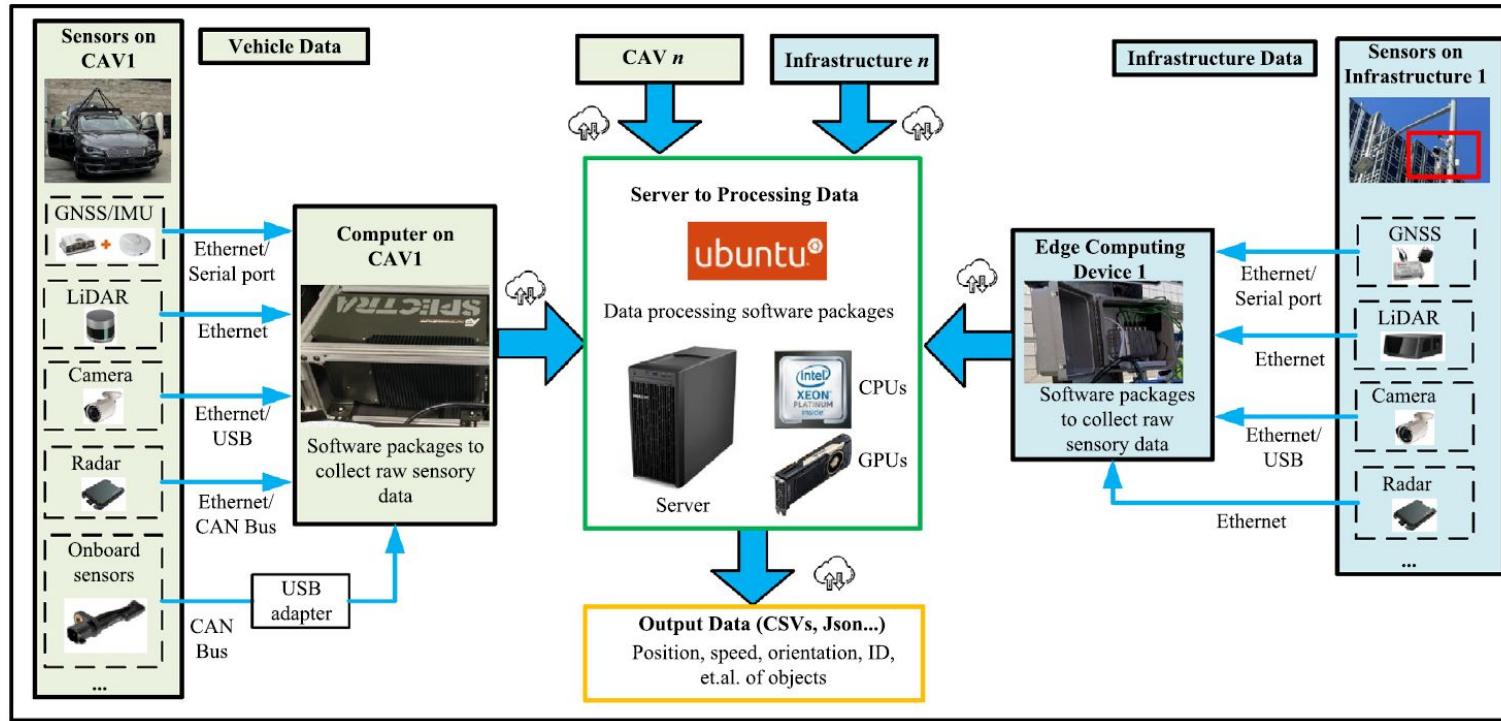


Image credit: Tak S., Lee J.-D., Song J., Kim S. *Development of AI-based vehicle detection and tracking system for C-ITS application*, J. Adv. Transp. (2021)

Data preparation in C-ITS

Processing of sensor data (data including position, speed, ID, etc. through CSVs or JSONs)

Example: Onboard sensors → Communication logs

ASN1DATA:

"0202F069BCBDE08B4059F3DFB24DB431F72000000003BD1E00924FC0007EBFE9ED0737FEEBFFF600014200007FFFF1CE201C180F1B1CE1FDA78155B1CE1FD648140B1CE1FD00816A71CE1FC7B818371CE1FC5A818BB1CE1FA43810371CE1FEE57E3471CE1FF267E2931CE1FE6F7E4A71CE1FC5A7EC331CE1FB607EFDB1CE1FC277EF131CE1FD757EE471CE1FED37EBF31CE1FFBD7E1871CE200F97DE231CE202467E3171CE202BB7E84F1CE202AA7E8CF1CE202477EC371CE2030E7E7C71CE202AA7EA5F1CE2031F7E6FF1CE2030F7E88F1CE203517E67B1CE202DC7E8D31CE2030E7E9DB1CE202367EECF1CE202577EAE71CE202577E6371CE2029A7EA1F1CE202577EF531CE202997EE8F1CE202AB7EBAF1CE202357EC771CE202897EC771CE202257ECFB1CE2029A7E6FF1CE0"

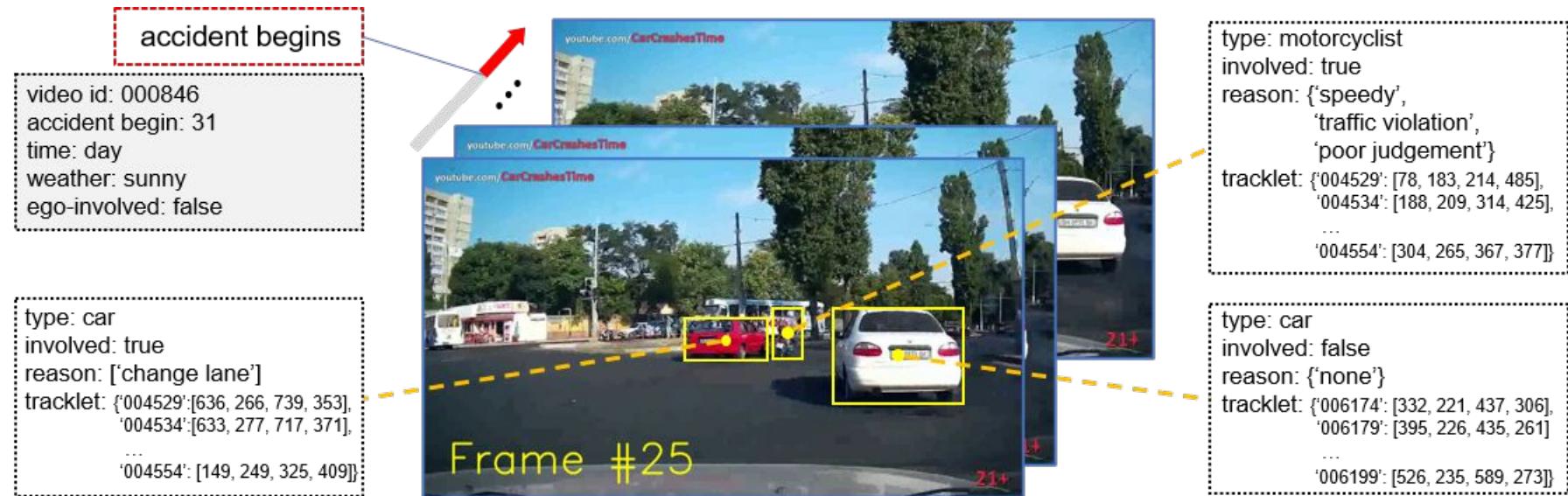
Fields in .csv:

protocolversion	messageid	stationid	generationdelatime	stationtype	referencetposition_postgis	referencetposition_latitude	referencetposition_longitude	referencetposition_semmajorconfidence	referencetposition_seminorconfidence
					geography	double precision	double precision	double precision	double precision
2	2	4033461437	57483	5	0101000020E610000A1061DBE96710F40A22EADE17DCA4540	43.5819666	3.9304633	0	
referencetposition_semmajororientation	referencetposition_altitudevalue	referencetposition_altitudeconfidence	headingvalue	headingconfidence	speedvalue	speedconfidence	drivedirection	vehiclelengthvalue	vehiclelengthconfidenceindication
		altitudeconfidence	double precision	double precision	double precision	double precision	drivedirection	double precision	vehiclelengthconfidenceindication
0	5960	unavailable	2340	[null]	0	[null]	unavailable	[null]	unavailable
referencetposition_semmajoracceleration	referencetposition_accelerationvalue	referencetposition_accelerationconfidence	longitudinalaccelerationvalue	longitudinalaccelerationconfidence	longitudinalaccelerationvalue	longitudinalaccelerationconfidence	longitudinalaccelerationvalue	longitudinalaccelerationconfidence	longitudinalaccelerationvalue
		referencetposition_accelerationconfidence	double precision	double precision	double precision	double precision	double precision	double precision	double precision
0	5960	unavailable	2340	[null]	0	[null]	unavailable	[null]	unavailable

Data preparation in C-ITS

Processing of sensor data (data including position, speed, ID, etc. through CSVs or JSONs)

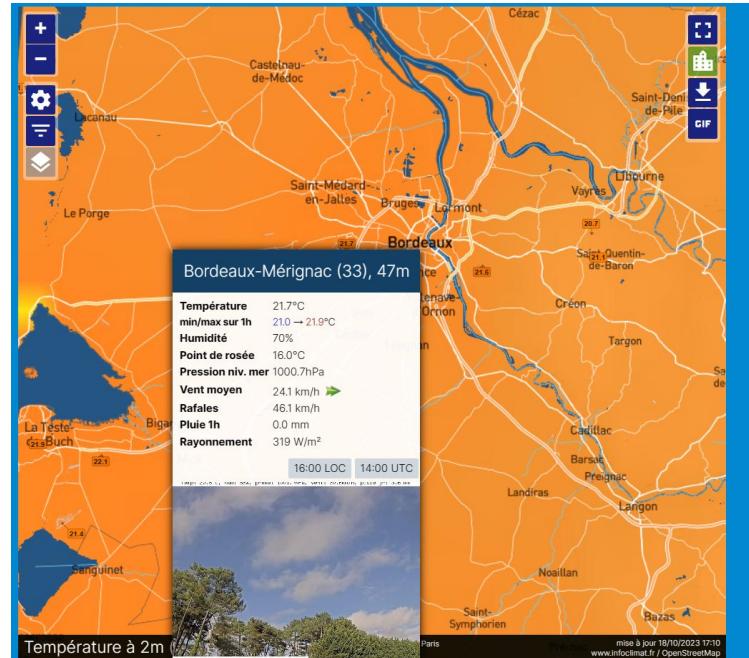
Example: Cameras → Videos



Data preparation in C-ITS

Processing of sensor data (data including position, speed, ID, etc. through CSVs or JSONs)

Example: Meteorological sensors



Cartes en temps réel des stations météorologiques mondiales, available at

<https://www.data.gouv.fr/fr/reuses/cartes-en-temps-reel-des-stations-meteorologiques-mondiales/>

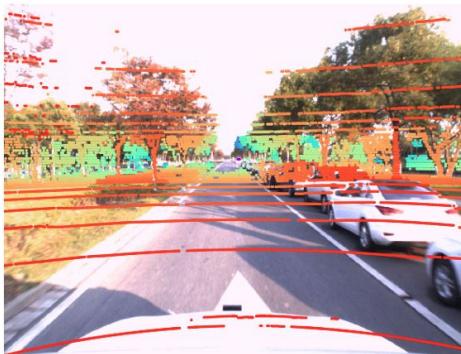
Data preparation in C-ITS

Processing of sensor data (data including position, speed, ID, etc. through CSVs or JSONs)

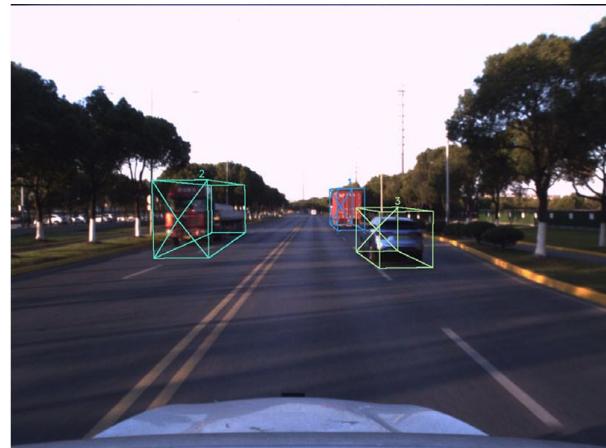
Example: Radars, Lidars



(a) 4D radar point clouds



(b) Lidar point clouds

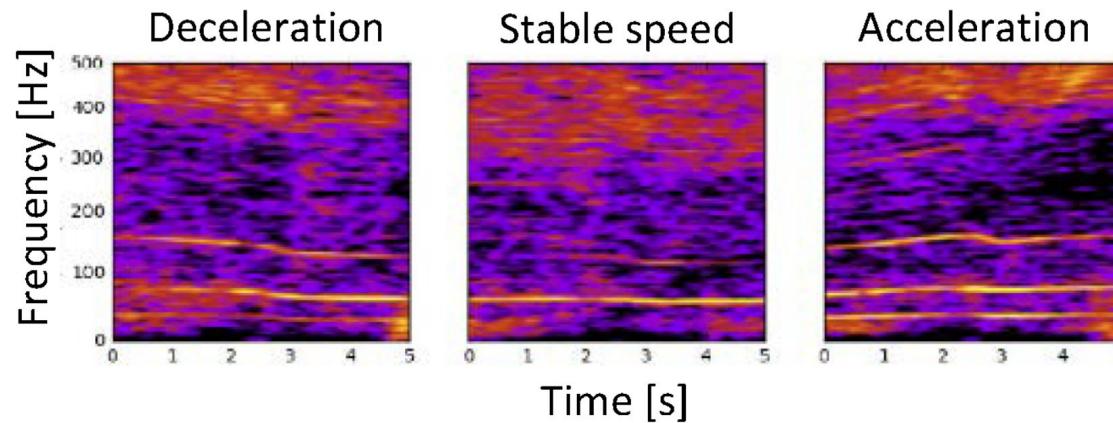


3D bounding box of each object includes the center point (x,y,z), length, width, height (l,w,h), and orientation angle (yaw)

Data preparation in C-ITS

Processing of sensor data (data including position, speed, ID, etc. through CSVs or JSONs)

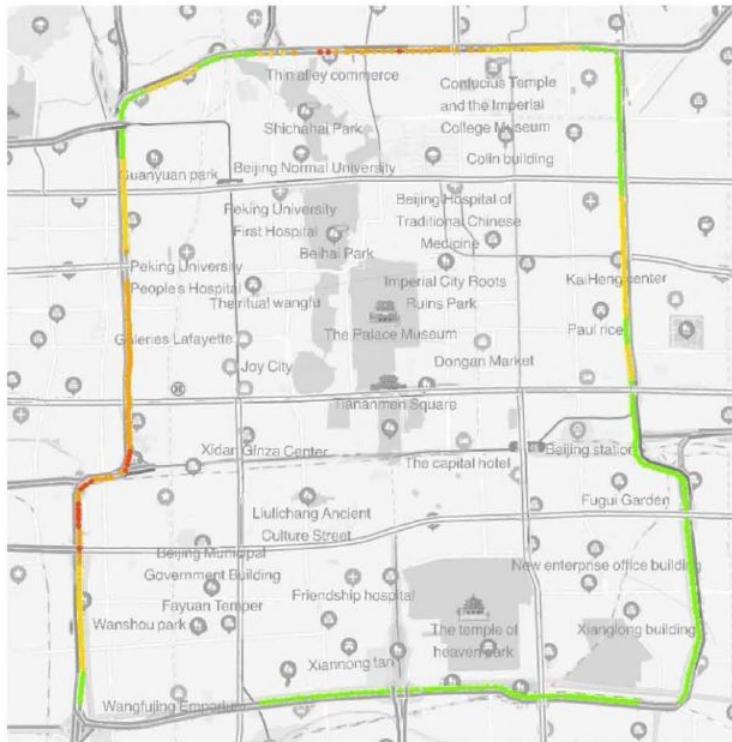
Example: Audio sensors → Spectrogram



Zero Crossings Rising, Audio Spectrum Envelope (SE), SumSE, MaxSE, IndMaxSE (index of the SE maximum), Audio Spectrum Flatness, Energy (energy of the entire spectrum), Audio Spectrum Centroid, Audio Spectrum Spread

C-ITS Problems

Traffic prediction



400 points

Between 7:00-20:00

During 1 month

Pekin/China

Features: month, day, weekday, hour, minute, and index

Traffic density classes:

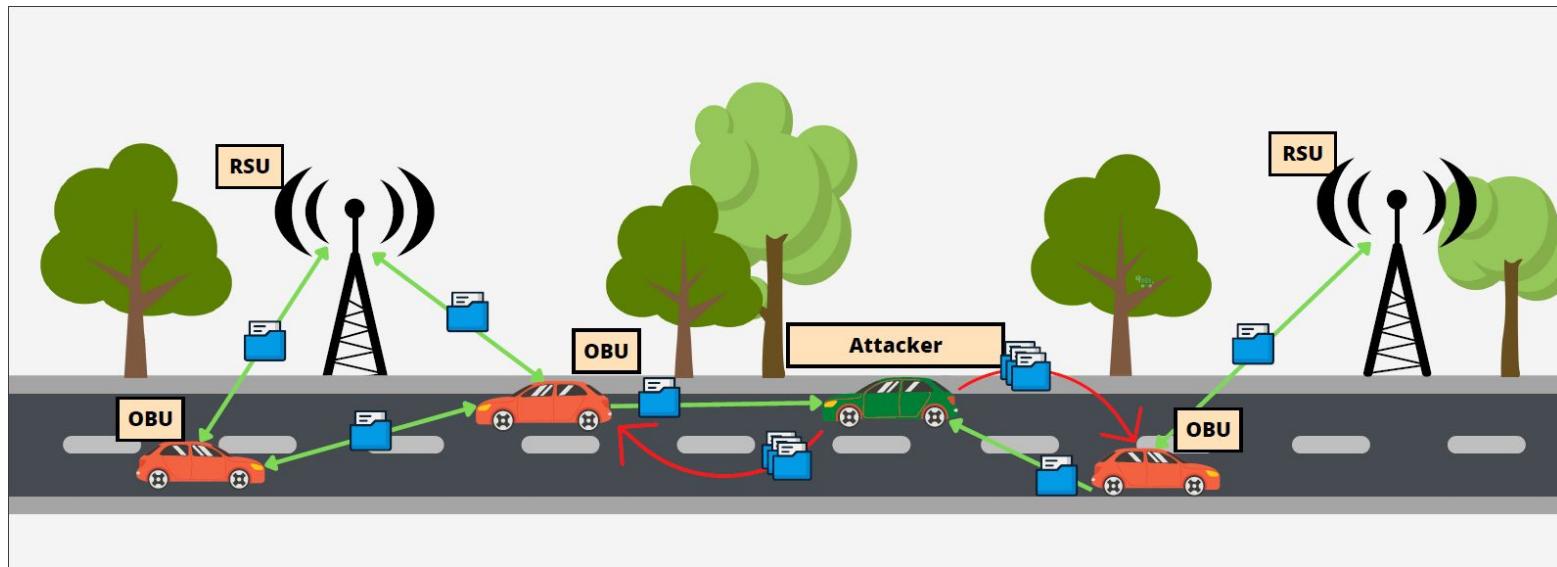
- Unimpeded
- Slow
- Congested
- Seriously congested

C-ITS Problems

Intrusion detection

Different attack types:

- DoS (denial-of-service)
- DoS Random
- DoS Disruptive
- DoS Random Sybil
- DoS Disruptive Sybil



C-ITS Problems

Driver behavior detection



Class: **Talk Left**
Modality: **RGB**
Driver ID: **P007**
Camera ID: **C1**
Vehicle ID: **V3**
Scene ID: **S3**



Class: **Eat Right**
Modality: **RGB**
Driver ID: **P008**
Camera ID: **C4**
Vehicle ID: **V1**
Scene ID: **S2**



Class: **Sleep**
Modality: **NIR**
Driver ID: **P022**
Camera ID: **C2**
Vehicle ID: **V1**
Scene ID: **S2**



Class: **Text Right**
Modality: **NIR**
Driver ID: **P038**
Camera ID: **C3**
Vehicle ID: **V1**
Scene ID: **S1**

4 cameras for each driver
(front-left, front, front-right,
and side-right)

> 470K images



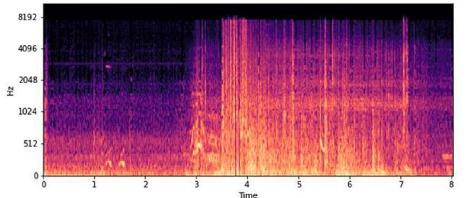
No.	Behavior
1	Normal driving
2	Sleeping
3	Yawning
4	Talk with cellphone (left)
5	Talk with cellphone (right)
6	Texting (left)
7	Texting (right)
8	Hair / makeup
9	Looking left
10	Looking right
11	Looking up
12	Looking down
13	Smoking (left)
14	Smoking (right)
15	Smoking (mouth)
16	Drinking / Eating (left)
17	Drinking / Eating (right)
18	Adjusting radio
19	Operating GPS / entertainment system
20	Reaching behind
21	Hands off the steering wheel
22	Talking to passengers

C-ITS Problems

Accident detection

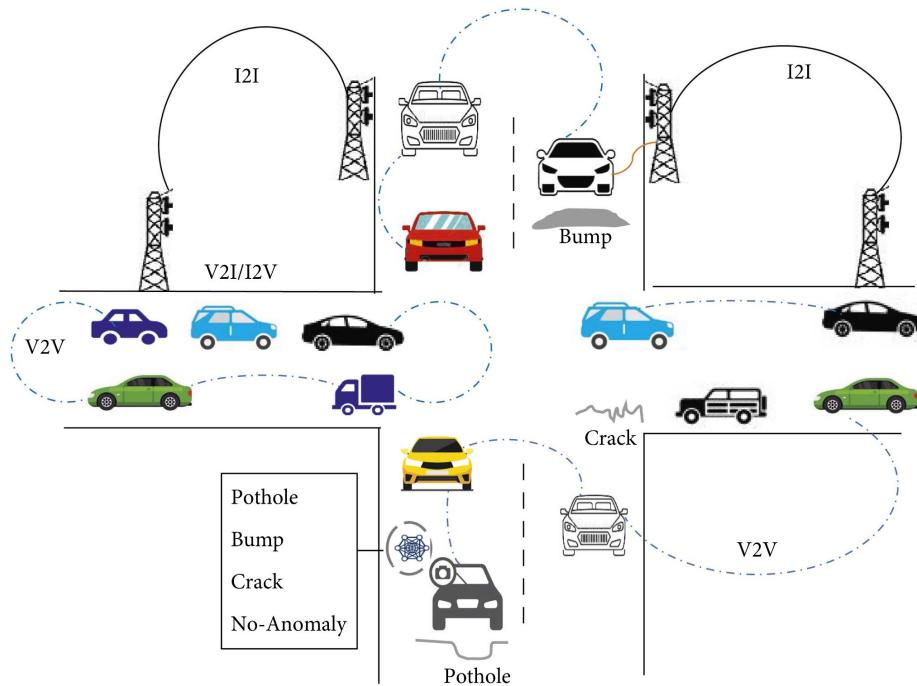
Video data
Audio data

- audio features (energy, entropy, spectral flux, etc.)
- spectrogram images



C-ITS Problems

Obstacle detection



Obstacles:

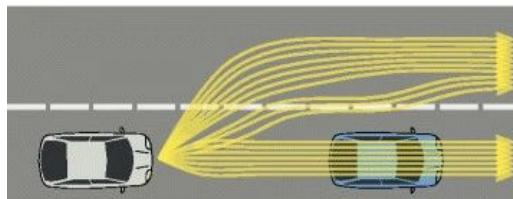
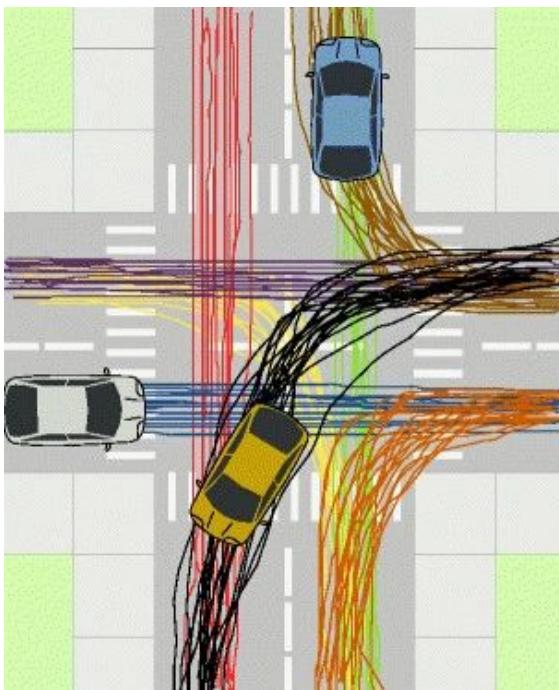
- Bump
- Pothole
- Crack

Images:

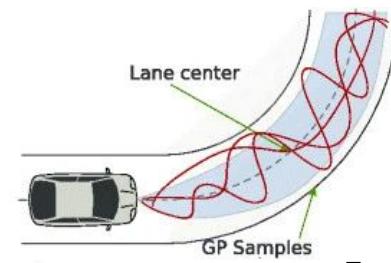


C-ITS Problems

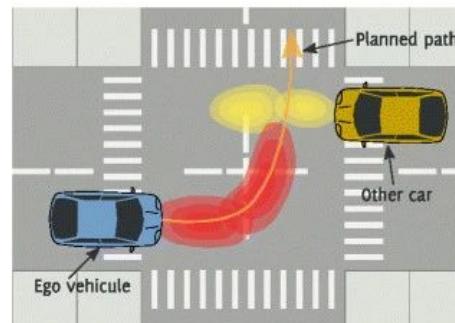
Vehicle movement prediction



a



b



c

Maneuver classes:

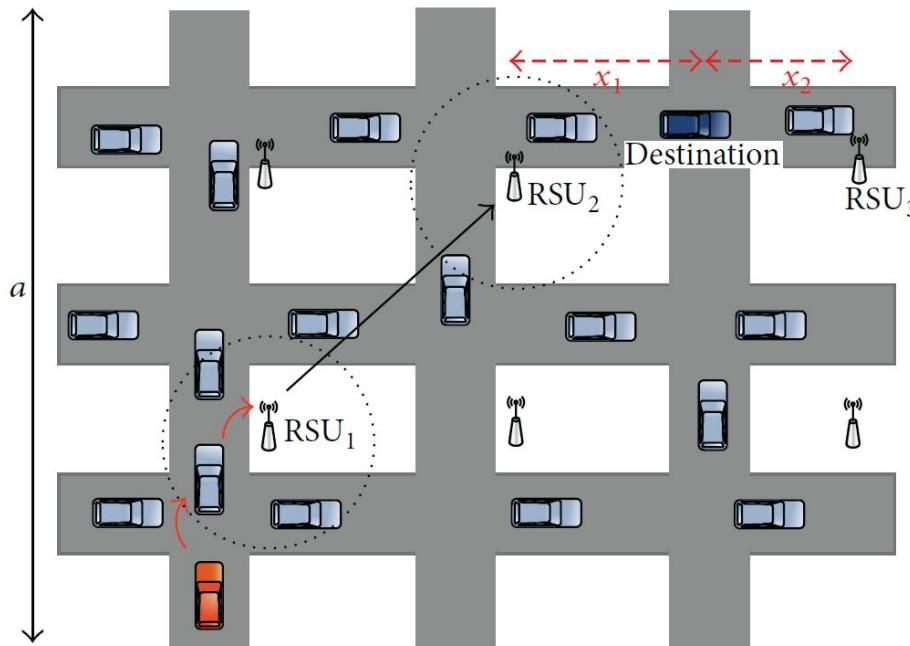
- Go straight
- Left turn
- Right turn
- Stop

Features:

- distance to intersection, position,
- speed,
- longitudinal acceleration,
- distance to lane center,
- steering angle, orientation,
- road curvature,
- turn signal

C-ITS Problems

Routing protocol



Prediction mechanism of vehicle location

Classes

- go straight,
- turn left,
- turn right

Features:

- lane number,
- direction,
- speed,
- traveling roads

Evaluation mechanism of transmission capacity (collision probability)

Evaluation mechanism of forwarding direction

Application

Installation

- For ubuntu users:

https://github.com/lmendiboure/DP_TP/blob/master/installation-guide-updated.md

- Download **anaconda**:

<https://www.anaconda.com/download>

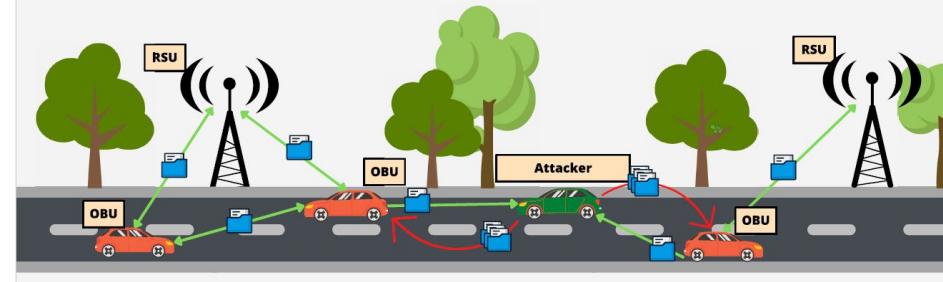
- Download **Python**:

<https://www.python.org/downloads/>

- Install necessary libraries:

pip install tensorflow

Intrusion Detection problem



- Anomaly based Intrusion Detection System
- Different attack types: Denial-of-Service (DoS), Sybil, position falsification, black-hole and gray-hole attack
- An open public dataset: VeReMi
 - Vehicular Reference Misbehavior (built specifically for testing V2X security)
 - A simulated dataset, generated using LuST and VEINS (based on OMNET++ and SUMO)
- Download dataset (only the zip file: **VeReMi_54000_57600_2022-9-11_19.12.56.zip**)
https://mega.nz/folder/z0pnGA4a#WFEUJSyS5_maabhcEI7HQA/folder/a1QxhaqC
- Python file for the application

https://github.com/ercansec/DataProcessing_CITS

Intrusion Detection problem - Dataset



VeReMi_50400_54000_2022-9-11_19_12_56



traceGroundTruthJSON-14.json



traceJSON-9-7-A0-50404-14.json



traceJSON-15-13-A13-50404-14.json

...



VeReMi_54000_57600_2022-9-11_19_12_56



traceGroundTruthJSON-15.json



traceJSON-6033-6031-A13-54004-15.json



traceJSON-6039-6037-A0-54004-15.json

...



VeReMi_57600_61200_2022-9-11_19_12_56



traceJSON-10143-10141-A0-57600-16.json



traceJSON-10149-10147-A0-57600-16.json

- Any json file
 - Type = 2 (GPS)
 - Sent messages
 - Type = 3 (BSM-Basic Safety Message)
 - Received messages
- Ground Truth json file
 - Type = 4 (Ground truth)
 - Received messages

traceJSON-**6033**-**6031**-**A13**-54004-15.json

- **6033**: receiver
- **6031**: OMNET++ module ID
- **A13**: attacker type

Intrusion Detection problem - Dataset

type = 2 (GPS)

- rcvTime
- pos & noise
- spd & noise
- acl & noise
- hed & noise

type = 3 (BSM)

- rcvTime
- sendTime
- sender
- senderPseudo
- messageID
- pos & noise
- spd & noise
- acl & noise
- hed & noise

type = 4 (Ground Truth)

- sendTime
- sender
- senderPseudo
- messageID
- pos & noise
- spd & noise
- acl & noise
- hed & noise

Data Processing Application

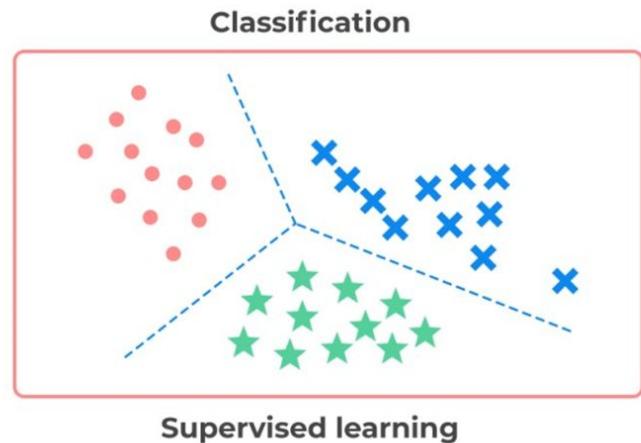
- Import data
- Descriptive statistics
- Check null values, missing data
- Data split for learning
- Implementation of ML methods
 - Classification
 - Clustering
- Output performance indicators
- Save results

Classification

Supervised Learning - with labels (classes)

Examples of Classification Methods:

- Decision Tree (DT)
- Support Vector Machine (SVM)
- k-Nearest Neighbor (kNN)
- Random Forest (RF)
- Extreme Gradient Boosting (XGBoost)
-

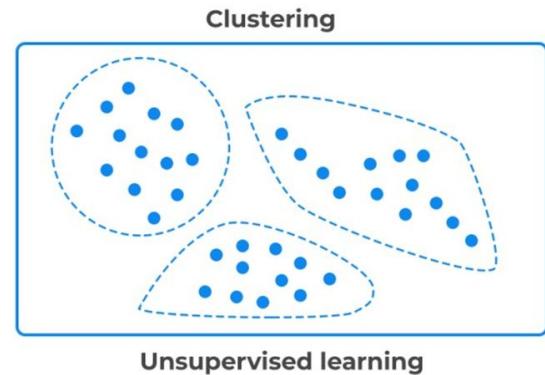


Clustering

Unsupervised Learning - without labels (classes)

Examples of Clustering Methods:

- Self Organizing Map (SOM)
- k-Means
- Gaussian Mixture Model (GMM)
-



Performance Indicators

Confusion Matrix:

	Predicted 0	Predicted 1
Actual 0	TN	FP
Actual 1	FN	TP

TN: True Negative

TP: True Positive

FN: False Negative

FP: False Positive

Evaluation Indicators

- ▶ Accuracy = $(TP+TN)/(TP+TN+FP+FN)$
- ▶ Precision = $TP/(TP+FP)$
- ▶ Recall = $TP/(TP+FN)$
- ▶ F1-Score = $2*Precision*Recall/(Precision+Recall)$

Homework

Homework

Driver Behavior Dataset

- Download dataset (**features_14.csv**):

<https://www.kaggle.com/datasets/shashwatwark/driving-behavior-dataset?resource=download>

<https://data.mendeley.com/datasets/jj3tw8kj6h/2>

Context

Driver behavior is one of the most important aspects in the design, development, and application of Advanced Driving Assistance Systems (ADAS) and Intelligent Transportation Systems (ITS), which can be affected by many factors. If you are able to measure the driving style of your staff, there is a lot of actions you can take in order to improve fleet safety, global road safety as well as fuel efficiency and emissions.

Content

- Dataset for modeling risky driver behaviors based on accelerometer (X,Y,Z axis in meters per second squared (m/s²) and gyroscope (X,Y, Z axis in degrees per second (^/s)) data.
- Sampling Rate: Average 2 samples (rows) per second
- Cars: Ford Fiesta 1.4, Ford Fiesta 1.25, Hyundai i20
- Drivers: 3 different drivers with the ages of 27, 28 and 37
- Driver Behaviors:
 - 1.Sudden Acceleration (Class Label: 1)
 - 2.Sudden Right Turn (Class Label: 2)
 - 3.Sudden Left Turn (Class Label: 3)
 - 4.Sudden Break (Class Label: 4)
- Best Window Size: 14 seconds
- Sensor: MPU6050
- Device: Raspberry Pi 3 Model B

Acknowledgements

Yuksel, Asim; Atmaca, Şerafettin (2020), "Driving Behavior Dataset", Mendeley Data, V2, doi: 10.17632/jj3tw8kj6h.2

Homework

A FAIRE :

- Importez les données (features_14.csv)
- Nettoyez le jeu de données (si besoin)
- Découvrez les variables
- Implémentez une méthode de Machine Learning pour la classification (Choisissez une méthode si vous travaillez individuellement ou deux méthodes si vous travaillez en binom et comparez avec XGBoost comme on a fait. **Il est interdit d'utiliser la méthode Random Forest!**)
- Utilisez les indicateurs accuracy, precision et recall pour évaluer la performance de votre méthode de classification et celle de XGBoost
- Commentez les résultats

Travaillez individuellement ou en binom. Envoyez votre fichier Python et compte-rendu de vos résultats avec votre commentaire à sercan@bordeaux-inp.fr

La date limite de dépôt : 8 décembre 2023 à 23h59