Forecasting Urban Traffic Patterns in London using Hybrid AI Techniques

Master's thesis for MSc in Fundamental Principles of Data Science

Author: Theodoros Lambrou
Supervisor: Dr. Jordi Vitrià
Universitat de Barcelona

Introduction

- Urban traffic prediction critical for managing congestion and enhancing mobility.
- Importance in real-time management and long-term planning (infrastructure, reducing emissions, safety).
- Historically reliant on traditional methods (time-series, statistical modeling).
- Increasing potential due to advancements in data availability and machine learning (ML).

Problem Formulation

- Objective: Predict hourly traffic severity on London road segments.
- Multi-class classification task:
  - Class 0: Normal traffic
  - Class 1: Mild congestion
  - Class 2: Severe congestion
- Challenges:
  - Highly imbalanced classes, especially severe congestion.
  - lIntegration of diverse data sources
  - Need for interpretability and fairness

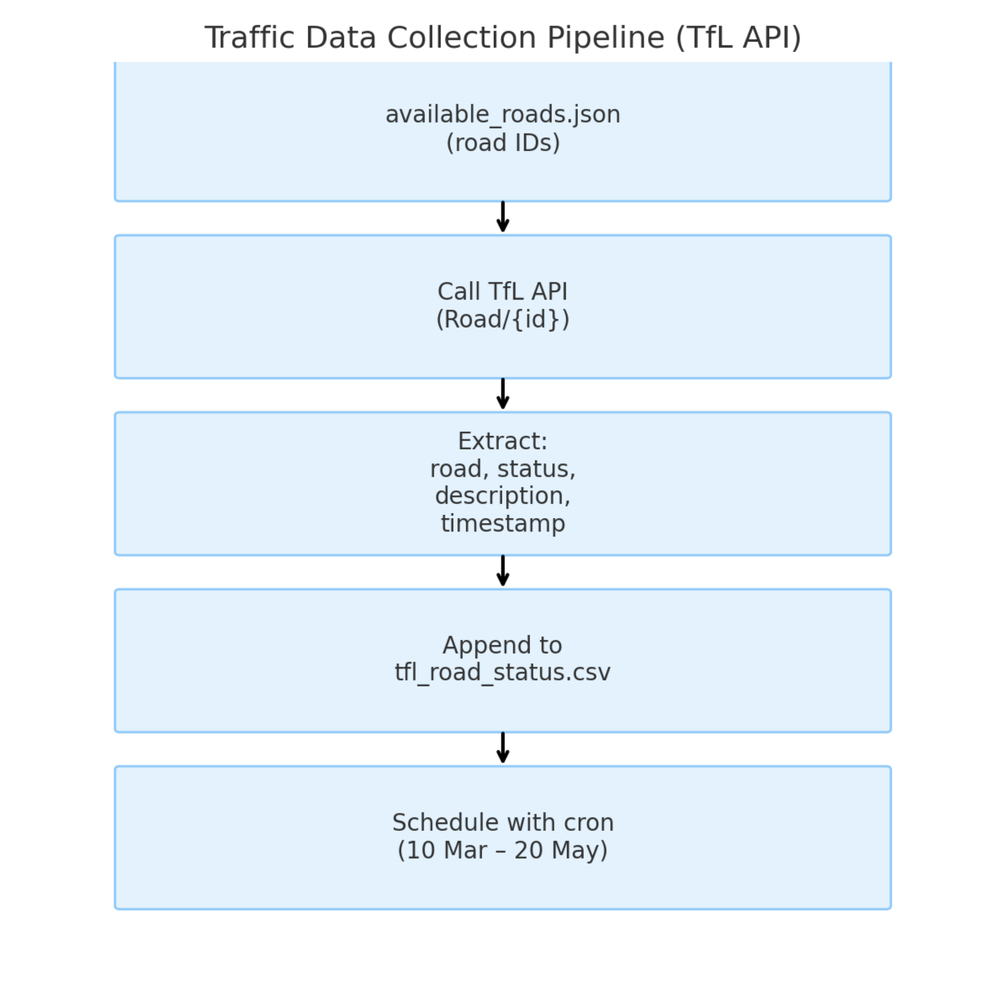
Data Sources

Traffic Data

- Source: Transport for London (TfL) Unified API (https://api.tfl.gov.uk/)
- Collection method: Python script via CRON job
- Period: March 10 to May 20, 2025
- Attributes: Timestamp, road name, severity (numeric, description)

Data Sources

Traffic Data




```python
APP_KEY = "045e9ab8f6164cb8bc07cd27bcff2109"

# Loading road list
project_root = os.path.abspath(os.path.join(os.path.dirname(__file__), ".."))
data_dir = os.path.join(project_root, "data")
with open(os.path.join(data_dir, "available_roads.json")) as f:
    roads = json.load(f)

# there may have too many roads for one request so splitting into chunks
def chunk(lst, size):
    for i in range(0, len(lst), size):
        yield lst[i:i + size]

records = []
for road_chunk in chunk(roads, 30):  # 30 at a time first testing
    url = f"https://api.tfl.gov.uk/Road/{','.join(road_chunk)}"
    params = {"app_key": APP_KEY}
    response = requests.get(url, params=params)

    if response.status_code == 200:
        data = response.json()
        timestamp = datetime.now().isoformat()
        for road in data:
            records.append({
                "road": road["displayName"],
                "status": road["statusSeverity"],
                "description": road["statusSeverityDescription"],
                "timestamp": timestamp
            })
    else:
        print("Error for chunk:", response.status_code)
        print(response.text)

# Saving results
df = pd.DataFrame(records)
os.makedirs(data_dir, exist_ok=True)
file_path = os.path.join(data_dir, "tfl_road_status.csv")
df.to_csv(file_path, index=False, mode='a', header=not os.path.exists(file_path))

print(f"Saved {len(df)} records to {file_path}")
```

Data Sources

Traffic Data

	road	status	description	timestamp
0	A1	Good	No Exceptional Delays	2025-03-10T00:08:00
1	Western Cross Route	Good	No Exceptional Delays	2025-03-10T00:08:00
2	Southern River Route	Good	No Exceptional Delays	2025-03-10T00:08:00
3	Inner Ring	Good	No Exceptional Delays	2025-03-10T00:08:00
4	Farringdon Cross Route	Good	No Exceptional Delays	2025-03-10T00:08:00
...	...	...	...	...
79483	A2	Good	No Exceptional Delays	2025-05-20T23:34:00
79484	A1	Good	No Exceptional Delays	2025-05-20T23:34:00
79485	Southern River Route	Good	No Exceptional Delays	2025-05-20T23:34:00
79486	A24	Minor	Minor Delays	2025-05-20T23:34:00
79487	Western Cross Route	Good	No Exceptional Delays	2025-05-20T23:34:00

79488 rows × 4 columns

Data Sources

Weather data

- Source: Open-Meteo (https://open-meteo.com/en/docs)
- Historical weather data for 10 March 2025 to 20 may 2025
- Attributes: temperature_2m (°C), precipitation (mm), rain (mm), snowfall (cm), wind_speed_10m (km/h), wind_gusts_10m (km/h), cloud_cover (%)
- Scaling: I used StandardScaler to transform the weather values to small standardised range, to make sure that features with different numeric ranges are on a comparable scale, so the model treats them fairly during learning. I
- This data was merged into the final working dataset

Data Sources

Weather data

latitude	longitude	elevation	utc_offset_seconds	timezone	timezone_abbreviation		
51.493847	-0.1630249	1.0	3600	Europe/London	GMT+1		
time	temperature_2m (°C)	precipitation (mm)	rain (mm)	snowfall (cm)	wind_speed_10m (km/h)	wind_gusts_10m (km/h)	cloud_cover (%)
2025-03-10T00:00	8.7	0.00	0.00	0.00	6.9	15.8	97
2025-03-10T01:00	8.6	0.00	0.00	0.00	6.4	14.0	29
2025-03-10T02:00	8.8	0.00	0.00	0.00	7.3	14.8	99
2025-03-10T03:00	8.4	0.00	0.00	0.00	7.5	14.8	99
2025-03-10T04:00	8.0	0.00	0.00	0.00	7.1	14.8	11
2025-03-10T05:00	7.4	0.00	0.00	0.00	7.8	15.1	98
2025-03-10T06:00	7.5	0.00	0.00	0.00	7.5	16.6	100
2025-03-10T07:00	7.5	0.00	0.00	0.00	8.1	16.6	100
2025-03-10T08:00	7.3	0.10	0.10	0.00	8.0	17.6	100
2025-03-10T09:00	7.5	0.20	0.20	0.00	7.8	16.9	100
2025-03-10T10:00	7.9	0.00	0.00	0.00	6.2	17.3	100
2025-03-10T11:00	8.7	0.00	0.00	0.00	5.3	13.3	100
2025-03-10T12:00	9.7	0.10	0.10	0.00	6.2	16.9	100
2025-03-10T13:00	10.7	0.00	0.00	0.00	5.4	15.8	100
2025-03-10T14:00	12.0	0.00	0.00	0.00	9.1	23.0	89
2025-03-10T15:00	12.9	0.00	0.00	0.00	9.3	23.8	85
2025-03-10T16:00	13.9	0.00	0.00	0.00	11.3	27.0	57
2025-03-10T17:00	13.5	0.00	0.00	0.00	12.7	29.9	94
2025-03-10T18:00	12.3	0.00	0.00	0.00	11.9	29.2	100
2025-03-10T19:00	10.7	0.00	0.00	0.00	13.7	30.6	100
2025-03-10T20:00	9.4	0.00	0.00	0.00	14.7	32.8	80
2025-03-10T21:00	8.1	0.00	0.00	0.00	13.5	33.1	72
2025-03-10T22:00	7.2	0.00	0.00	0.00	11.1	29.9	100
2025-03-10T23:00	7.0	0.00	0.00	0.00	10.9	24.1	100
2025-03-11T00:00	6.8	0.00	0.00	0.00	12.0	26.3	100
2025-03-11T01:00	6.9	0.00	0.00	0.00	13.8	30.2	100
2025-03-11T02:00	6.8	0.10	0.10	0.00	14.8	33.1	100
2025-03-11T03:00	6.6	0.10	0.10	0.00	14.3	33.5	100

Feature Engineering

- Checked data for any inconsistencies/missing values
- Temporal features:
  - Hour of day, day of week, weekend indicator, rush hour indicator
- Historical features: Rolling historical severity (1 & 2 hours beforehand)
  - These features (prev_1h_severity, prev_2h_severity) are meant to help the models incorporate patterns from recent traffic severity at each road and time.

Final (merged) Dataset

road	status	description	timestamp	hour	weekday	day_of_week	is_weekend	is_rush_hour	severity_level	prev_1h_severity	prev_2h_severity
A1	Good	No Exceptional Delays	2025-03-10 00:08:00	0	Monday	0	0	0	0	-1	-1
A1	Good	No Exceptional Delays	2025-03-10 00:38:00	0	Monday	0	0	0	0	0	-1
A1	Minor	Minor Delays	2025-03-10 01:08:00	1	Monday	0	0	0	1	0	0
A1	Good	No Exceptional Delays	2025-03-10 01:38:00	1	Monday	0	0	0	0	1	0
A1	Good	No Exceptional Delays	2025-03-10 02:08:00	2	Monday	0	0	0	0	0	1
A1	Good	No Exceptional Delays	2025-03-10 02:38:00	2	Monday	0	0	0	0	0	0
A1	Good	No Exceptional Delays	2025-03-10 03:08:00	3	Monday	0	0	0	0	0	0

temperature_2m	precipitation	rain	snowfall	wind_speed_10m	wind_gusts_10m	cloud_cover
-0.6353594835889136	-0.13449201654075912	-0.13424639854273523	-0.027862365509370404	-0.7334937630303514	-0.941858819842457	1.1248389044578764
-0.6353594835889136	-0.13449201654075912	-0.13424639854273523	-0.027862365509370404	-0.7334937630303514	-0.941858819842457	1.1248389044578764
-0.6553642426655077	-0.13449201654075912	-0.13424639854273523	-0.027862365509370404	-0.8504369670564482	-1.1129019124671082	-0.445975906392917
-0.6553642426655077	-0.13449201654075912	-0.13424639854273523	-0.027862365509370404	-0.8504369670564482	-1.1129019124671082	-0.445975906392917
-0.6153547245123191	-0.13449201654075912	-0.13424639854273523	-0.027862365509370404	-0.6399391998094741	-1.0368827601894854	1.171039340071135
-0.6153547245123191	-0.13449201654075912	-0.13424639854273523	-0.027862365509370404	-0.6399391998094741	-1.0368827601894854	1.171039340071135
-0.695373760818696	-0.13449201654075912	-0.13424639854273523	-0.027862365509370404	-0.5931619181990354	-1.0368827601894854	1.171039340071135
-0.695373760818696	-0.13449201654075912	-0.13424639854273523	-0.027862365509370404	-0.5931619181990354	-1.0368827601894854	1.171039340071135

Baseline Models

- Simple categorical models: logistic regression, decision tree, random forest
  - Used only the temporal features (i.e. without lag features and weather data)
  - High accuracy but highly imbalanced; important benchmark to compare later
- Graph Neural Network (GNN) exploratory experiment: Limited due to insufficient spatial metadata, excluded from further experiments
- Probability-based baseline (historical severity): Assign class with highest historical probability
  - Moderately successful, useful baseline
  - Incorporated this data later on as part of the advanced ML models.

Hybrid ML approaches

Random Forest & XGBoost

- For these approaches I used the full engineered dataset, as well as the historical probabilities obtained from the baseline statistical model
- For both models I experimented and added new features iteratively to check the effect of each addition

Advanced ML approaches

Random Forest

- Motivation: Robust to noise, captures feature interactions effectively, suitable for imbalanced datasets.
- Initial model (8a):
  - Performance: Good accuracy (0.753) but low recall for minority classes (Class 1: 0.13, Class 2: 0.06).
- Class Weighting (8b):
  - Adjusted class weighting to 'balanced'.
  - Slight improvement in minority class recall (Class 1: 0.16, Class 2: 0.07), minor drop in accuracy (0.721).

Advanced ML approaches

Random Forest

- Extended Temporal Features (8c):
  - Added binary time indicators (weekend, rush hour).
  - Negligible improvement; similar results to 8b, highlighting limited incremental value of basic temporal indicators.
- GridSearch Hyperparameter Tuning (8d):
  - Explored parameters: `n_estimators, max_depth, min_samples_leaf`
  - Optimized parameters significantly improved minority class recall (Class 2 recall increased substantially to 0.65).
  - Trade-off: Overall accuracy dropped to 0.561; Macro F1 improved (0.42).

Advanced ML approaches

Random Forest

- Final Selected RF Model (8e):
  - Balanced subsampling (`class_weight='balanced_subsample'`) chosen to handle class imbalance effectively.
  - Optimal hyperparameters from GridSearch (`n_estimators=200, max_depth=10, min_samples_leaf=2`).
  - Achieved best balanced performance:
    - Accuracy: 0.638
    - Recall Class 1: 0.39; Recall Class 2: 0.43
    - Macro F1-score: 0.44 (best compromise between accuracy and fairness).

  - Further Explorations (8f–8i): Tested additional complexity (entropy features, interaction terms) without significant improvements, confirming model 8e as optimal.

Advanced ML models

XGBoost

- Motivation: Popular gradient boosting model, effective in structured data problems; excellent for performance benchmarking.
- Initial XGBoost Model (9):
  - All engineered features included.
  - Baseline results: Accuracy comparable to Random Forest initial models (0.753); low minority recall (0.06 Class 2).
  - Class Balancing with `scale_pos_weight` (9b):
    - Improved minority recall marginally, but overall metrics remained similar.

Advanced ML models

XGBoost

- Hyperparameter Tuning (9c, 9d):
  - Explored `max_depth`, `n_estimators`, `min_child_weight`, and early stopping.
  - Improved Macro F1 and minority recall, with stable accuracy (~0.75).
- Final Selected XGBoost Model (9e):
  - Best hyperparameters: `n_estimators=250`, `max_depth=8`, `min_child_weight=4`, `subsample=0.8`.
  - Performance:
    - Highest accuracy across all experiments: 0.778
    - Weighted F1: 0.73
    - Moderate minority recall (Class 1: 0.23, Class 2: 0.16), lower than RF 8e.

Handling Class Imbalance

- Significant challenge: Severe congestion (Class 2) underrepresented
- Class weighting strategies (balanced subsampling)
- Prioritized metrics: Macro F1-score and recall for minority class to ensure fairness and robustness

Model Selection

- Metrics evaluated: Accuracy, Macro F1, weighted F1, class-specific recall
- Importance of minority class recall over accuracy for operational relevance
- Random Forest 8e prioritized fairness and minority class recall, crucial for operational relevance.
- XGBoost 9e provided benchmark in accuracy and overall performance, confirming robustness of the feature set.
- Final model selection (RF 8e) driven by balanced performance and operational ethics: underestimating severe congestion has higher costs than slight accuracy reductions.

Comparative Analysis of Models

- Baseline models: accuracy (~0.61–0.67), very poor minority recall (close to 0)
- Random Forest (8e): Balanced performance
- XGBoost (9e): Highest overall accuracy but lower minority recall
- Macro F1 is key, since the aim is fairness: ensuring the model doesn't just predict 'good traffic' but also identifies minor and serious congestion reliably.

Metric	RF 8e	XGB 9e
Accuracy	0.75	**0.778**
Macro F1	**0.44**	0.41
Weighted F1	0.70	**0.73**
Recall (Class 1)	~**0.43**	~0.38
Recall (Class 2)	~**0.38**	~0.32

Detailed Evaluation

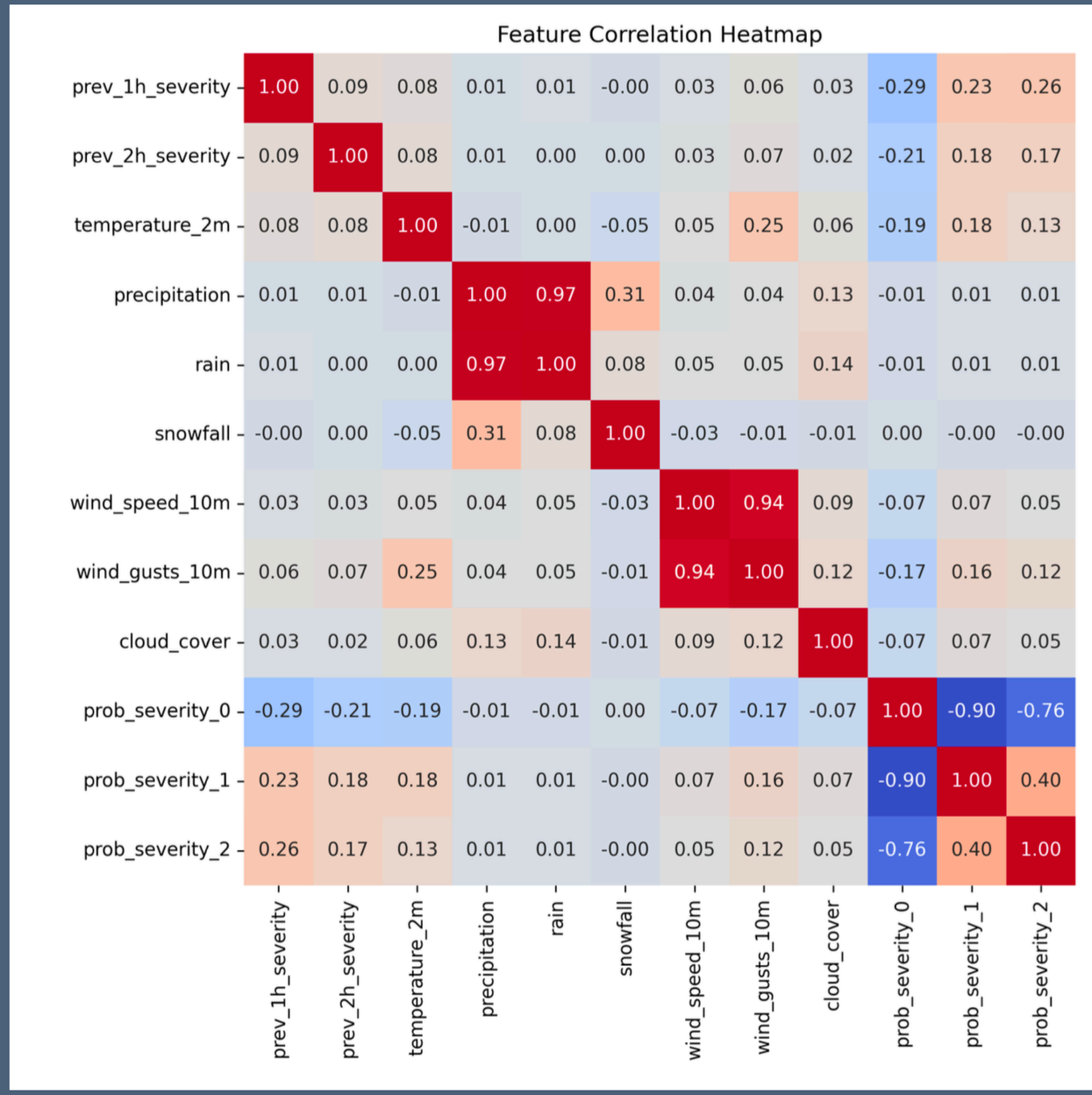
Random Forest (model ‘8e’)

- Feature correlation heat map
- Learning curve analysis
- Precision-Recall curves
- Confusion Matrix

Detailed Evaluation

Feature Correlation Heatmap

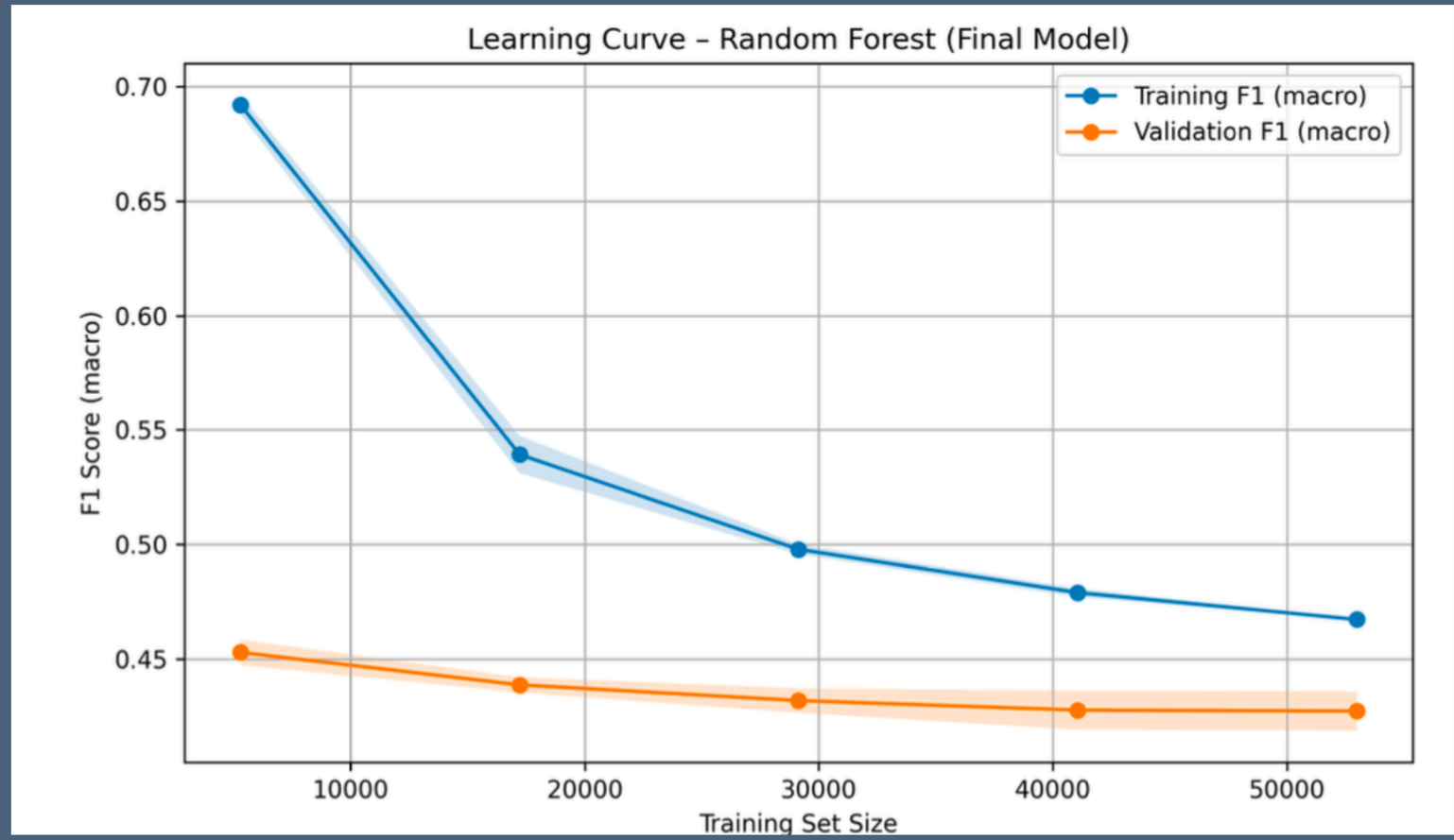
- Low Correlation with Target Features: Weather features appear weakly correlated with severity a signal that they may play a smaller role in predictions.”
- Baseline probabilities for different severity levels are inversely related, when the model thinks severity 0 is likely, 1 and 2 are less likely, and vice versa.
- prev_1h_severity shows moderate positive correlation with prob_severity_1 and prob_severity_2. This supports the intuition that recent severity history is predictive of future congestion.



Detailed Evaluation

Learning Curve Analysis

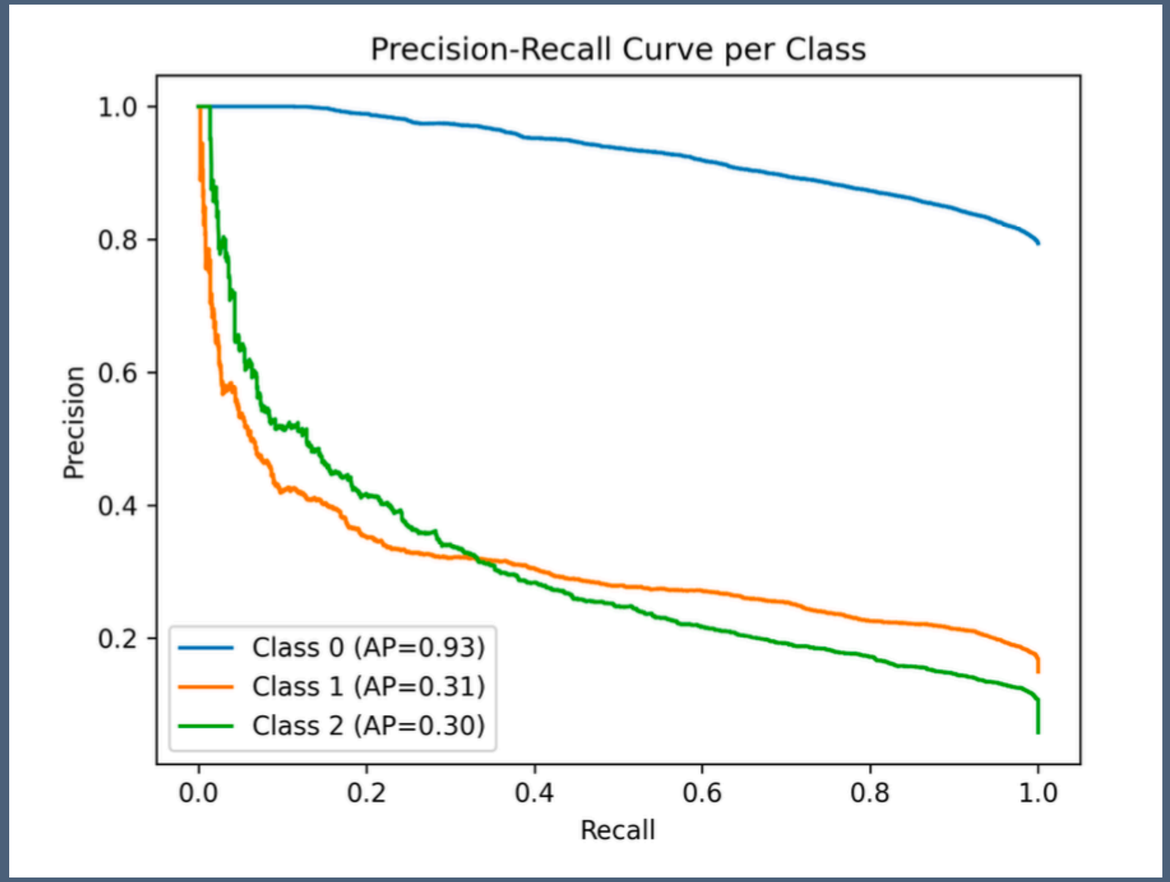
- Validation performance plateaus beyond 30,000 samples. The gap between training and validation curves suggests mild overfitting, but performance remains stable and does not degrade with additional data.
- While training performance is consistently higher, validation scores plateau smoothly, suggesting good generalization.



Detailed Evaluation

Class-wise Precision-Recall Analysis

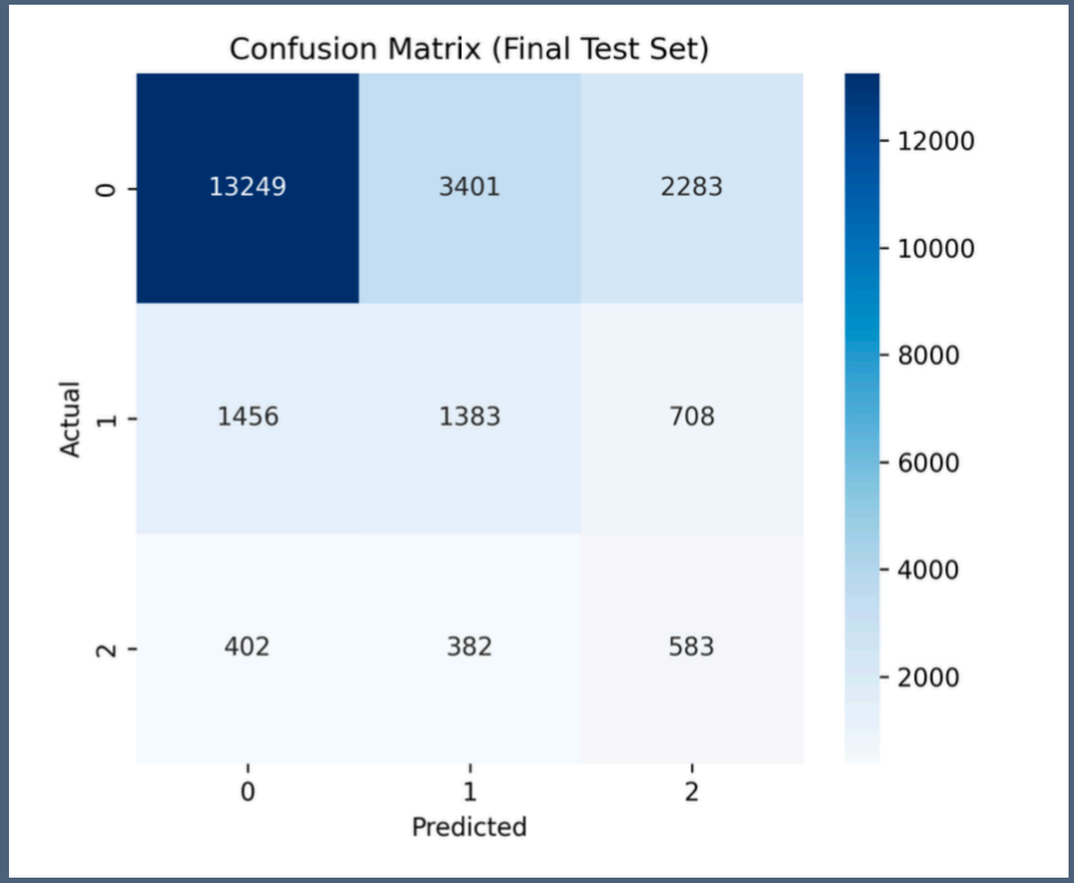
- Because of the class imbalance in the dataset, precision-recall (PR) curves offer a clearer picture of model behavior than accuracy alone.
- Class 0 has very high average precision (0.93), while classes 1 and 2 achieve lower values



Detailed Evaluation

Confusion Matrix Analysis

- Class 0 is correctly predicted in most cases
- Confusion remains between Class 1 and Class 2, which are more difficult to separate possibly due to overlapping patterns in historical severity and weather conditions.



Limitations & Challenges

- Short data collection window limits seasonal variability insights
- Spatial interactions not modeled (road segments treated independently)
- External context missing (events, roadworks, incidents could further boost accuracy)

Ethical & Operational Considerations

- Ethical importance: accurate prediction of severe congestion critical for public safety.
- Model transparency and interpretability crucial for building trust with stakeholders.
- Real-world impact: Reliable forecasting enhances proactive traffic management and congestion mitigation.

Future Directions

- Integration of incident/event data for improved predictive context
- Advanced spatial modeling with Graph Neural Networks (dependent on richer data)
- Deployment feasibility: real-time model adaptation and scalability
- Ensuring fairness in deployment (geographical and temporal fairness considerations)

Conclusion

- A hybrid machine learning approach significantly outperformed simpler baselines.
- The final Random Forest model achieved a strong balance between accuracy, fairness, and interpretability.
- Integrating historical severity probabilities with contextual and engineered features proved essential.
- The findings have practical value for real-time traffic monitoring and urban mobility planning.

Q&A

Questions & Discussion

GitHub: `github.com/theol-10/datascience-thesis`