DATA MINING PROJECT

Elnara Yerbolatova

Hadiqa Alamdar Bukhari

Nishant Sushmakar

Olha Baliasina

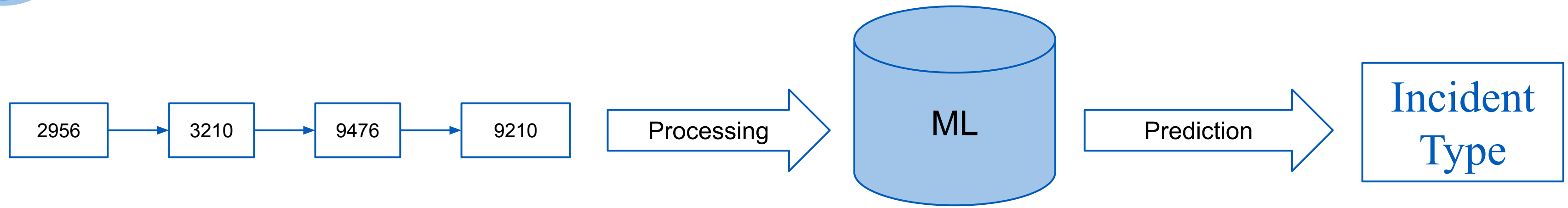
Link to the GitHub repository

ECOLE POLYTECHNIQUE DE BRUXELLES

PROBLEM STATEMENT

ULB ECOLE POLYTECHNIQUE DE BRUXELLES

1 Try to distinguish different incidents much more efficiently using ML



2 Try to predict if an incident occured in a given window sequence or not

ULB

ECOLE
POLYTECHNIQUE
DE BRUXELLES

Objective and methods

- Exploratory Data Analysis
- Feature Engineering
- Unsuccessful Exploratory Approaches
- Model 1: Multinomial Naive Bayes
- Model 2: Additional Filtering
- Comparison with Baseline SNCB model

ULB

ECOLE
POLYTECHNIQUE
DE BRUXELLES

EXPLORATORY DATA ANALYSIS

Extract meaningful signals from the data

Refine the validation strategy

Select the appropriate evaluation metric

Deeper understanding of the data sets structure

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1011 entries, 0 to 1010
Data columns (total 11 columns):
 #   Column                       Non-Null Count  Dtype
---  ------                       --------------  -----
 0   Unnamed: 0                   1011 non-null   int64
 1   incident_id                  1011 non-null   int64
 2   vehicles_sequence            1011 non-null   object
 3   events_sequence              1011 non-null   object
 4   seconds_to_incident_sequence 1011 non-null   object
 5   approx_lat                   1011 non-null   float64
 6   approx_lon                   1011 non-null   float64
 7   train_kph_sequence           1011 non-null   object
 8   dj_ac_state_sequence         1011 non-null   object
 9   dj_dc_state_sequence         1011 non-null   object
 10  incident_type                1011 non-null   int64
dtypes: float64(2), int64(3), object(6)
memory usage: 87.0+ KB
```

```
data.nunique()
```

```
Unnamed: 0                      1011
incident_id                     1011
vehicles_sequence               1011
events_sequence                 1011
seconds_to_incident_sequence    1011
approx_lat                      1011
approx_lon                      1011
train_kph_sequence              1004
dj_ac_state_sequence             745
dj_dc_state_sequence             966
incident_type                     12
dtype: int64
```

Column Description

When analyzing the data, we discovered the following insights:

- Multiple sensors can transmit data simultaneously, meaning the events are not sequential all the time but concurrent as well.
- Each sensor's data includes metadata, such as the vehicle ID, train speed, and the AC/DC state of the vehicle at the time of the event.

These observations allow us to construct a dictionary that represents the data in a more intuitive and organized way.

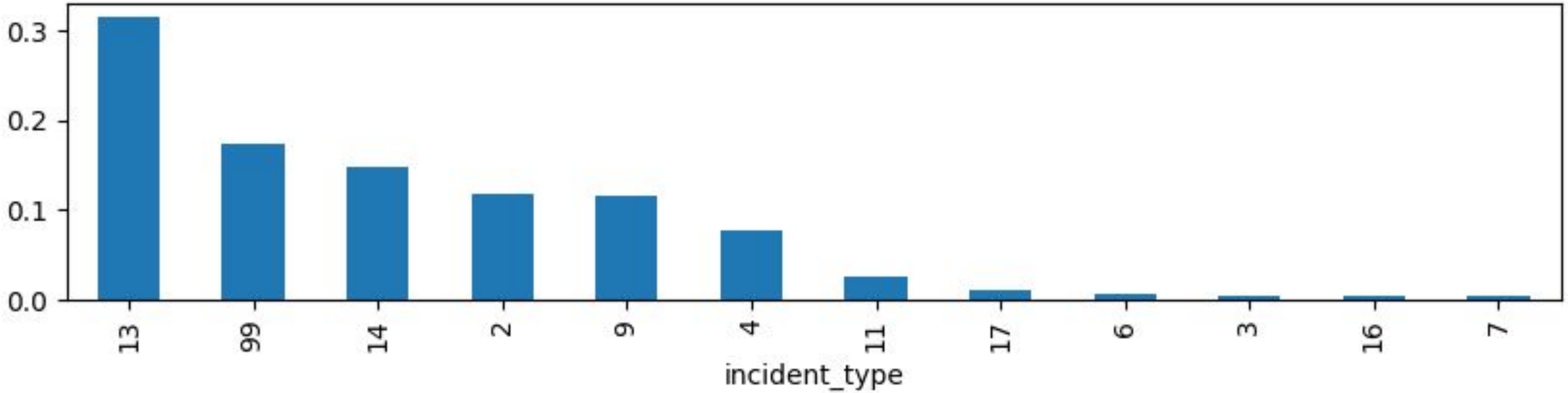
```
'time': {'vehicle_id': {'event_id':
{'train_speed': '0.0',
    'ac_state': 'False',
    'dc_state': 'False'}}}
```

Representation on the dictionary

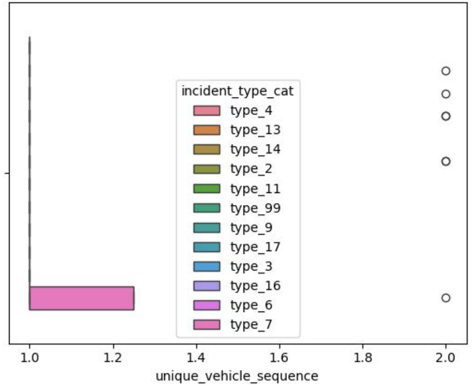
```
'-3942': {'609': {'4030': {'train_speed': '0.0',
    'ac_state': 'False',
    'dc_state': 'False'}}}

'1440': {'609': {'3236': {'train_speed': '0.0',
    'ac_state': 'False',
    'dc_state': 'True'},
    '2708': {'train_speed': '0.0',
    'ac_state': 'False',
    'dc_state': 'True'}}}
```

Dictionary Structure on the dataset

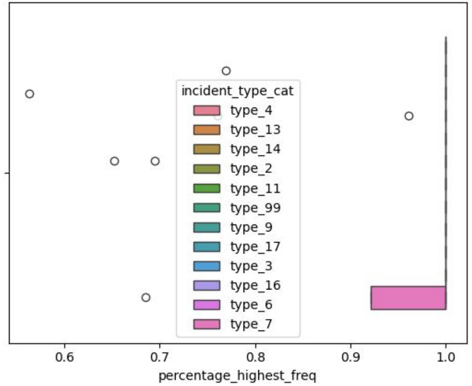
After Analyzing the incident_type in the dataset it becomes very clear that we have to use Stratified K-Fold Cross Validation and Evaluation Metric Should be F1-Macro as there is imbalance in the classes.



ECOLE
POLYTECHNIQUE
DE BRUXELLES

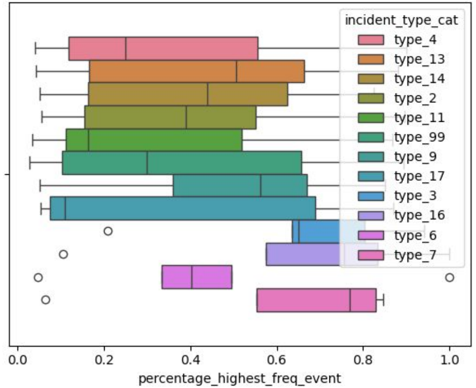


Percentage of frequency of the highest occuring event id in the sequence

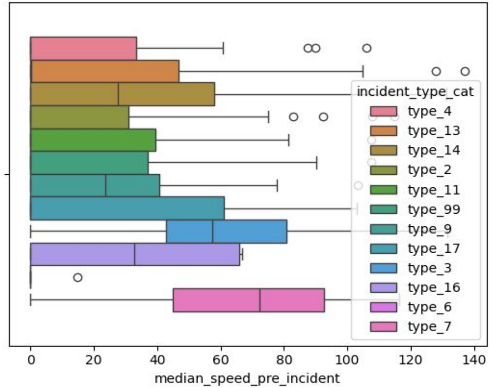


Median speed of vehicle in a given window of sequence

Unique number of vehicle ids seen in the list of sequence

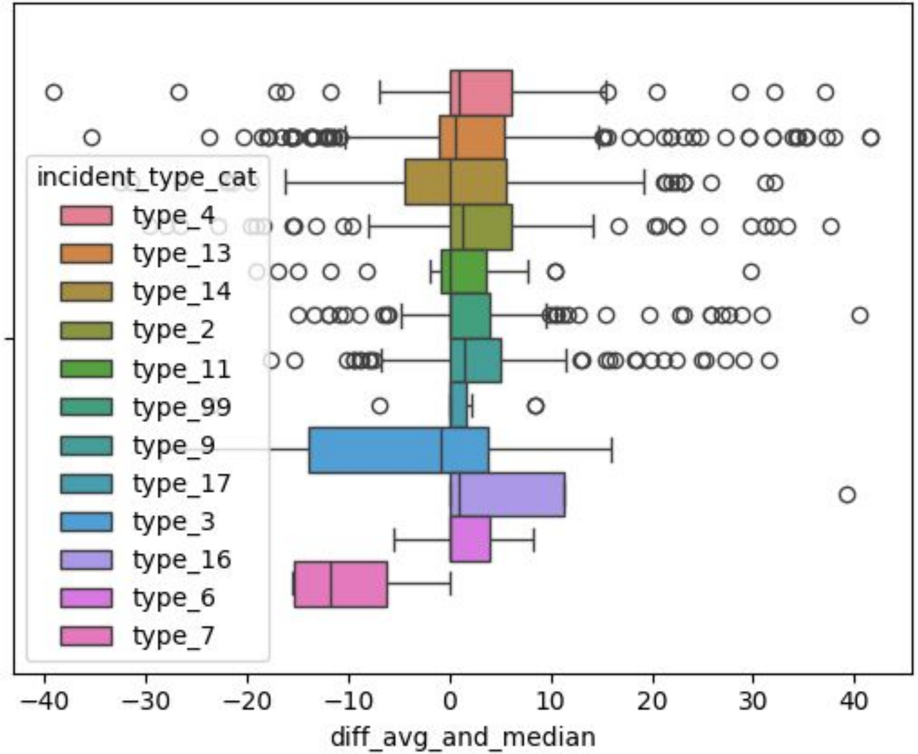


Percentage of frequency of the highest occuring vehicle id in the sequence



ULB ECOLE POLYTECHNIQUE DE BRUXELLES

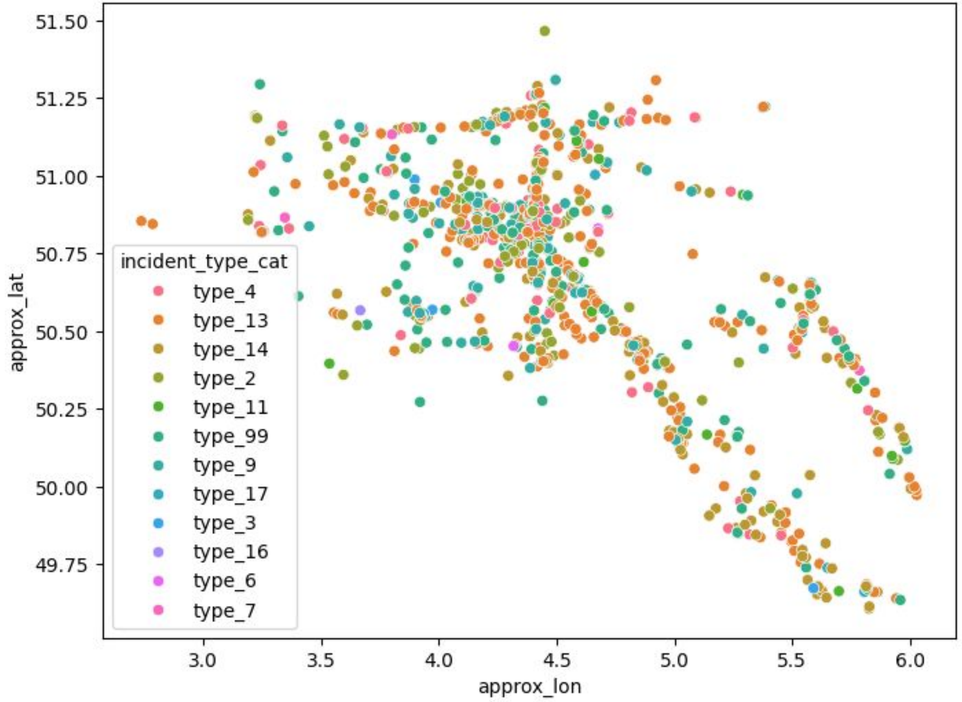
Mean and Median Difference of Speeds



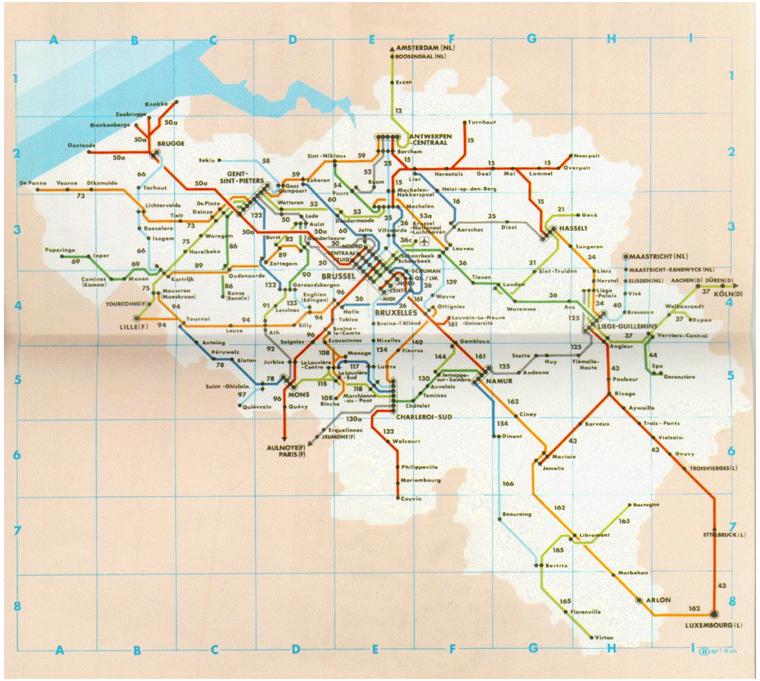
Difference between mean and median of the speed in a sequence window intuition was to grasp acceleration and deceleration of the train using this feature

ECOLE
POLYTECHNIQUE
DE BRUXELLES

Inspecting geodata



Scatter plot of the incidents



SNCB Network map

Word Cloud for Target Label: 4

Word Cloud for Target Label: 13

Word Cloud for Target Label: 14

Word Cloud for Target Label: 2

Word Cloud for Target Label: 11

Word Cloud for Target Label: 99

Word Cloud for Target Label: 9

Word Cloud for Target Label: 17

ECOLE
POLYTECHNIQUE
DE BRUXELLES

Word Cloud for Target Label: 3

Word Cloud for Target Label: 16

Word Cloud for Target Label: 6

Word Cloud for Target Label: 7

ULB ECOLE POLYTECHNIQUE DE BRUXELLES

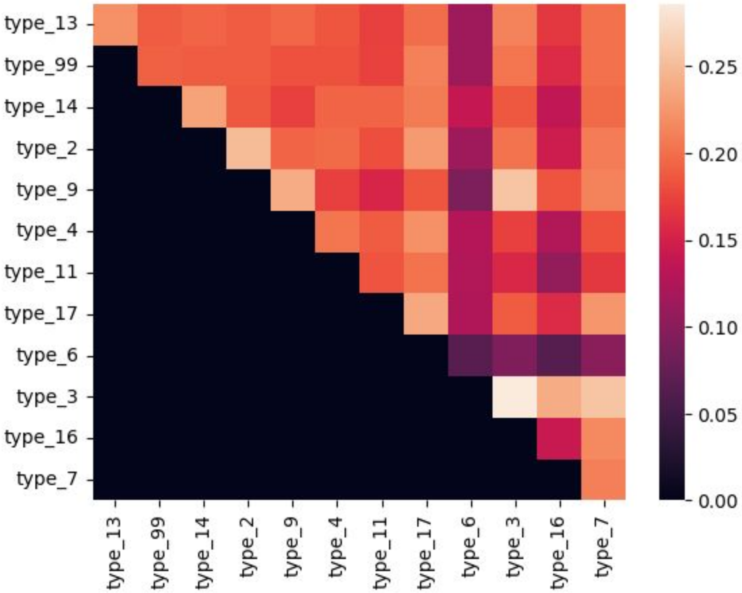
Bag of words for incidents 3, 6, 7 and 16

- Very few samples of these incidents are present
- Model accuracy reduced as they were often overlooked
- Idea was to classify these incidents separately
- BOW used to calculate the frequency of ngrams from lengths 1-5.
- event_id=2956 was the most common for incidents 3 and 7
- hard to distinguish these incidents.

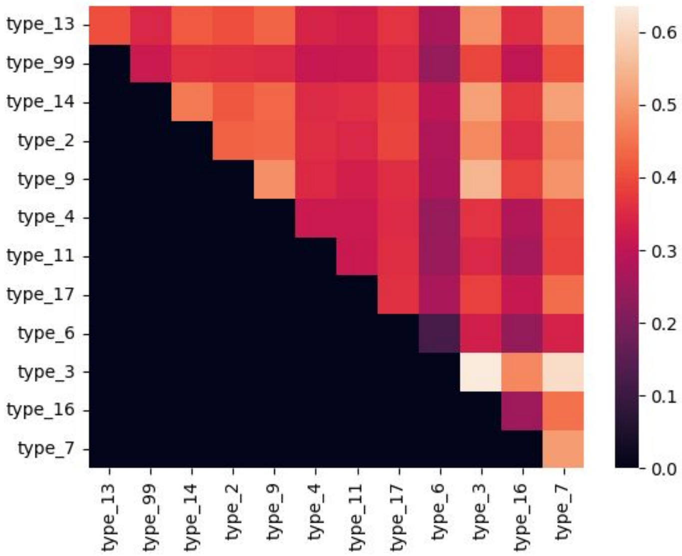
```
{3: {1: {'2956': 937}, 2: {'2956 2956': 871}, 3: {'2956 2956 2956': 809}, 4: {'2956 2956 2956 2956': 754}, 5: {'2956 2956 2956 2956 2956': 703}}
, 16: {1: {'3636': 23}, 2: {'3636 3658': 23}, 3: {'3636 3658 4120': 10}, 4: {'3636 3658 4120 3636': 9}, 5: {'3636 3658 4120 3636 3658': 9}}
, 6: {1: {'3636': 31}, 2: {'3636 3658': 31}, 3: {'4066 3636 3658': 19}, 4: {'3636 3658 4066 3636': 13}, 5: {'3636 3658 4066 3636 3658': 13}}
, 7: {1: {'2956': 644}, 2: {'2956 2956': 593}, 3: {'2956 2956 2956': 547}, 4: {'2956 2956 2956 2956': 503}, 5: {'2956 2956 2956 2956 2956': 466}}}
```

We conducted a similarity analysis of event sequences using metrics like token overlap similarity and cosine similarity. For each analysis, we calculated the mean similarity scores of a record compared to records of the same incident type and records of other incident types. This provided insights into how event patterns differ within and between incident types.

The similarity scores from both token overlap and cosine metrics produced consistent results. They revealed that **type_6 incidents** are highly dissimilar, both internally and compared to other incident types. In contrast, **type_3 incidents** show strong similarity within their own records, making them easier to identify. However, type_3 also shares significant similarity with other types like 16, 7, and 9, suggesting these might represent related or overlapping categories of incidents.



Mean token overlap similarity



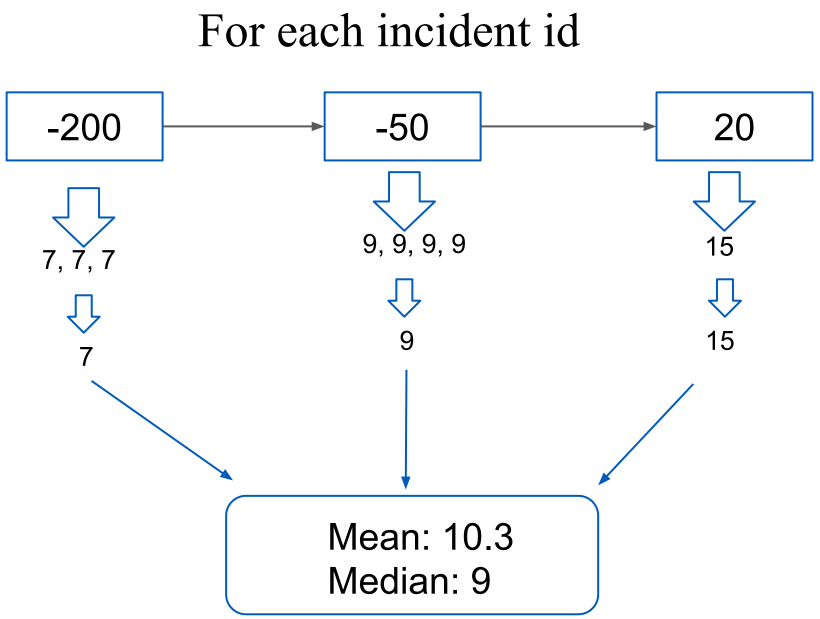
Mean Cosine Similarity

FEATURE SELECTION

Mean and Median Speed

- Create dictionary:
  - save just one value of speed for a vehicle in a sequence at a given time
- Dictionary structure:
  - {-3583: {609: {2970: {'train_speed': 0.0, 'ac_state': False, 'dc_state': True}}}

    , -3546: {609: {4092: {'train_speed': 0.0, 'ac_state': False, 'dc_state': True}}}
- Calculate mean and median speed for each vehicle in the window of speed sequence



AC and DC states

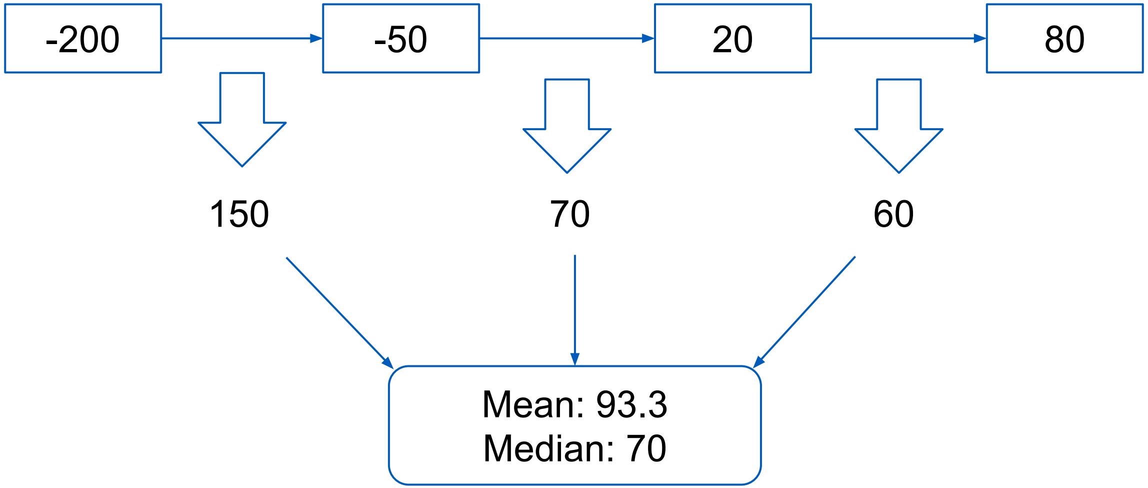
ac_state	dc_state		
True	True	⇨	ac_dc_t_t_count+=1
True	False	⇨	ac_dc_t_f_count+=1
False	True	⇨	ac_dc_f_t_count+=1
False	False	⇨	ac_dc_f_f_count+=1

TF-IDF most frequent subsequence

- Term Frequency-Inverse Document Frequency used to evaluate the importance of a term (n-gram of event_ids) in a document (incident_ids) relative to a collection of documents (incident_types).
- Used to see which event subsequence happened most frequently for each incident_type
- Helped overcome the frequency bias of specific event ids if they had a low IDF score.

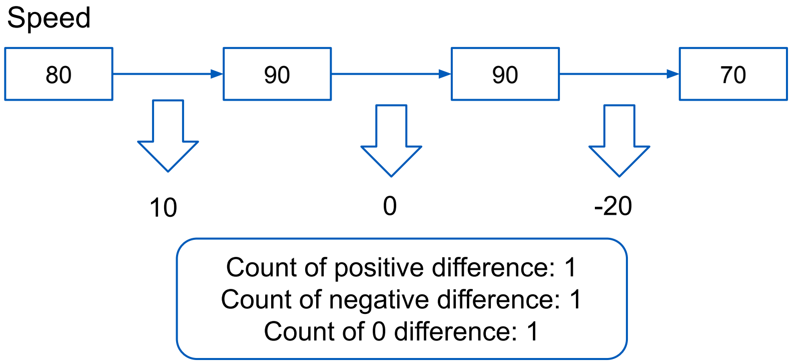
Mean and Median Time Difference between Consecutive Events

For each incident id:



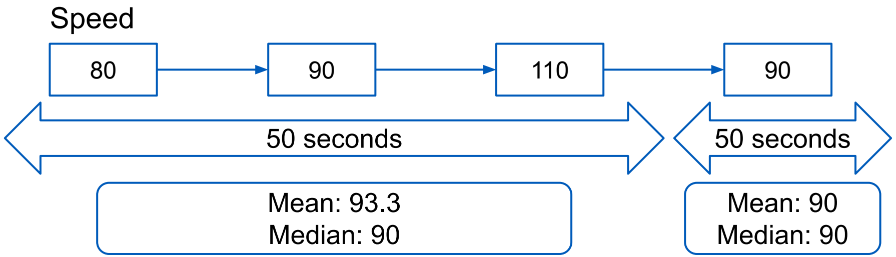
Acceleration and Deceleration Frequencies

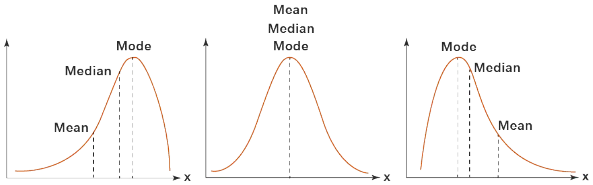
Approach 1



1. Calculate the differences between consecutive speeds
2. Count the cases of positive, negative and 0 differences
3. Find frequencies of each case for an incident id

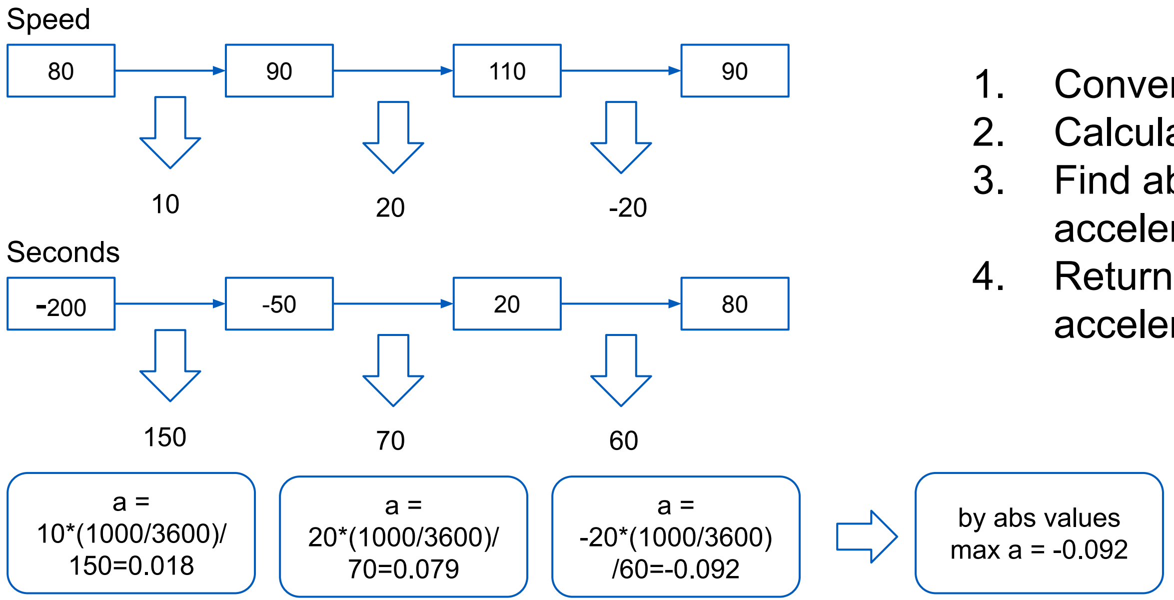
Approach 2





1. For each 50 second-window calculate mean and median of speeds
2. Acceleration: mean>median, deceleration: mean<median, constant: mean = median
3. Find frequencies of each case for an incident id

Maximum Acceleration and Deceleration



1. Convert k/h to m/s
2. Calculate acceleration
3. Find absolute maximum of accelerations
4. Return the signed value of maximum acceleration / deceleration

Incident Category

Based on the count of pre or post incident events reported

Category -1

Seconds

-200 → -50 → -30 → 20

Category 1

Seconds

-10 → 40 → 100 → 200

Category 0

Seconds

-200 → -50 → 20 → 80

UNSUCCESSFUL EXPLORATORY APPROACHES

SCIENTIA VINCERE TE

Using Large Language Models to find the incident types

We designed prompts containing examples of event sequences and asked the LLM to determine whether each given sequence belonged to the same incident type. However, the results showed that the LLM models, specifically ChatGPT 4o and o1, did not perform well on these prompts, highlighting its limitations in this task.

Link to the Chat (model 4o)

Link to the Chat (model o1)

ULB ECOLE POLYTECHNIQUE DE BRUXELLES

Clustering

Steps applied

Remove:

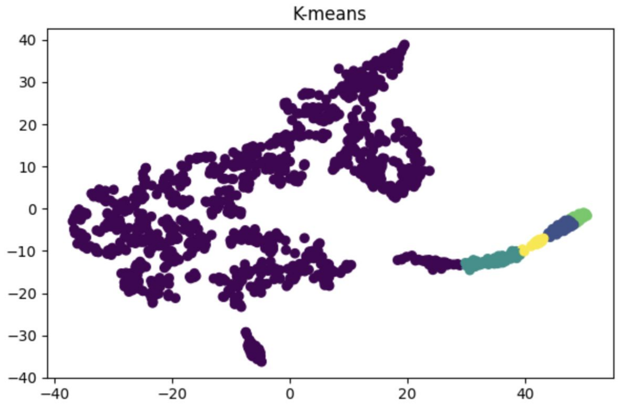
- one-dominant-value features
- highly-correlated features*

Apply:

- robust scaler
- PCA

Try:

- K-means
- DBSCAN
- Agglomerative clustering



Example

Outcome

Best result:

- identifying five distinct classes using K-means

The problem:

- the classes did not describe the incident types
- within one incident type we could find all of the 5 classes

Event Time Sequence

Overlapping Time Windows

Data leakage

Overly optimistic performance evaluations

Event Time Sequence

No Overlap Time Windows

No huge performance boost

Model gets confused

Inadequate features

Most frequent incident location:

- DBScan Clustering to segregate by incident type

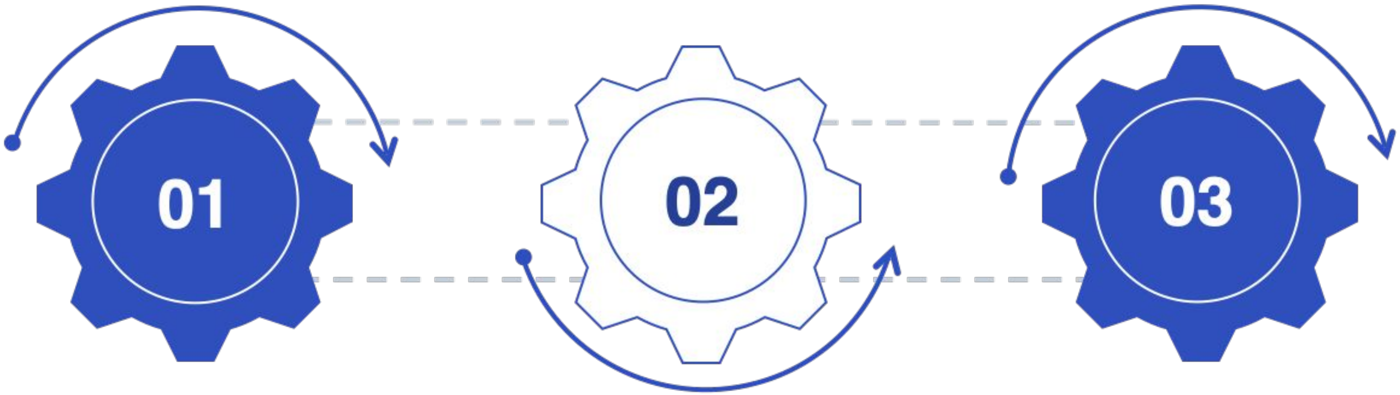
Most frequent subsequence per incident type:

- Prefix spanning
- Apriori algorithm

FIRST APPROACH (MODEL 1)

ULB ECOLE POLYTECHNIQUE DE BRUXELLES

Data Cleaning



Removal of rare classes

incident types 3, 6, 7, 16

Noise reduction

threshold 0.15

Time window selection

from -9,600 sec to +600 sec

ECOLE POLYTECHNIQUE DE BRUXELLES

Next steps

Feature Creation

- TF-IDF
- ngrams range [1;5]
- best average validation score

Scaled Data

- MinMaxScaler
- scaling range [0;1]

Model 1

Model Used:

Multinomial Naive Bayes

Validation Strategy Used:

5 fold Stratified K Fold Cross Validation

ULB ECOLE POLYTECHNIQUE DE BRUXELLES

Model 1. Performance Overview

Incident type	Precision	Recall	F1-Score	Support
2	0.86	0.79	0.83	24
4	1	1	1	16
9	1	1	1	24
11	1	1	1	5
13	0.92	0.95	0.94	63
14	1	1	1	30
17	1	1	1	2
99	1	1	1	33
Accuracy			0.96	197
Macro Avg	0.97	0.97	0.97	197
Weighted Avg	0.96	0.96	0.96	197

Average training Precision:
0.9940830168

Average training Recall:
0.9929852194

Average training f1-macro:
0.9934108884

Average validation Precision:
0.947236988

Average validation Recall:
0.9157421985

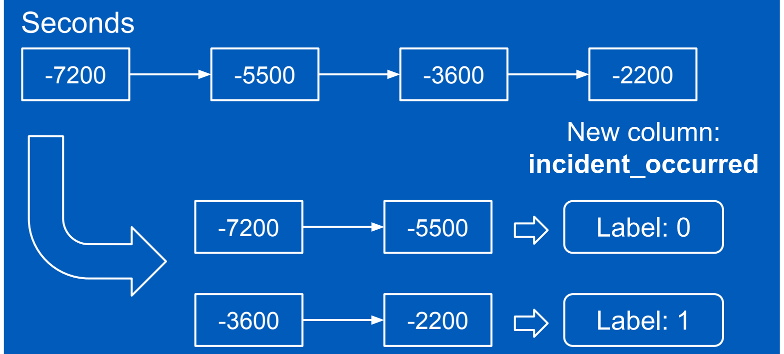
Average validation f1-macro:
0.9275312312

SECOND APPROACH (MODEL 2)

ULB ECOLE POLYTECHNIQUE DE BRUXELLES

Incident Prediction Model

Scheme



Steps

Create a dataset with non overlap windows of duration 1hr 200 sec from +200 sec to -14400 secs

- Label the the window which includes the reporting time i.e 0 sec as the incident_occured = 1
  - New target label - incident_occurred
    - Apply a classifier on the new dataset

Results

Best model:	LightGBM
F1-score:	0.75

Assumption

The events before 1 hours prior to the reporting of incident do not lead to any incident!

1. **Noise removal**
   Filter data using a relevance threshold (0.15)

2. **Data augmentation**
   set window size as 1hr 200 sec, create records per incident type from +200 sec to -14400 sec

3. **Feature creation**
   Apply TF-IDF (5-gram) and other listed features on augmented data

4. **Initial prediction**
   Run the augmented dataset through LGBM to predict incidents

5. **Filtering**
   Keep records with incident probability ≥ 0.8

6. **Refinement**
   Extract TF-IDF features from these selected records

Model 2. Performance Overview

Incident type	Precision	Recall	F1-Score	Support
2	0.9	0.64	0.75	14
4	0.71	1	0.83	5
6	0	0	0	2
11	0.2	1	0.33	1
13	0.87	0.93	0.9	42
14	0.86	1	0.92	12
99	1	0.17	0.29	6
Accuracy			0.82	82
Macro Avg	0.65	0.68	0.57	82
Weighted Avg	0.84	0.8	0.82	82

Average training Precision:
0.9986111111
Average training Recall:
0.9998084291
Average training f1-macro:
0.9991916324

Average validation Precision:
0.5438416553
Average validation Recall:
0.5162435428
Average validation f1-macro:
0.4880435156

Model 2. Performance Overview

Average training Precision:	Average validation Precision:
1.0	0.6337479915
Average training Recall:	Average validation Recall:
1.0	0.5837711199
Average training f1-macro:	Average validation f1-macro:
1.0	0.5834868638

Scores, obtained by excluding the rare incident types: 16,6,3 and 7

Model comparison with Baseline SNCB

It is not an apple to apple comparison.

We really appreciate the idea to remove noise from the data using the relevance metric it came handy in both of our approaches.

While the baseline tries to use ensemble model on various windows we try to improve the selection of the best window for prediction of incident types.

We also focus on the most recurring set of incident types and try to denoise by not getting the model confused with certain different classes which don't give us statistically significant results.