**Les Cahiers du GERAD** ISSN: 0711–2440

**Comptes rendus du 15e atelier de résolution de problèmes industriels de Montréal, 2–6 juin 2025**

**Proceedings of the 15th Montréal Industrial Problem Solving Workshop, June 2–6, 2025**

Jean-François Plante,
Helen Samara Dos Santos, éditeurs

G–2025–72

Octobre 2025
October 2025



La publication de ces rapports de recherche est rendue possible grâce au soutien de HEC Montréal, Polytechnique Montréal, Université McGill, Université du Québec à Montréal, ainsi que du Fonds de recherche du Québec – Nature et technologies.

Dépôt légal – Bibliothèque et Archives nationales du Québec, 2025
– Bibliothèque et Archives Canada, 2025

The publication of these research reports is made possible thanks to the support of HEC Montréal, Polytechnique Montréal, McGill University, Université du Québec à Montréal, as well as the Fonds de recherche du Québec – Nature et technologies.

Legal deposit – Bibliothèque et Archives nationales du Québec, 2025
– Library and Archives Canada, 2025

**GERAD** HEC Montréal
3000, chemin de la Côte-Sainte-Catherine
Montréal (Québec) Canada H3T 2A7

**Tél. : 514 340-6053**
Téléc. : 514 340-5665
info@gerad.ca
www.gerad.ca

# Préface

Le CRM, IVADO et le GERAD organisèrent conjointement le Quinzième atelier de résolution de problèmes industriels de Montréal, qui eut lieu du 2 au 6 juin 2025 à HEC Montréal. Nous tenons à remercier les partenaires qui ont fourni des problèmes, en particulier Air Canada et la Banque Nationale du Canada, ainsi que les professeurs qui ont accepté de coordonner les travaux des équipes : Janosch Ortmann, Ting-Huei Chen, Rafael Cruz et Gilles Caporossi. Nous soulignons également la contribution des employées du CRM et du GERAD, qui se sont occupées avec diligence de l'organisation matérielle de l'atelier et de la mise en forme de ces comptes rendus : Flore Lubin, Marie Perreault, Marilyne Lavoie et Karine Hébert.

Helen Samara Dos Santos (Memorial University of Newfoundland)
Jean-François Plante (HEC Montréal, conseiller spécial aux partenariats du CRM)

CENTRE DE RECHERCHES MATHÉMATIQUES
IVADO
GERAD
GROUPE D'ÉTUDES ET DE RECHERCHE EN ANALYSE DES DÉCISIONS

## Foreword

The CRM, IVADO, and GERAD organized jointly the Fifteenth Montreal Industrial Problem Solving Workshop, which was held at HEC Montréal on June 2-6, 2025. We wish to thank our faithful partners, particularly Air Canada and the National Bank of Canada, for submitting problems to the workshop. We are also grateful to the professors who coordinated the work of the teams, namely Janosch Ortmann, Ting-Huei Chen, Rafael Cruz, and Gilles Caporossi. The employees of the CRM and GERAD, especially Flore Lubin, Marie Perreault, and Marilyne Lavoie ensured that the workshop ran smoothly and Karine Hébert put the proceedings in the "Cahier du GERAD" format.

Helen Samara Dos Santos (Memorial University of Newfoundland)
Jean-François Plante (HEC Montréal and Special Advisor, Partnerships, CRM)

CENTRE DE RECHERCHES MATHÉMATIQUES
IVADO
GERAD
GROUPE D'ÉTUDES ET DE RECHERCHE EN ANALYSE DES DÉCISIONS

# Contents

# 1 Air Canada Cargo: Modeling freight transportation no-shows

**Samah Abdalrhaman** [a, d]
**Ali Barooni** [b, d]
**Milad Barooni** [b, d]
**Paul Glickman** [c]
**María Camila Gómez López** [a, d]
**Serly Ishkhanian** [c]
**Shamim Mahmoudzadeh Vaziri** [b, d]
**Hervé Riboulet** [e]
**Salma Ben Ameur** [e]
**Janosch Ortmann, coordinator** [a, d, f]

[a] *UQAM*
[b] *Polytechnique Montréal*
[c] *Concordia University*
[d] *GERAD*
[e] *Air Canada*
[f] *CRM*

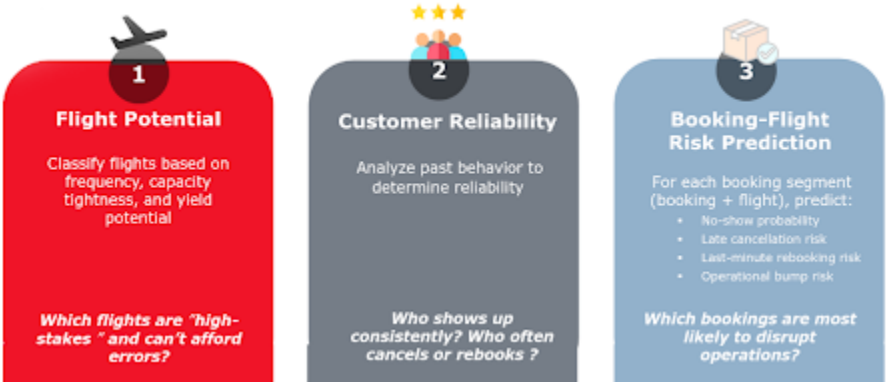
**October 2025**
**Les Cahiers du GERAD**

**Abstract:** *This report summarizes the findings of the team working on Problem 1 of the 2025 CRM Industrial Problem Solving Workshop (IPSW), with Air Canada as the industry partner. In this project, we addressed how to assist the Air Canada Cargo capacity management team in prioritizing flights and customers, given that cargo bookings can be cancelled or rebooked up until departure. We propose predictive and prescriptive models to identify critical routes and customers, as well as decision rules for Air Canada.*

## 1.1 Introduction

Air Canada Cargo (ACC) handles a large volume of shipments every day. However, the cargo business is subject to many uncertainties. For example, most customers in this industry are companies that need to ship large quantities several times a month or year, depending on their own customers' demands. There is no firm commitment from these customers to send their reserved shipments at the scheduled flight time, and Air Canada does not collect payment in advance. As a result, there is always a risk of last-minute changes — customers may rebook or cancel just one day before departure, or sometimes fail to show up at all without prior notice.

Given strong competitors in the cargo industry, revenue lost due to flights not operating at full capacity, and the increasing risk of customer churn, it is crucial for ACC to better understand and analyze customer behaviour and flight and route characteristics. The goal is to maximize profit, increase the rate of active loyal customers, and accurately predict the customer disruption rate (the rate of late rebookings, late cancellations, and no-shows) in order to minimize losses caused by unused capacity.Therefore, the purpose of this project is to support the capacity management team in prioritizing flights and customers more effectively by combining profiling and prediction at the booking-flight level.




We discuss the available data in Section 1.2. In this project, we have focused on the first two parts of the challenge: Flight Potential, discussed in Section 1.3, and Customer Reliability, discussed in Section 1.4. We conclude in Section 1.5.

## 1.2 Data presentation

ACC has provided several large and well-structured data sets. Each data set has two versions: the *Curve*, which includes all the parameter changes up to departure, and the *Performance* data set, which describes the state of affairs when the aircraft doors have closed. Our analysis has mostly focused on two data sets: Flight and Customer Station OD (Origin-Destination).

### 1.2.1 The Flight data sets

The Flight Curve data set tracks daily changes in available capacity, booking status, cancellations, and other flight details, allowing us to monitor occupancy trends and booking behaviour. The Flight Performance data set includes key operational metrics such as flown cargo, load factor, and show/no-show rates. This is essential for assessing the efficiency and reliability of different flights and routes.

### 1.2.2 The Customer Station OD data sets

This data set details the customer behaviour per origin-destination. In the Customer Station OD Curve version, every booking, cancellation, and rebooking is indicated, while the Customer Station OD Performance data set reports each parameter at the point of departure.

## 1.3 Flights and routes

In our analysis, we focused only on flights originating from Canada. We further limited the scope to flights with substantial cargo capacity, that is, those operated by freighter or wide-body aircraft.

Since the objective was to identify the critical flights, where a mistake in capacity planning has the greatest effect, we first studied the route network and then developed a *criticality score* $\Sigma$ for each flight. We then developed machine learning models to predict $\Sigma$.

### 1.3.1 Route network

Among the data we considered, there were about 82 000 international flights, slightly outnumbering domestic flights of which there were just over 70 000.

We visualized the filtered flight network in two maps. The first, Figure 1.1, displays the domestic routes, showing connections between Canadian cities. These routes are mainly concentrated around the three Air Canada hubs, Toronto (YYZ), Vancouver (YVR), and Montreal (YUL), forming a dense network of domestic cargo movement.

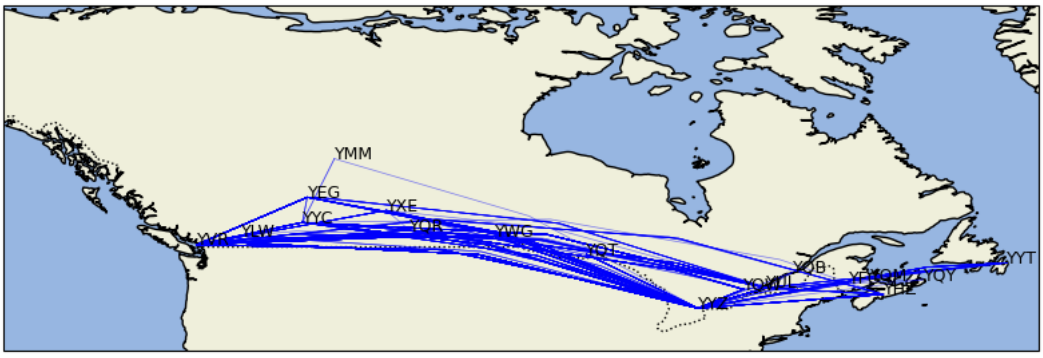



**Figure 1.1: Map of domestic routes with meaningful cargo capacity**

The second, Figure 1.2, illustrates the international routes, highlighting Air Canada Cargo's global connectivity. From the Air Canada hubs, cargo is transported to destinations across North America, Europe, Asia, South America, and Oceania.

### 1.3.2 Critical flights

To identify and prioritize the most critical flights in the network, we defined a Criticality Score, combining three main factors:

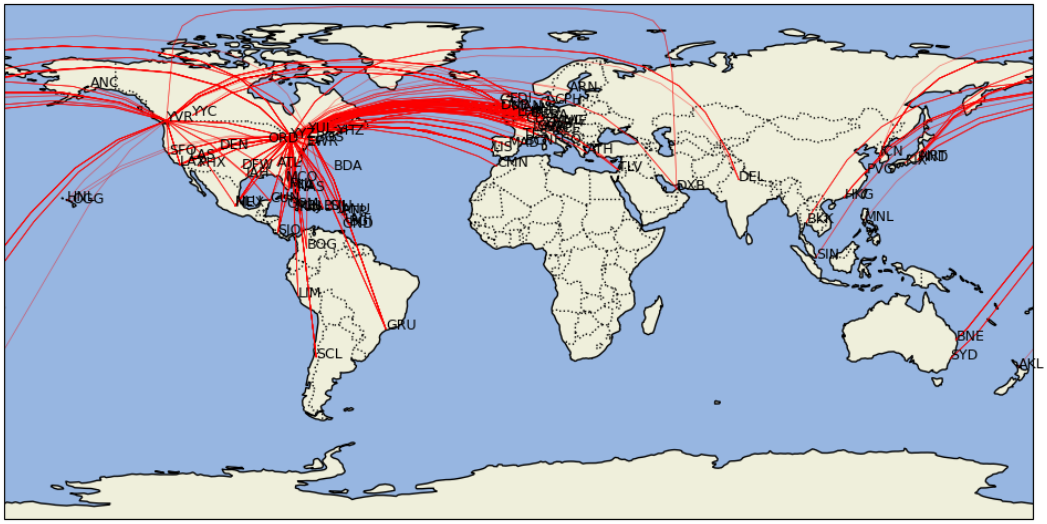



**Figure 1.2: Map of international routes with meaningful cargo capacity**

1. **Yield** – the revenue efficiency per unit of flown weight, denoted $Y$.
2. **Load Factor** – the ratio of cargo flown to total available capacity, denoted $L$.
3. **Flight Frequency** – how often a particular route operates over time, denoted $F$.

It is important to note that the yield variable is an estimate only. Yield is difficult to define precisely since it is not obvious how to distribute cost and revenue across customers, products, legs, and markets. A more careful analysis of the yield would lead to a better estimate of a flight's criticality. For simplicity, we imposed a linear relationship between each of these factors and the criticality score $\Sigma$. This leads to the formula

$$\Sigma = \alpha Y + \beta L + \gamma F, \tag{1.1}$$

for positive weights $\alpha, \beta, \gamma$. Each component is explained and visualized below to support the rationale behind the formulation. A large part of our analysis regarding load factor and yield is not presented in this report. This is due to the sensitive nature of this data for Air Canada.

**Seasonal fluctuations.** To analyze seasonal variation in network coverage, we visualized the flight routes (LEGOD) for each season separately. The four figures below illustrate the international reach of the cargo network during spring, summer, fall, and winter. These maps reveal changes in operational coverage across seasons. For instance, certain OD pairs – such as Santiago de Chile (SCL) and Auckland (AKL) – appear only in specific seasons, which suggests a temporal focus in cargo deployment. This variability is critical for interpreting fluctuations in load factor and planning future capacity allocation. See Figures 1.3 and 1.4.

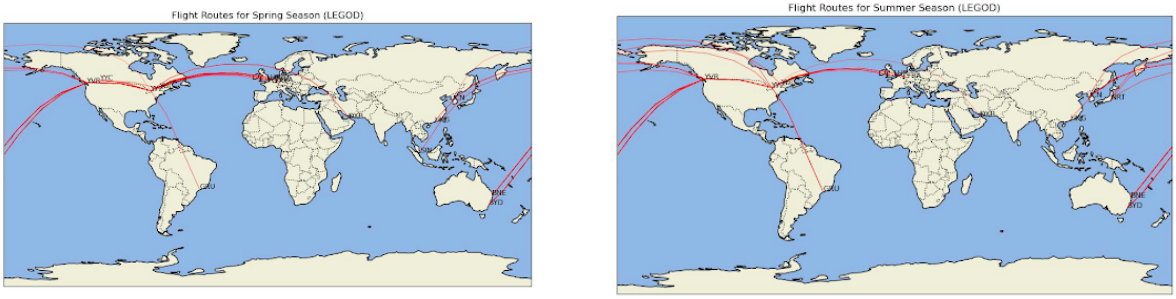



**Figure 1.3: Spring and summer seasonal route network**

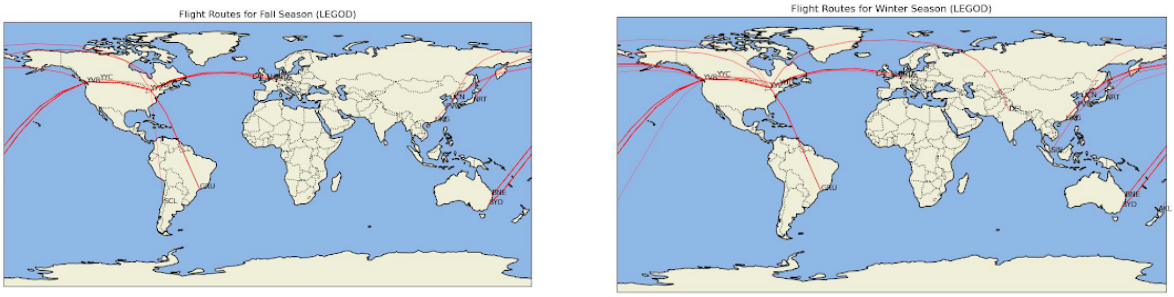



**Figure 1.4: Autumn and winter seasonal route network**

### 1.3.3 Predicting the critical factor

To identify critical flights or routes, we next applied machine learning (ML) techniques. A key challenge is the variable number of data points per flight in the Flight Curve database. For example, some flights have 15 records, others up to 35. This is because data collection started at the time of the first booking. This variability necessitates feature engineering to construct a consistent feature space, enabling the use of models that require fixed-size inputs. In the literature, this challenge is commonly referred to as the *variable-length time series problem*.

Each time series in the data set can provide valuable insights into various aspects of the flights. For example, one of the key features is the number of days remaining before the flight's departure. Additionally, several series capture details about the current reservation, such as the number of different cargo types on board, including fresh cargo, animals, e-commerce items, and more. Other series track operational metrics, such as the rates of standby, cancelled, or late-cancelled reservations. Most of these series show an increasing trend, as the number of reservations and cargo generally grows closer to the flight's departure time.

#### Feature engineering

One approach to the variable-length time series problem is to extract a fixed set of statistical features from each time series. This allows the resulting data set to be compatible with machine learning models that require a consistent input size. We have chosen the following features:

**Min, Max, Mean, Std, Range:** Basic statistical descriptors of the time series values.

**First Day, Last Day:** The value of the feature on the first and last recorded day.

**Slope (Trend):** The overall linear trend of the series, calculated by finding the slope of the series from the first day to the last day.

**Non-zero Count:** Number of time points with a non-zero value.

**Positive/Negative Steps Count:** Count of time steps where the series increased or decreased compared to the previous day.

**Exponentially Weighted Moving Average (EWMA):** A weighted average that gives more importance to recent values; used to capture recent trends. If the time series in question is $(x_t)_{t=0}^{n-1}$, the EWMA is the quantity $A_{\text{EWM}}$ where

$$A_{\text{EWM}} = \frac{\sum_{t=0}^{n-1}(1-\alpha)^t x_{n-1-t}}{\sum_{t=0}^{n-1}(1-\alpha)^t}, \tag{1.2}$$

where $\alpha$ is a factor that determines how much more weight is given for more recent values than those observed further in the past.

In this way, each time series was converted into a fixed-length vector of 12 features. We then used these fixed-length features as input for our model. The next subsection explains the model that we used.

### ML model: Gradient Boosted Trees (GBT)

For training and evaluation, we used gradient boosted trees (GBTs) [2] because it has several advantages that are key to our prediction problem. This model is well-known for capturing non-linear relationships between features, being robust to irrelevant features, performing a limited form of feature selection, and handling different types of input features.

GBTs, such as XGBoost, LightGBM [4], and CatBoost [6], are among the top performers in many real-world prediction tasks and, in some cases, may even outperform deep learning [1]. In the case of time series prediction, gradient-boosted models are among the most widely used tools by researchers worldwide, often outperforming other models [5].

In the gradient boosted model, the prediction is derived by taking the average or sum of a set $\mathcal{F} = \{f_1, \ldots, f_K\}$ of various weak learners, as described by the following formula. The weak learners might be simple trees, simple regression models, or other weak models. In most modern boosting libraries, the decision tree is the most commonly used base learner.

$$\widehat{y}_j = \phi\left(x_j\right) = \sum_{r=1}^{K} f_r\left(x_j\right) \tag{1.3}$$

At each step, the model's simple function is built or selected from a larger set of functions that may significantly reduce prediction error. To do this, a loss function is defined and minimized at each step. The following formula shows this loss function at the $t^{\text{th}}$ iteration.

$$\mathcal{L}^{(t)} = \sum_{j=1}^{n} \ell\left(y_j, \widehat{y}_j^{(t-1)} + f_t\left(x_j\right)\right) + \Omega\left(f_t\right) \tag{1.4}$$

where $\Omega(f_t)$ is a measure of the complexity of the model.

### Experimental results

In this subsection, we will discuss the experimental setup and the settings we used, along with the results for the prediction task. To implement our prediction after the feature engineering section, we used the XGBoost library and Python API.

We attempted both a regression and a classification task.

In the **regression task**, the target variable was the load factor itself. Based on input from Air Canada regarding their business need, we fed the model the series data until four days before departure. As is standard in supervised ML, the data set was split into a training and a test set. We report the results in Table 1.1 below. Importantly, performances on the training and test sets are compatible, which rules out overfitting.

**Table 1.1: Performance of the regression model**

| Metric | Training set performance | Test set performance |
|---|---|---|
| Mean absolute error (MAE) | 13.15 | 14.86 |
| Mean Absolute Percentage Error (MAPE) | 13.12 | 15.81 |
| Median Absolute Error | 10.74 | 11.81 |
| Root Mean Square Error (RMSE) | 16.85 | 19.19 |

One important benefit of using gradient-boosted models is their interpretability. Figure 1.5 shows the relative importance of each input feature. Perhaps unsurprisingly, the most important feature for predicting load factor at departure is the current load factor four days prior to departure. The next most important feature appears to be related to rebooking.

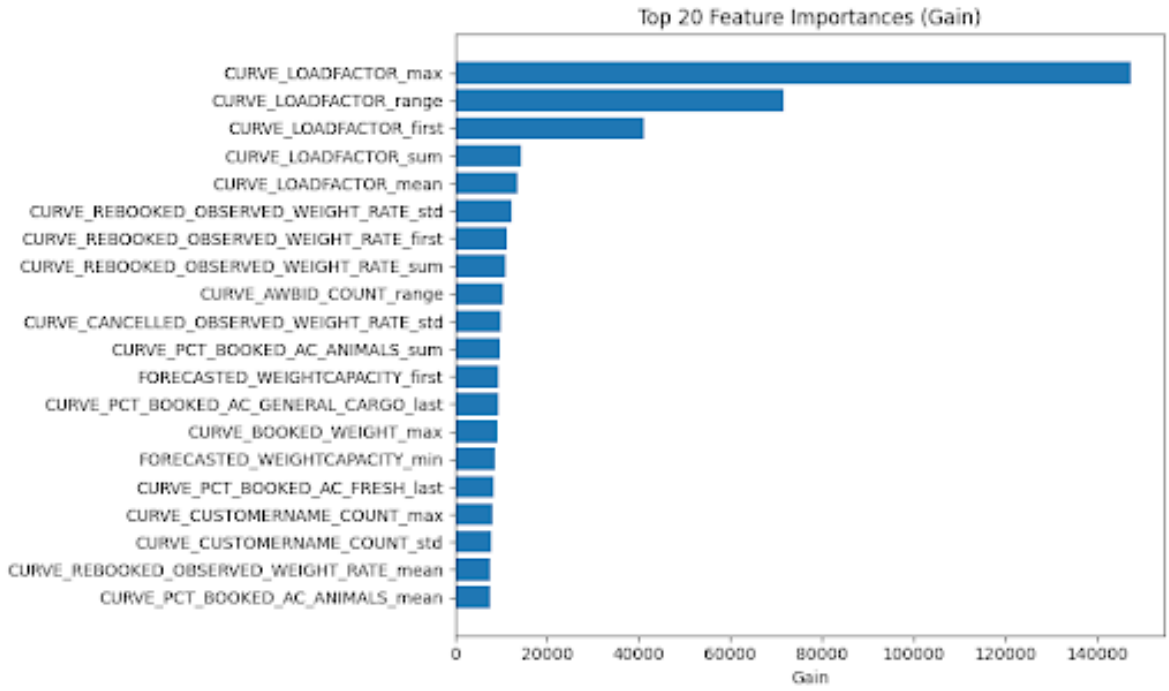



**Figure 1.5: Relative variable importance for the regression model**

Next, we attempted a **classification task** where the target variable was an indicator of whether the flight in question is critical or not. We set a threshold for the actual load factor on the day of departure. If the load factor was greater than that number, then we labelled the flight as critical.

This led to a problem of highly imbalanced classes, since only about 3% of flights were labelled as critical. Nevertheless, we were able to train a model with an accuracy of 0.69 and an ROC AUC of 0.83. Further details regarding the performance of the classification model are shown in the classification report, Table 1.2, and the confusion matrix, Figure 1.6.

**Table 1.2: Performance of the classification task**

| Class | Precision | Recall | F1 |
|---|---|---|---|
| Non-critical | 0.99 | 0.68 | 0.81 |
| Critical | 0.13 | 0.83 | 0.23 |

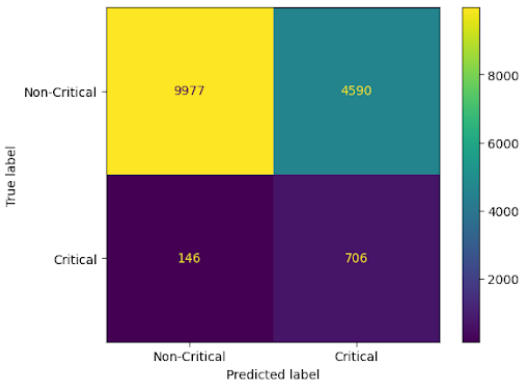



**Figure 1.6: Confusion matrix of the classification model**

## 1.4 Shipments and customers

The second team analyzed which customers or particular shipments were particularly at risk for cancellations or even late cancellations. We began by creating customer profiles to gain an understanding

of general customer behaviour on different routes. ACC internally designates high-value or high-volume shippers as Summit-tier customers. In what follows, we will focus on Summit customers, who represent approximately 37% of the annual weight.

We were first interested in understanding Air Canada Cargo's top customers and studied the top 10% by annual flown weight in 2023 and 2024. These are shown (anonymized) in Figure 1.7. Each of the top 3 customers has shipped over 150 tons in a year, reaching a maximum of 300 tons. The top three clients overall were consistent across the two years. In fact, they all flew slightly more in 2024 than in 2023. Some of the other major clients in 2023 did not fly as much in 2024.

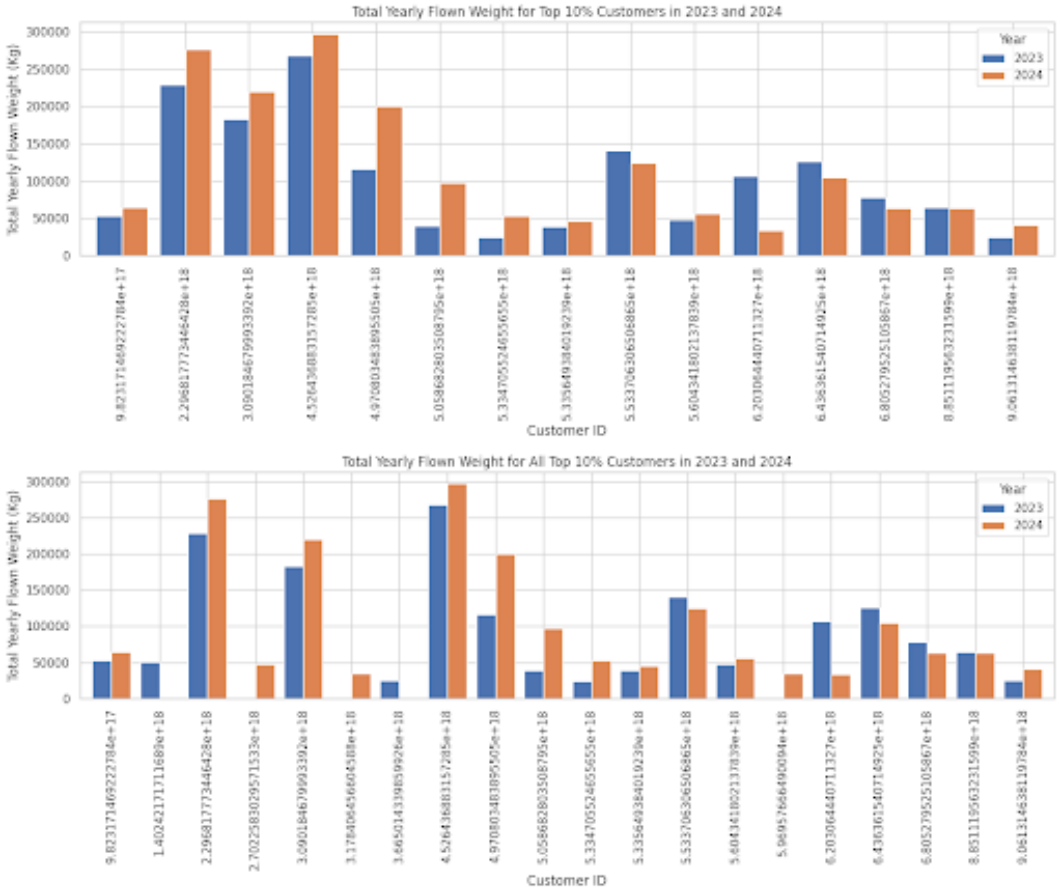



**Figure 1.7: Top 10% ACC customers by annual flown weight: top 10% in 2023 (above) compared across 2023 and 2024, and top 10% in either 2023 or 2024 (below)**

Next, we analyzed the top routes by flown weight in 2024. Figure 1.8 shows the top 20 high-demand routes in 2024 by the total flown weight on that route. The five busiest routes were Toronto to Atlanta (YYZ → ATL), Vancouver to Toronto (YVR → YYZ), Tokyo to Toronto (HND → YYZ), Montréal to Toronto (YUL → YYZ), and Lima to Toronto (LIM → YYZ).

The customers who can impact ACC are those who cancel late, rebook late, or fail to show up (flight conditions should also be considered). For this section, we will consider only the behaviour of customers for each route.

Customers with the biggest impact on operations are those who frequently cancel late, rebook late, or fail to show up. In order to study this impact, we first developed a *disruption score*.

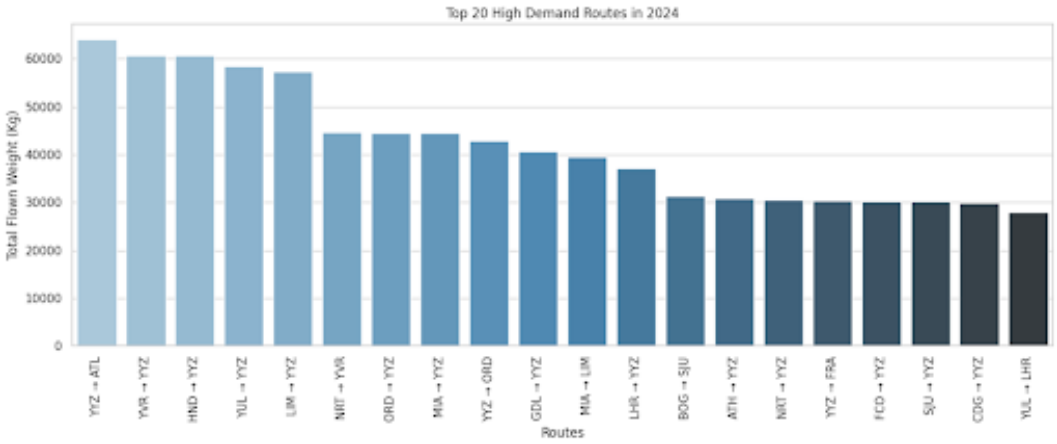



**Figure 1.8: Top routes by weight flown in 2024**

### 1.4.1 The customer disruption score

We first computed a new variable, *disruption weight* $W_D$, for each customer, which is the sum of the weights of late cancellations, late rebookings, and no-shows. It is crucial to consider a customer's portion of the disrupted weight with respect to the initial total booked weight. For instance, if a customer books heavily and makes a late cancellation of 75% of the booking at the last minute, this causes a substantial loss for Air Canada. To assess customer reliability, we assigned a risk level to each customer. This risk level was based on the average risk score of a customer per route.

Since each route is unique, booking limits are set per customer, and the risk score was calculated separately for each route. Once we obtained a score for each booking made on that route, we needed to assign a single score for each customer. Therefore, we first calculated the risk score based on the following equation. Let $S_B$, $m_B$, and $M_B$ denote the total, minimal, and maximal bookings per customer and define the *normalized booking sum* by $S_{NB}$ where

$$S_{NB} = \frac{S_B - m_B}{M_B - m_B} \tag{1.5}$$

We defined a *risk score* $\sigma_R$ for each customer to be $S_{NB}$ multiplied by the weighted average of the no-show rate (weighted 0.5), the late cancellation rate (weighted 0.3), and the late rebooking rate (weighted 0.2). On the other hand, we defined the *risk ratio* $\rho_R$ via

$$\rho_R = \frac{W_D}{W_\Sigma}, \tag{1.6}$$

where $W_\Sigma$ denotes the total booked weight of that customer. Our *hybrid risk score* was computed as the product of the risk score $\sigma_R$ and the risk ratio $\rho_R$. We then assigned each customer one of three groups:

**Very Risky:** Customers whose average risk score lies in the top 20% of all customers with a non-zero disruption weight.

**Risky:** Customers whose average risk score lies between the $20^{\text{th}}$ and $50^{\text{th}}$ percentiles.

**Low Risk:** Customers whose average risk score lies in the bottom 50%.

This classification can help the capacity team prioritize customers and resources based on risk level. Figures 1.9 and 1.10 show the risk groups by summit level and route.

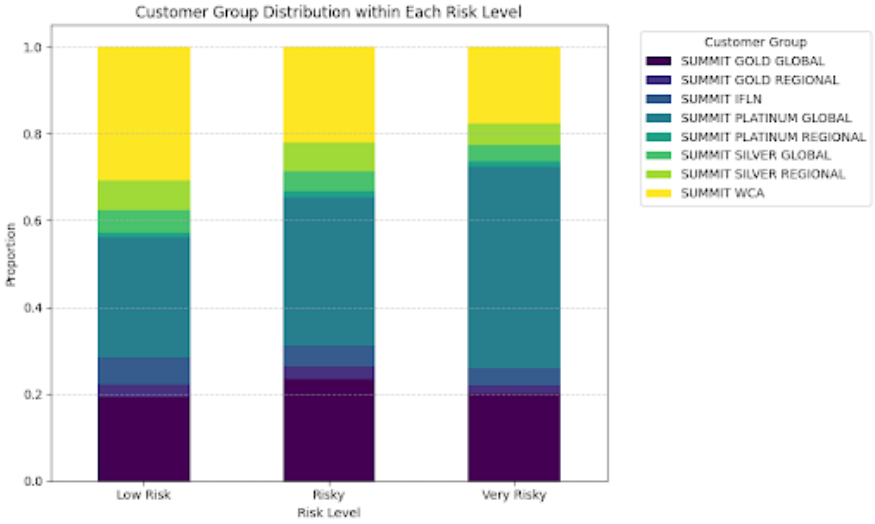



**Figure 1.9: Distribution of Summit level by risk group**

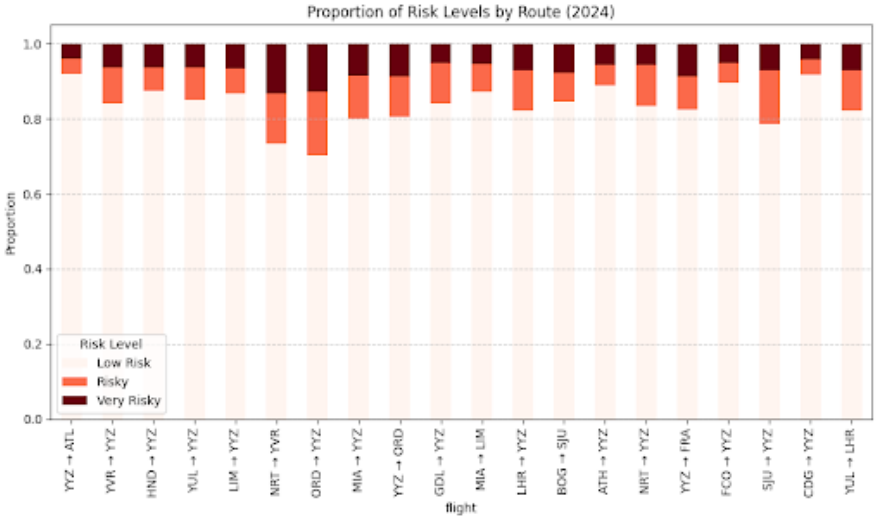



**Figure 1.10: Distribution of risk levels by route**

### 1.4.2 Overbooking rate

To minimize the risk of excessive unused space on flights, we introduced a variable to set the *overbooking rate*. This involved predicting the expected disruption weight (i.e., the weight from accepted customers who would cancel late, rebook late, or fail to show) so we could adjust with additional bookings. For example, we might accept shipments up to 1.2 times the actual capacity, knowing that some customers would not follow through.

Since we handle various types of cargo, including live animals and fresh goods, the overbooking rate can fluctuate from month to month. In Figure 1.11, we present monthly performance metrics such as the late rebooking rate, late cancellation rate, and total booked rate to identify operational or environmental factors that explain fluctuations in customer demand. This helped us define an appropriate overbooking rate for each month.

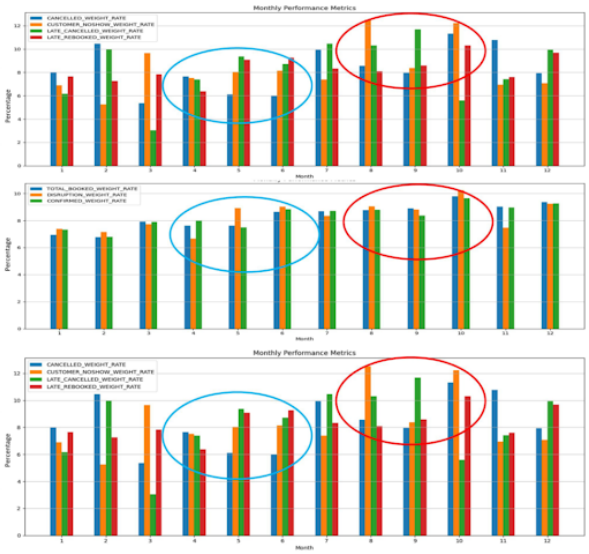



**Figure 1.11: Performance metrics for the overbooking rate**

### 1.4.3 Optimization model

The objective of the customer shipment selection analysis is to minimize the total disruption weight for each flight. Specifically, we aimed to determine the optimal combination of shipment requests to accept for each flight operating from an origin O to a destination D, subject to the flight's total weight capacity. The decision was based on multiple factors, including each customer's hybrid risk score, historical disruption behaviour, and the weight of the requested shipment.

We introduced the following mathematical formulation: suppose that we have $n$ shipment requests (or customers) associated with a specific OD flight, and $m$ OD routes with $z$ OD flights. We have the following parameters:

- The hybrid risk score $\rho_{ij}^H$ for customer $i$ on route $j$
- The requested shipment weight $w_{ij}$ of customer $i$ for route $j$
- The predicted disruption rate $d_{ij}^P$ of customer $i$ for route $j$
- The capacity on route $j$
- The overweight rate $o_j$ for the flight on route $j$

The decision variables of the model are $y_{ij}$, which take the value 1 if the request by customer $i$ is accepted on route $j$ and 0 otherwise. We obtained the following optimization model:

$$\min \sum_{j=1}^{m} \sum_{i=1}^{n} \rho_{ij}^H w_{ij} d_{ij}^P y_{ij} \tag{1.7}$$

$$\sum_{i=1}^{n} w_{ij} \left(1 - d_{ij}^P\right) y_{ij} \leq c_j \qquad \forall\, j \in [m] \tag{1.8}$$

$$\text{s.t.} \sum_{i=1}^{n} w_{ij} y_{ij} \leq (1 + o_j)\, c_j \qquad \forall\, j \in [m] \tag{1.9}$$

$$\sum_{i=1}^{n} w_{ij} y_{ij} \geq 0.9 c_j \qquad \forall\, j \in [m] \tag{1.10}$$

$$y_{ij} \in \{0, 1\} \qquad \forall\, i \in [n], \forall\, j \in [m].$$

In equation (1.8), we ensure that the total weight of accepted shipment requests, after subtracting the predicted disruption weight, does not exceed the flight's capacity. Equation (1.9) enforces the constraint that the total weight of accepted requests must not exceed the sum of the flight's capacity and the predicted overweight allowance for that flight, while we restrict the maximum allowable empty space on a flight to 10% of its capacity in (1.10).

To solve this optimization problem, we first introduced a Long Short-Term Memory (LSTM) model [3] to predict two key parameters based on historical data:

- The disruption rate for each customer on a given origin-destination (OD) route and
- The overweight rate associated with each OD route flight.

Subsequently, we modelled each flight as an agent in a *deep reinforcement learning* (DRL) framework. The state representation for each agent (flight) included the following information:

- The total available flight capacity,
- The weight of each shipment request,
- The predicted disruption rate per customer,
- The predicted overweight risk for the flight, and
- The hybrid risk score associated with the OD route.

The action space consists of binary decisions for each shipment request: accept (1) or reject (0). The output of the DRL model is therefore a binary action vector indicating which shipment requests should be accepted in order to minimize overall disruption weight while respecting flight capacity constraints. The reward function is defined as the negative weighted sum of the predicted disruption weights for all accepted shipment requests. Specifically, for each accepted request, the negative predicted disruption weight is multiplied by the corresponding hybrid risk score, thereby penalizing the model for decisions that may lead to late cancellations, no-shows, or overweight scenarios. This formulation ensures the selection of low-risk, reliable shipments. See Figure 1.12.

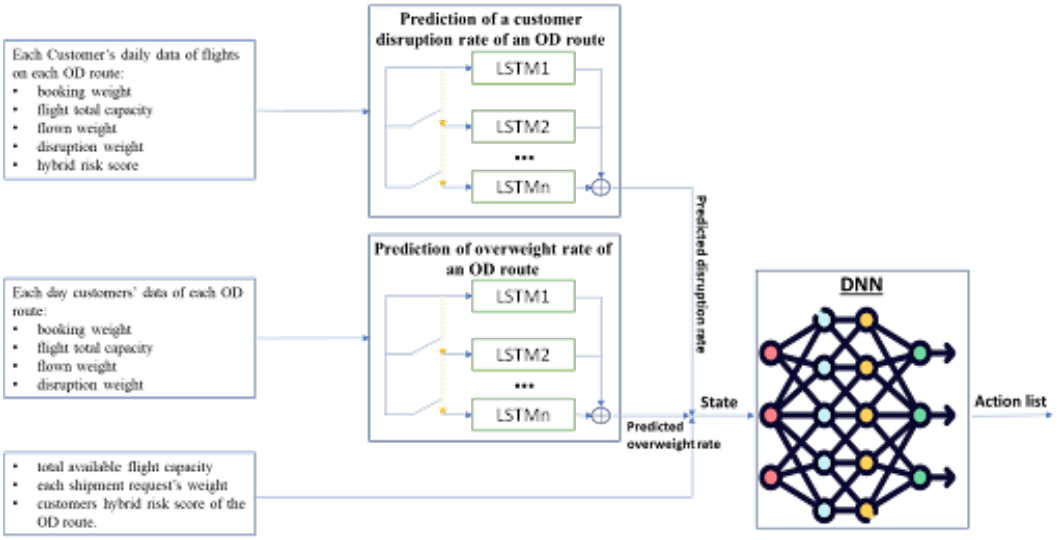



**Figure 1.12: Model overview for each future flight, each of which is considered an agent**

.

### 1.4.4 Future work

As a next step, we propose combining both the flight-level prioritization and customer-level risk assessment into a unified mathematical model. The objective would be to maximize total value by

giving preference to flights with high criticality and low risk, as represented in the expression

$$\max \sum_{f \in \text{flights}} \left( \text{CriticalityScore}_f \left( 1 - \text{RiskProportion}_f \right) \right) \tag{1.11}$$

We propose to train and apply the model to our data set in order to evaluate its ability to minimize the disruption weight associated with each origin–destination (OD) flight while optimally assigning customers to flights. The model will incorporate prioritization based on the latest delivery deadlines and the frequency of available flights for each OD route. Additionally, we aim to integrate the revenue associated with each customer request into the optimization process, with the objective of minimizing total disruption weight while concurrently maximizing Air Canada's revenue per flight.

Given the ability of complex machine learning models such as GBTs to capture complex relationships between the features and the target variable, it would also be interesting to train a model directly on the entire data set, without first creating summary features. This would, however, imply a serious risk of overfitting and would require a more careful analysis.

## 1.5 Conclusion

During the week of the Industrial Problem Solving Workshop, we proposed several predictive and prescriptive models for identifying critical routes and customers. These models proved promising but require further investigation.

## Bibliography

[1] Vadim Borisov, Tobias Leemann, Kathrin Sesßler, Johannes Haug, Martin Pawelczyk, and Gjergji Kasneci. Deep neural networks and tabular data: A survey. IEEE Transactions on Neural Networks and Learning Systems, 35(6):7499–7511, 2024.

[2] Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), pages 785–794. ACM, 2016.

[3] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. Neural Computation, 9(8):1735–1780, 1997.

[4] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. LightGBM: A highly efficient gradient boosting decision tree. In Proceedings of the 31st Conference on Neural Information Processing Systems (NeurIPS), pages 3146–3157, Long Beach, CA, USA, 2017. Curran Associates, Inc.

[5] A. D. Lainder and R. D. Wolfinger. Forecasting with gradient boosted trees: augmentation, tuning, and cross-validation strategies. International Journal of Forecasting, 38(4):1426–1433, 2022.

[6] Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. Catboost: Unbiased boosting with categorical features. In Advances in Neural Information Processing Systems (NeurIPS), volume 31, Montréal, Canada, 2018.

# 2 Air Canada Cargo: Image recognition for verifying cargo anchor compliance

**Mina Arzaghi** [a, d]
**Myrine Barreiro-Arevalo** [b]
**Alireza Farashah** [c, e]
**Faramarz Farhangian** [f]
**Michael R. Lindstrom** [b]
**Rafael M. O. Cruz, coordinator** [f]
**Paul Sanyang** [g]
**Joel Williams** [b]

[a] *HEC Montréal*
[b] *University of Texas Rio Grande Valley*
[c] *McGill University*
[d] *GERAD*
[e] *Mila*
[f] *ETS*
[g] *Concordia University*

**October 2025**
**Les Cahiers du GERAD**

**Abstract:** *Optional abstract: This report presents a computer-vision approach to automate cargo lock verification for Air Canada Cargo. We first evaluate zero-shot detection with OWL-ViT and then develop a supervised pipeline built on single-shot object detectors, supported by super-resolution preprocessing, data augmentation, and hyperparameter tuning. Our results demonstrate strong performance on a limited dataset, and the analysis indicates that improved data quality and diversity would further strengthen model reliability.*

## 2.1 Introduction

Air Canada has been a globally renowned airline and Canada's largest air carrier for over 86 years. As one of the biggest employers in Canada and a proud member of the Star Alliance, which includes 25 other carriers, they enable their passengers to reach hundreds of destinations across numerous countries. In addition to passenger transport, Air Canada Cargo plays a vital role in global logistics, moving thousands of containers and pallets daily with high standards of safety and reliability.

Aircraft cargo operations demand security checks to ensure that containers and pallets remain securely fastened throughout flight. Traditionally, ground personnel inspect each cargo lock visually and photograph the setup for manual verification. However, variable lighting, cramped spaces, and occlusions often make it difficult to determine whether a lock is fully engaged. Moreover, locks usually appear in clusters at random positions across the frame, making manual review slow and error-prone. Our goal is to automate this step with a computer-vision system that works reliably on just a few dozen labelled examples.

In this report, we present an end-to-end framework for automating cargo lock verification using object detection and classification techniques. We begin by evaluating *zero-shot detection* with OWL-ViT (Open-World Localization Vision Transformer) [14], demonstrating the strengths and limitations of foundation models guided by natural-language prompts. To overcome the shortcomings of zero-shot approaches on fine-grained latch states, we design a *supervised pipeline* built on single-shot object detectors (e.g., YOLO family [17]), enhanced by three key components: (1) super-resolution preprocessing to improve image quality, (2) extensive data augmentation to compensate for the small dataset size, and (3) meta-heuristic hyperparameter tuning through genetic algorithms.

Our study reveals several key insights for deploying computer vision in this setting:

- While zero-shot detection models show promise for general object recognition, they struggle with fine-grained state classification tasks, even with careful prompt engineering.
- Supervised object detectors require substantial preprocessing and data augmentation to overcome the limitations of small, low-quality datasets; super-resolution techniques and synthetic data expansion prove essential for achieving reliable performance.
- Hyperparameter optimization plays a critical role in maximizing model accuracy when training data is scarce, with evolutionary methods outperforming default configurations.

This work demonstrates that it is possible to build an effective cargo lock verification system even under constrained data and computing resources. While our pipeline achieved strong results, the study revealed that data quality is the main limiting factor. Future deployments should prioritize structured data collection to obtain even stronger models for this task.

## 2.2 Object detection

Object detection involves identifying and localizing instances of objects belonging to predefined classes within an image. The field has evolved significantly from early methods that relied on handcrafted features such as Haar cascades or HOG descriptors [5, 23] to the modern paradigm dominated by deep convolutional neural networks (CNNs). Today's detectors produce a list of predictions $\{(b_i, c_i, s_i)\}_{i=1}^{N}$, where $b_i$ is the bounding box, $c_i$ is the class label, and $s_i$ is the confidence score. The training process aims to minimize a composite loss function that penalizes errors in both object localization and classification [17].

In this work, we investigate two strategies to detect cargo lock states with minimal labelled data. First, we explore *zero-shot detection* using the OWL-ViT foundation model, which leverages vision–language pretraining to recognize and localize new object categories without task-specific retraining [14]. Second, we employ a *supervised fine-tuning* approach based on a single-stage detector—YOLO—which directly regresses bounding boxes and class probabilities in one pass [17]. Object detectors are generally divided into two architectural families: two-stage methods (e.g., Faster R-CNN) that generate region proposals before classification and refinement [18], and single-stage methods, such as YOLO, that dispense with the proposal step for faster processing. We choose YOLO for the supervised path because its efficiency allows rapid iteration on data augmentations, super-resolution preprocessing, and hyperparameter tuning under our resource constraints.

### 2.2.1 Zero-shot object detection with the OWL-ViT

Zero-shot object detection (ZSD) addresses the critical limitation of conventional detectors, which require exhaustive annotated examples for every target class. By leveraging semantic relationships between seen and unseen categories, ZSD models recognize objects that were *never encountered during training* through transferable visual-semantic alignment. This capability is particularly valuable in specialized domains, such as industrial inspection, where annotation scarcity is common. Formally, given a set of seen classes $\mathcal{C}_s$ with labelled instances $\mathcal{D}_s = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N}$ and unseen classes $\mathcal{C}_u$ ($\mathcal{C}_s \cap \mathcal{C}_u = \emptyset$), a zero-shot detector learns a mapping $\phi : \mathcal{X} \times \mathcal{C} \to \mathbb{R}^4 \times \mathbb{R}$ that localizes and classifies objects from both $\mathcal{C}_s$ and $\mathcal{C}_u$ without exposure to $\mathcal{C}_u$ during training. Here, $\mathcal{X}$ denotes the input space (images), and $\mathcal{C} = \mathcal{C}_s \cup \mathcal{C}_u$ is the set of all classes. The dataset $\mathcal{D}_s$ consists of labelled examples with $\mathbf{x}_i \in \mathcal{X}$ and $\mathbf{y}_i \in \mathcal{C}_s$.

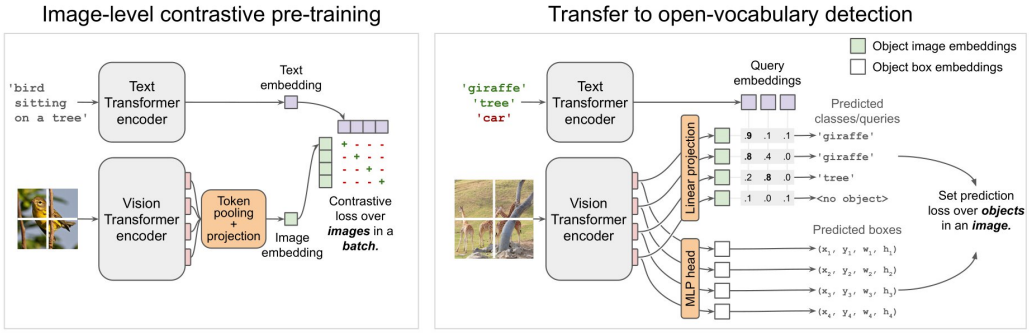



**Figure 2.1: The OWL-ViT (short for Vision Transformer for Open-World Localization)**

OWL (Object-World Learning) [14] proposes unifying vision-language pretraining with detection-specific optimization. OWL builds upon the CLIP framework [16] but introduces critical architectural innovations for spatial reasoning. As shown in Figure 2.1, the model comprises: (i) a Vision Transformer (ViT) image encoder that outputs spatial feature maps $\mathbf{F} \in \mathbb{R}^{h \times w \times d}$; (ii) a text encoder projecting class names to embeddings $\mathbf{T} \in \mathbb{R}^{|\mathcal{C}| \times d}$; and (iii) a detection head that predicts bounding boxes conditioned

on text-aligned visual features. This design enables open-vocabulary detection by comparing image region embeddings against text prototypes through cosine similarity:

$$s_{i,j} = \frac{\mathbf{F}_i \cdot \mathbf{T}_j}{\|\mathbf{F}_i\|\|\mathbf{T}_j\|} \tag{2.1}$$

where $s_{i,j}$ represents the detection score for region $i$ and class $j$.

OWL first initializes using CLIP's vision–language knowledge, then fine-tunes with detection-specific objectives using the loss :

$$\mathcal{L}_{\text{OWL}} = \lambda_1 \mathcal{L}_{\text{contrast}} + \lambda_2 \mathcal{L}_{\text{loc}} + \lambda_3 \mathcal{L}_{\text{cls}} \tag{2.2}$$

The contrastive term $\mathcal{L}_{\text{contrast}}$ enforces alignment between image regions and text descriptions, $\mathcal{L}_{\text{loc}}$ optimizes box coordinates via GIoU loss [19], and $\mathcal{L}_{\text{cls}}$ refines class predictions using focal loss [12]. OWL-ViT's main advantage is its ability to perform open-vocabulary detection—localizing and classifying categories defined only by text prompts—without any task-specific labelling. Since zero-shot alone was insufficient, we next turned to fully supervised, single-stage detectors, specifically the YOLO family.

### 2.2.2 Fine-tuned detectors

#### The YOLO Architecture

While the original YOLO model [17] introduced the core concept of a single-pass grid-based prediction, its architecture has evolved substantially. Modern YOLO variants such as YOLOv5 and YOLOv8 follow a standardized three-part structure, as illustrated in Figure 2.2.

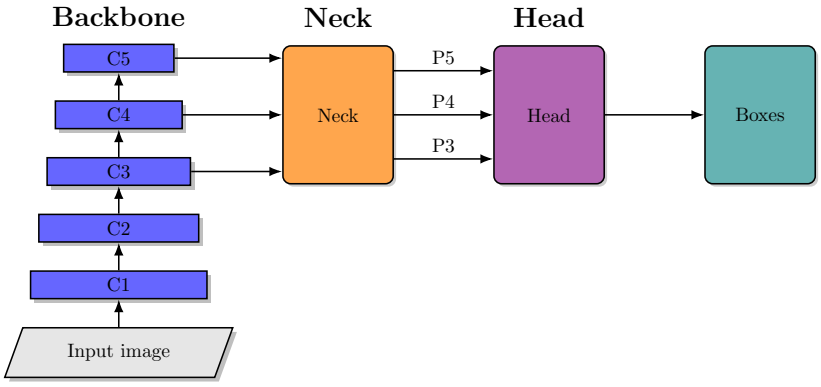



**Figure 2.2: The general architecture of a modern YOLO model with its main components**

**Backbone:** This is a deep CNN (e.g., ResNet [8]) that serves as the primary feature extractor. Such backbones are typically pre-trained on a large-scale classification dataset, such as ImageNet [6], and subsequently fine-tuned on a specific object detection dataset, such as MS-COCO [13]. This transfer learning strategy is crucial for achieving high accuracy, as the pretraining on millions of diverse images makes the model learn general-purpose features. When this pre-trained model is fine-tuned on a smaller dataset, it does not have to learn these features from scratch.

**Neck:** The neck connects the backbone to the prediction heads. Its purpose is to aggregate and fuse features from different stages of the backbone. Modern YOLO models typically employ structures such as a Path Aggregation Network (PANet), which combines features from both top-down and bottom-up pathways to create feature maps rich in both semantic and spatial information.

**Head:** The heads are responsible for the final predictions. Modern YOLO models are multi-scale, using separate heads to make predictions on feature maps of different resolutions. This enables the model to detect small, medium, and large objects effectively. Figure 2.3 shows the process of multiple bounding box suggestions and the final detection.

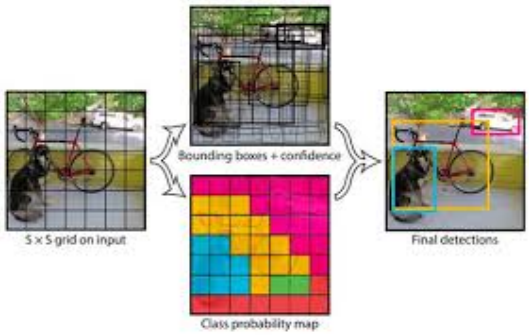



**Figure 2.3: Bounding box predictions (red) are refined from anchor priors (dashed) or from centre-point offsets (anchor-free variants), with class probabilities assigned per grid cell**

### Evolution of YOLO Variants

The YOLO family has undergone rapid evolution, with each new version introducing architectural improvements and/or training objectives. In this work, we consider both YOLOv5 and YOLOv8, based on the implementations from the Ultralytics package.[1]

In addition, modern YOLO releases typically include several models of varying sizes (e.g., small, medium, large) to provide a trade-off between speed, memory footprint, and accuracy. Table 2.1 compares the specifications for the small, medium, and large versions of YOLOv5 and YOLOv8, the two primary models considered in this work. For this project, we utilized the small ('s') versions of the models. This decision was driven primarily by the significant time and computational resource constraints.

**Table 2.1: Comparison of model size and computational cost for YOLOv5 and YOLOv8 variants. Parameters are in millions (M), and GFLOPs are calculated at an input resolution of 640x640**

| Family | Variant | Parameters (M) | GFLOPs |
|---|---|---|---|
| YOLOv5 | Small (s) | 7.2 | 16.5 |
| | Medium (m) | 21.2 | 49.0 |
| | Large (l) | 46.5 | 109.1 |
| YOLOv8 | Small (s) | 11.2 | 28.6 |
| | Medium (m) | 25.9 | 78.9 |
| | Large (l) | 43.6 | 165.2 |

### Bounding Box Prediction

A key architectural choice in object detectors is the strategy used for bounding box prediction. Early and highly successful models like Faster R-CNN [18] and YOLOv5 [10] employed an **anchor-based** approach. This method relies on a set of predefined bounding boxes (named "anchors") that have scales and aspect ratios learned from the object dimensions present in the training dataset. The indicators are tiled across the image, and the network predicts offsets to refine their position and size, along with a class probability for each.

More recent models, including YOLOv8, have transitioned to an **anchor-free** design. Instead of refining predefined boxes, these methods predict bounding boxes directly. A common technique is to predict the object's center point within a grid cell and then regress the distances from that center to the four sides of the bounding box. This approach simplifies the model design by removing the need for hand-picked anchor configurations and has been shown to achieve state-of-the-art performance.

[1] https://www.ultralytics.com

### Loss Function

The loss function used in our YOLO is denoted by equation 2.3 and comprises three terms:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{\text{box}} + \lambda_2 \mathcal{L}_{\text{cls}} + \lambda_3 \mathcal{L}_{\text{dfl}} \tag{2.3}$$

**1. Localization Loss ($\mathcal{L}_{\text{box}}$)**
This term ensures accurate bounding box predictions by optimizing their coordinates and dimensions. Modern YOLO variants use **Complete IoU (CIoU)** loss, which considers overlap, center distance, and aspect ratio:

$$\mathcal{L}_{\text{box}} = 1 - \text{IoU} + \frac{\rho^2(\mathbf{b}, \mathbf{b}_{\text{gt}})}{c^2} + \alpha v \tag{2.4}$$

where IoU is the Intersection over Union; $\rho$ is the Euclidean distance between predicted ($\mathbf{b}$) and ground truth ($\mathbf{b}_{\text{gt}}$) box centers, $c$ is the diagonal length of the smallest enclosing box, $v$ measures aspect ratio consistency and $\alpha = \frac{v}{(1-\text{IoU})+v}$.

**2. Classification Loss ($\mathcal{L}_{\text{cls}}$)**
This term penalizes misclassifications using **Focal Loss** [12] in order to handle class imbalance:

$$\mathcal{L}_{\text{cls}} = -\sum_{i=1}^{N} \left[ y_i \cdot (1 - p_i)^{\gamma} \log(p_i) + (1 - y_i) \cdot p_i^{\gamma} \log(1 - p_i) \right] \tag{2.5}$$

where $y_i$ is the ground truth label (0 or 1), $p_i$ is the predicted probability for class $i$, and $\gamma$ modulates the loss for hard-to-classify examples [12].

**3. Distribution Focal Loss ($\mathcal{L}_{\text{dfl}}$)**
Introduced in anchor-free models like YOLOv8, DFL treats bounding box regression as a probability distribution over discrete locations. It optimizes the predicted distribution $\mathbf{P} = [P(y_0), \ldots, P(y_n)]$ to match the target $y_{\text{gt}}$:

$$\mathcal{L}_{\text{dfl}} = -\sum_{i=1}^{N} \left( (y_{i+1} - y_{\text{gt}}) \log(P(y_i)) + (y_{\text{gt}} - y_i) \log(P(y_{i+1})) \right) \tag{2.6}$$

where $y_i$ and $y_{i+1}$ are the nearest discrete values to $y_{\text{gt}}$.

## 2.3 Hyperparameter tuning with genetic algorithms

Hyperparameter optimization is a fundamental challenge in machine learning, where model performance critically depends on the careful configuration of training hyperparameters that govern learning dynamics and generalization [2]. In particular, optimizing YOLO's performance requires careful tuning of approximately 30 hyperparameters that control training behaviour, regularization and data preprocessing hyperparameters. These parameters form a high-dimensional search space where exhaustive exploration is computationally infeasible due to exponential complexity. The optimization challenge is further compounded by the black-box nature of the objective function, which can only be evaluated through expensive training and validation cycles that may take hours per configuration.

This combination of high dimensionality and expensive function evaluation renders traditional methods inadequate: Grid search scales poorly beyond a few parameters, while random search lacks mechanisms to leverage past evaluations. Genetic Algorithms (GAs) [9] address these limitations by efficiently exploring the search space through population-based sampling, making them particularly suitable for complex black-box optimization, where we must strategically select which configurations

to evaluate next in order to optimize results with fewer calls to the expensive black-box function.We implemented GA optimization using both Ultralytics' built-in '.tune()' method and a custom solution for finer control over genetic operations.

### 2.3.1 Genetic Algorithms (GA)

Genetic Algorithms (GAs) are a class of population-based, stochastic optimization heuristics inspired by the principles of Darwinian evolution and natural selection [9]. Instead of evaluating a single point in the search space, a GA maintains a *population* of candidate solutions (called *individuals*). Each individual represents a complete set of hyperparameters, encoded as a *chromosome*. The population evolves over a series of *generations* through the application of genetic operators, gradually converging toward an optimal solution. The core iterative process is illustrated in Figure 2.4.

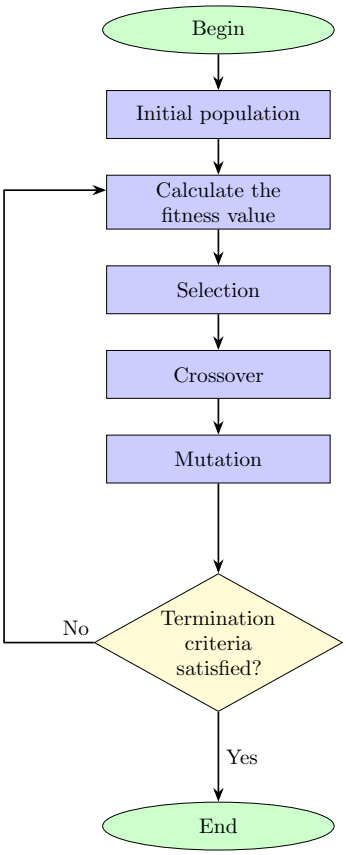



**Figure 2.4: Main steps in a genetic algorithm**

The process begins with an **initialization** step, where an initial population of individuals is generated, typically by sampling randomly from the valid range of each hyperparameter to ensure diversity. This is followed by **fitness evaluation**, where the *fitness* of each individual is determined by training a model with its corresponding hyperparameters and calculating the objective function (e.g., validation $\text{mAP}_{0.5:0.95}$). Subsequently, a **selection** mechanism probabilistically chooses individuals from the current population to become *parents*, giving preference to those with higher fitness.

The core of generating new solutions lies in **crossover (recombination)**, where selected parents are paired and their genetic material is combined to produce *offspring*, exploring new and potentially superior hyperparameter combinations. Finally, **mutation** introduces small, random alterations to the offspring's chromosomes, injecting genetic diversity to prevent premature convergence and enable exploration of new regions in the search space. This cycle of evaluation, selection, crossover, and

mutation is repeated for a fixed number of generations or until a predefined termination criterion is met (e.g., no change in the fitness function after $n$ iterations).

### 2.3.2 GA implementation for YOLO hyperparameter tuning

In our project, the Genetic Algorithm was adapted to optimize YOLOv8 hyperparameters using Ultralytics' mutation-focused approach. The implementation began with defining the **chromosome encoding** as a real-valued vector where each gene represented a hyperparameter: learning rate (`lr0`), momentum, weight decay, and loss coefficients (`box`, `cls`, `dfl`), constrained within predefined bounds (e.g., `lr0` $\in$ [`1e-5`, `1e-1`]). Crucially, Ultralytics' GA employs a **mutation-only strategy** without crossover operations, diverging from traditional GA designs [9].

For **mutation**, Gaussian perturbation was applied with:

$$\text{mutated\_value} = \text{original\_value} \times (1 + \mathcal{N}(0, \sigma)) \tag{2.7}$$

where $\sigma = 0.04$ controls mutation magnitude, and each gene has an 80% mutation probability per generation. Mutated values were clipped to maintain feasibility within parameter bounds. The **fitness function** used validation $\text{mAP}_{0.5:0.95}$ from full training runs, with top-performing individuals retained for subsequent mutations. **Termination** occurred after 300 generations (default) or upon fitness plateau computed over the validation set.

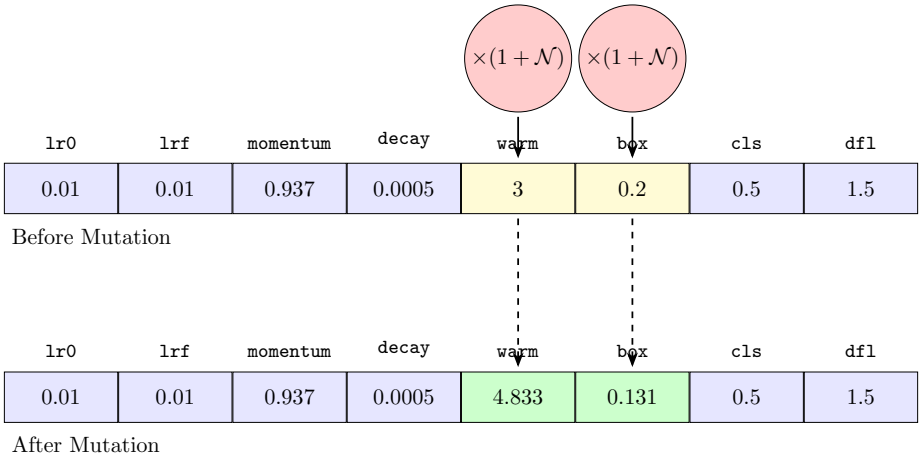



**Figure 2.5: Gaussian mutation operator applied to YOLO hyperparameters. Two hyperparameters (`warm` and `box`) are mutated, while the others remain unchanged. Each gene has an 80% mutation probability per generation. Shaded yellow = before mutation; shaded green = after mutation. Here, `lrf` = final LR multiplier, `warm` = warmup epochs (or steps), `decay` = `weight decay`, and $\mathcal{N} \sim \mathcal{N}(0, \sigma)$, where $\sigma = 0.04$**

## 2.4 Image super-resolution with Real-ESRGAN

In real-world industrial scenarios, image quality is frequently compromised due to motion blur, poor lighting, low sensor fidelity, or aggressive compression. This was particularly true for our dataset, which contained low-resolution images of aircraft cargo bays where the key features—such as latch markers—were often indistinct. Enhancing image resolution became a critical preprocessing step to improve downstream object detection performance.

To this end, we employed **Real-ESRGAN** [24], a state-of-the-art Single Image Super-Resolution (SISR) model designed to handle complex degradations typically found in real-world images. Unlike earlier deep learning-based super-resolution approaches, which focus on synthetic downsampling or ideal blur kernels, Real-ESRGAN utilizes a high-order degradation pipeline that combines blur, noise,

and JPEG artifacts to train on realistic image distortions. This enables it to reconstruct visually sharp textures with improved perceptual quality.

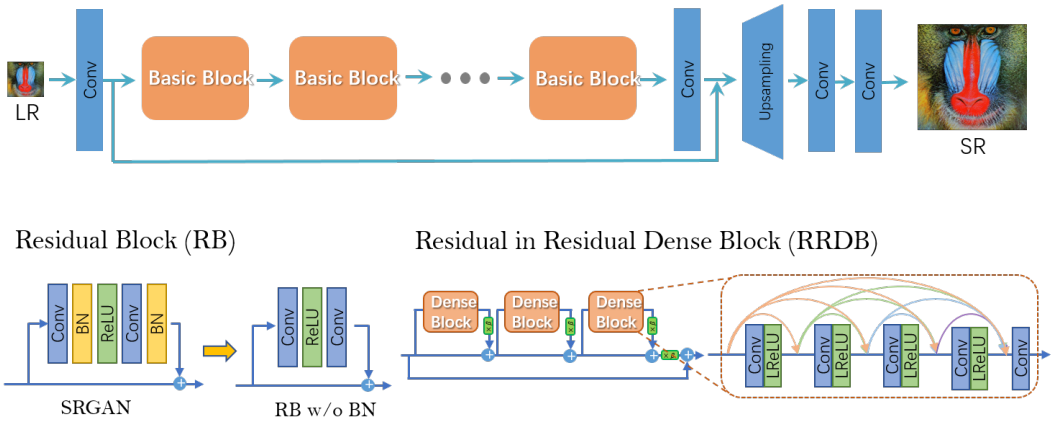



**Figure 2.6: Illustration of an ESRGAN-based architecture. Real-ESRGAN builds on this design with more robust training and high-order degradation modelling [24, 25]**

At its core, Real-ESRGAN integrates a deep generator network composed of Residual-in-Residual Dense Blocks (RRDB) [25] without batch normalization. The generator output is defined as:

$$\mathbf{f}_{l+1} = \mathbf{f}_l + \mathcal{R}(\mathbf{f}_l; \theta_{\mathcal{R}})$$

where $\mathcal{R}$ is the RRDB module parameterized by $\theta_{\mathcal{R}}$. The network depth reaches 23 convolutional layers, made trainable by the residual structure [8]. The discriminator follows a U-Net [20] structure with spectral normalization, allowing for both local and global adversarial feedback and improved gradient propagation.

The model is trained using a composite loss:

$$\mathcal{L} = \lambda_{\text{adv}}\mathcal{L}_{\text{adv}} + \lambda_{\text{perc}}\mathcal{L}_{\text{perc}} + \lambda_1\|\hat{\mathbf{y}} - \mathbf{y}\|_1 \tag{2.8}$$

where:

- $\mathcal{L}_{\text{adv}}$ is a relativistic adversarial loss that distinguishes real and generated images,
- $\mathcal{L}_{\text{perc}}$ is a perceptual loss computed over VGG-19 [21] features,
- $\|\hat{\mathbf{y}} - \mathbf{y}\|_1$ is an $L_1$ reconstruction loss.

Weights are typically set to $\lambda_{\text{adv}} = 1$, $\lambda_{\text{perc}} = 1$, and $\lambda_1 = 0.05$ to balance fidelity and realism.

### 2.4.1 Integration into our pipeline

Given the low visual quality of many cargo bay images, we used Real-ESRGAN [24] as a preprocessing step to enhance image resolution and sharpness before object detection. Before super-resolution, each image underwent denoising, sharpening via a Laplacian kernel, and histogram equalization on the luminance channel. We then applied the `RealESRGAN_x4plus` model (v0.3.0) with tiling to upscale images by a factor of four, resulting in final resolutions of $2560 \times 2560$ pixels. This step was conducted over the original training images before data augmentation, significantly improving the visual clarity of the latch contours and surface details, thereby enhancing model confidence during both training and inference.

## 2.5 Experiments

### 2.5.1 Data source

The dataset used in this project consists of 117 manually annotated images of cargo locks in aircraft containers. Each image was labelled with bounding boxes and classified as either “lock up” or “lock down.” All images were resized to $640 \times 640$ pixels to standardize input dimensions for model training. Annotation and preprocessing tasks, including orientation correction and YOLO-format export, were performed using the Roboflow platform. The images were split into 64 for training, 17 for validation, and 36 for testing.

Figures 2.7 and 2.8 illustrate two challenges in the dataset. While the distribution of lock states is balanced, the number of locks per image varies from two to over seven, introducing variability in object density. Additionally, the heatmap in Figure 2.8 shows that lock positions are scattered throughout the image space, requiring the model to generalize across diverse spatial arrangements despite the small dataset size.

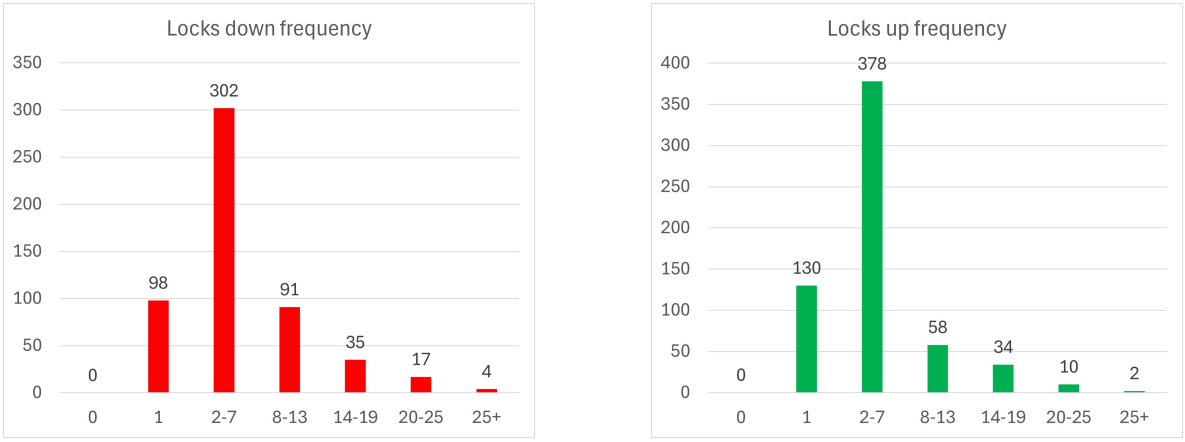



**Figure 2.7: Lock classes are evenly distributed, but the majority of images have 2–7 locks, which influences training when images have many locks**

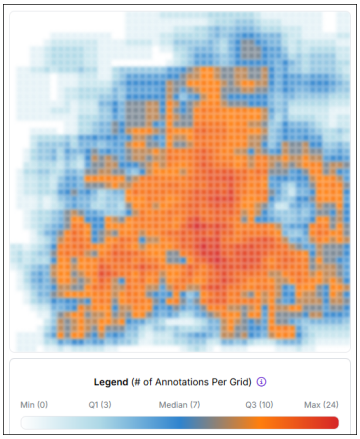



**Figure 2.8: Heatmap of lock positions across the dataset, showing that locks are scattered throughout images, making lock detection more difficult due to positional variability**

### 2.5.2 Performance metrics

To evaluate object detection performance, we use standard metrics: precision, recall, average precision (AP), and mean average precision (mAP). These are computed based on the overlap between predicted and ground-truth bounding boxes.

#### Intersection over Union (IoU)

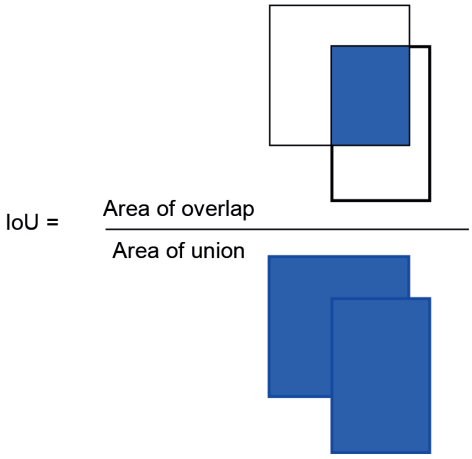



**Figure 2.9: Visual representation of the Intersection over Union (IoU) metric**

A predicted bounding box $B_p$ is considered a correct detection (True Positive) if its Intersection over Union (IoU) with a ground-truth box $B_{gt}$ exceeds a threshold $\tau$ (e.g., 0.5 or 0.75):

$$\text{IoU}(B_p, B_{gt}) = \frac{|B_p \cap B_{gt}|}{|B_p \cup B_{gt}|}.$$

#### Precision and Recall

Given predictions on a dataset, precision and recall are defined as:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}},$$

where TP, FP, and FN denote true positives, false positives, and false negatives, respectively.

#### Average Precision (AP)

Precision and recall vary depending on the detection confidence threshold. The average precision (AP) is computed as the area under the precision-recall curve:

$$\text{AP} = \int_0^1 \text{Precision}(\text{Recall})\, d\text{Recall}.$$

In practice, AP is approximated using interpolation at 101 recall levels (0.0 to 1.0), following the COCO evaluation protocol [13].

#### Mean Average Precision (mAP)

Mean average precision is the mean of AP values across multiple IoU thresholds or across multiple classes. In this work, we report:

- $\text{mAP}_{0.5}$: AP computed at a fixed IoU threshold of 0.5.
- $\text{mAP}_{0.5:0.95}$: Mean AP computed across 10 IoU thresholds from 0.5 to 0.95 in steps of 0.05.

### 2.5.3 Hyperparameter tuning

To identify the optimal YOLO configuration for our dataset, we applied genetic algorithms (GAs) for hyperparameter tuning. We compared two variants: the built-in `.tune()` method provided by Ultralytics and a custom implementation designed to explore a broader search space with the assistance of an LLM-based model. The fitness function was defined as the mean average precision (mAP) evaluated on the validation set.

The two methods converged to substantially different hyperparameter configurations, as shown in Tables 2.2 and 2.3. For this analysis, we considered YOLOv5 and YOLOv8 as object detection models. The **Ultralytics tuner** initializes from a strong default configuration and applies only **mutation** operations over successive generations. While computationally efficient, this strategy limits search space exploration. In contrast, **custom GA** starts from a randomly initialized population and employs both **crossover** and **mutation**, enabling a more diverse traversal of the hyperparameter space and leading to distinct optimal configurations.

**Table 2.2: Performances using .tune() and custom GA after tuning for mAP@0.5:0.95. YOLOv5 obtained the overall best performance during this tuning procedure. YOLOv8 did not converge in the specified time using the custom GA**

| version | batch | epoch | patience | optimizer | tune | custom GA |
|---|---|---|---|---|---|---|
| v5s | 8 | 10 | 2 | SGD | 0.22 | 0.18 |
| v8s | 16 | 20 | 50 | Adam | 0.069 | N/A |

**Table 2.3: GA-Tuned Hyperparameters for YOLOv5s**

| Hyperparameter | Ultralytics.tune() | Custom GA |
|---|---|---|
| lr0 | 0.01 | 0.01008 |
| lrf | 0.01 | 0.90903 |
| momentum | 0.937 | 0.79178 |
| weight_decay | 0.0005 | 0.00091 |
| warmup_epochs | 3 | 4.833 |
| box | 0.2 | 0.131 |
| cls | 0.5 | 3.488 |
| dfl | 1.5 | 1.151 |

YOLOv5 consistently outperformed YOLOv8 during our tuning experiments. While the exact reasons for this advantage would require further investigation, we hypothesize that YOLOv5's architecture may be better suited to small datasets, as evidenced by its more stable convergence during training. Based on these results, we selected YOLOv5 as our primary detector for all subsequent pipeline components.

### 2.5.4 Data augmentation

To enhance the robustness and generalization of our model, we employed a comprehensive data augmentation strategy. The augmentation pipeline was applied to both the standard and high-resolution training datasets to artificially expand the diversity of our training samples. For each image in the training set, we generated five augmented versions, each with a corresponding transformation of its bounding box annotations to ensure label accuracy. This process was facilitated by the Albumentations library [3], a standard tool for image augmentation in computer vision tasks.

The specific augmentations utilized were selected to simulate a range of real-world variations, as illustrated in Figure 2.10. These included a `HorizontalFlip` to introduce mirror-image perspectives. We also applied an Affine transformation, which combined scaling to 80% of the original size, a shear of 5 degrees, a 10% translation, and a rotation of 20 degrees, to account for changes in viewpoint and object position. To address variations in lighting and image quality, `RandomBrightnessContrast` was

used to adjust brightness and contrast, `GaussianBlur` was applied to simulate out-of-focus blurring, and `MultiplicativeNoise` was introduced to mimic sensor noise. These transformations were applied individually to each original image, ensuring a controlled and diverse set of augmented data for training.


(a) Affine (AT) (b) Brightness/Contrast (BC) (c) Gaussian Blur (GB)

(d) Horizontal Flip (HF) (e) Multiplicative Noise (MN)

**Figure 2.10: Examples of data augmentations applied to the cargo lock images. These simulate variations in orientation, lighting, motion blur, and sensor noise**

## 2.6 Results

In this section, we describe our experimental evaluation, organized as a sequence of progressively stronger baselines and enhancements. Each subsection title states the main conclusion drawn from the corresponding experiment.

### 2.6.1 Zero-shot detection cannot reliably distinguish lock states

We first evaluated OWL-ViT in a pure zero-shot setting, without any fine-tuning on our data. As shown in Table 2.4, the model achieves reasonable performance when asked simply to detect "cargo lock" in a class-agnostic way (precision 66.41%, recall 38.81%), but fails almost entirely on the fine-grained task of distinguishing "latch down" (recall 2.36%). This demonstrates that zero-shot detection alone is insufficient for our application.

#### Prompt engineering improves but does not fully close the gap

Adding more specific, domain-aware prompts substantially raises zero-shot recall and precision for detecting locks. For example, "a red ULD safety lock on an aircraft container" produced the best results (Figure 2.11. However, even the best prompt could not reliably detect the rare *latch down* class. These results suggest that relying solely on zero-shot models may be insufficient for this task.

**Table 2.4: OWL-ViT Zero-Shot Detection Results**

| Metric | Lock Down | Lock Up | All Locks |
|---|---|---|---|
| Total GT Boxes | 127 | 92 | 219 |
| True Positives (TP) | 3 | 66 | 85 |
| False Positives (FP) | 13 | 197 | 43 |
| False Negatives (FN) | 124 | 26 | 134 |
| Precision | 0.1875 | 0.2510 | 0.6641 |
| Recall | 0.0236 | 0.7174 | 0.3881 |

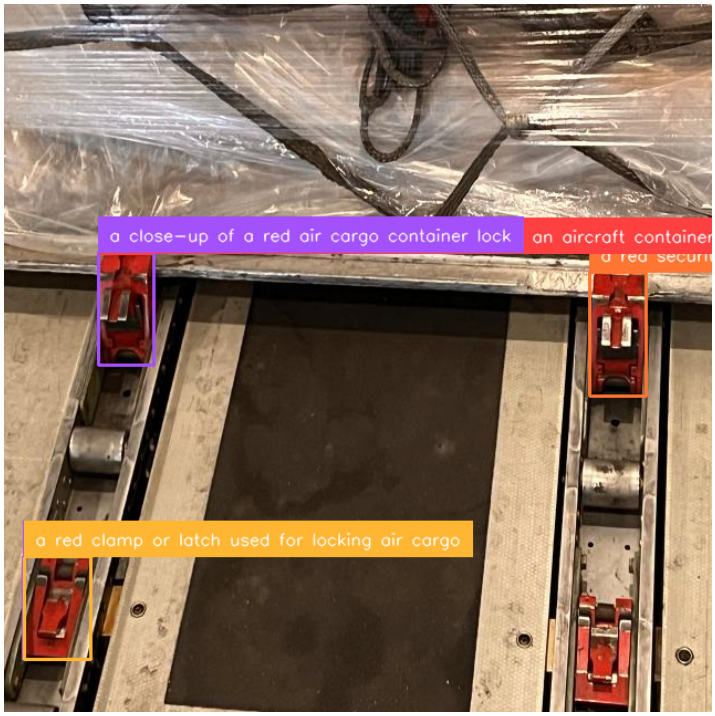



**Figure 2.11: Results after changing prompts to detect locks. Different descriptions enabled the model to detect various locks. However, it still misses one lock (bottom right)**

### 2.6.2 Fine-tuned model struggles to learn due to small and low-quality data

To establish a supervised baseline, we trained YOLOv5 directly on the original 117 Air Canada lock images –resized to $640 \times 640$ with no prior data augmentation or image enhancement. Table 2.5 presents the results obtained by the selected model over the test dataset. Despite using a model pre-trained on a large dataset, the obtained performance was below expectations. The low mAP@50:95 in particular indicates that many predicted boxes did not align tightly with ground truth, reflecting the challenges of learning from small, low-resolution inputs.

**Table 2.5: Baseline YOLOv5 performance on raw images**

| Metric | Value |
|---|---|
| mAP@50 | 0.60 |
| mAP@50:95 | 0.21 |
| mAP@75 | 0.10 |

This initial training, using a limited and low-quality dataset, presented significant challenges, as illustrated in Figure 2.12. While the training loss curves for bounding box, class, and DFL

(Distribution Focal Loss) show a consistent decrease, the corresponding validation loss curves are noisy. This considerable fluctuation in validation performance, particularly in the `val/box_loss` and `val/dfl_loss`, indicates that the model struggled to generalize from the images belonging to training data to unseen examples (validation set). The model was likely overfitting due to the small dataset.

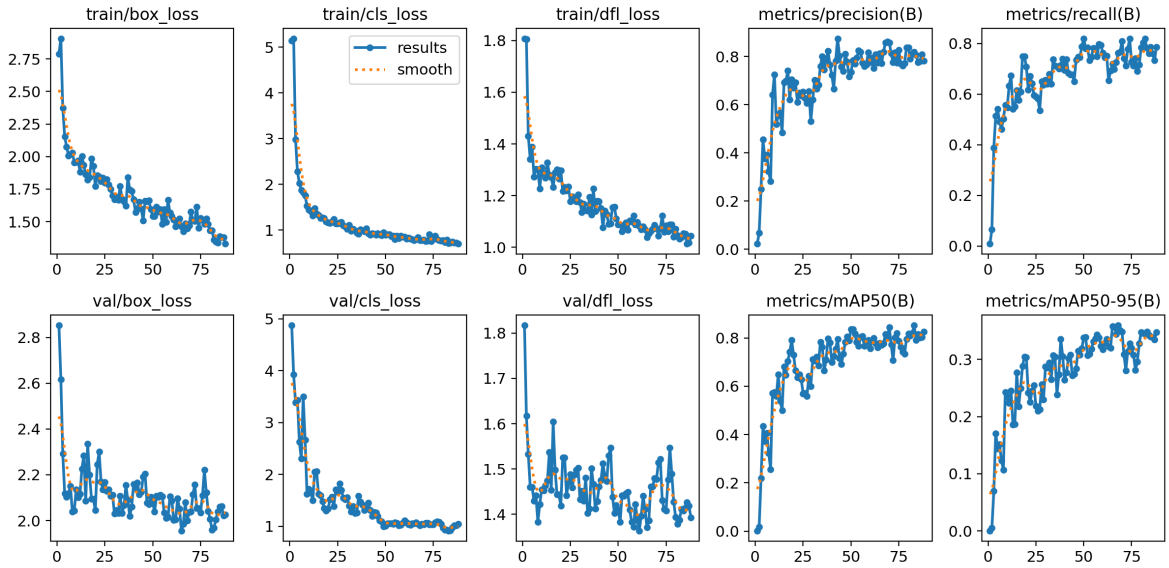
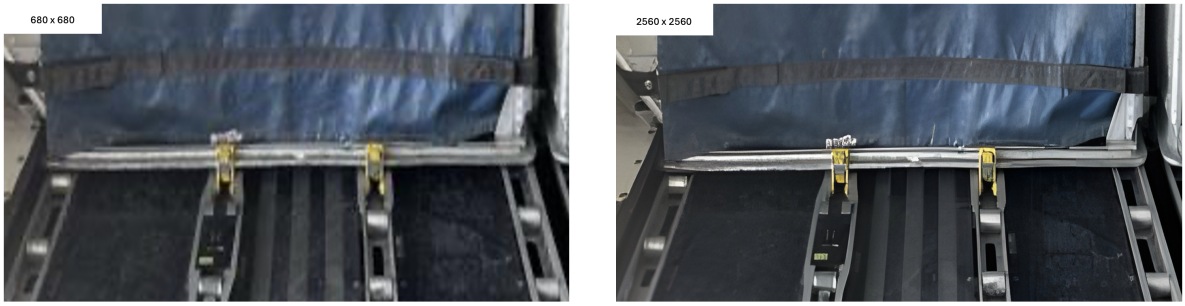



**Figure 2.12: Baseline YOLO training before applying the preprocessing pipeline**

### 2.6.3 Experimental results with enhanced and augmented data

To maximize model performance given our limited original dataset, we applied both **super-resolution** (Real-ESRGAN) and **data augmentation** (Albumentations) before YOLO training. These preprocessing steps were performed jointly, and all reported results reflect their combined effect. We do not isolate their individual contributions due to the lack of intermediate ablation stages.

Super-resolution was used to upscale all training images to $2560 \times 2560$ resolution, revealing sharper latch contours and richer details of the cargo structure. In parallel, data augmentation increased our training set fivefold through horizontal flips, affine transformations, brightness/contrast adjustments, Gaussian blur, and multiplicative noise. Figure 2.13 illustrates the visual improvement after enhancement.




**(a) Original image** **(b) Restored image (2560×2560)**

**Figure 2.13: Visual comparison of the image before and after super-resolution restoration**

Table 2.6 compares the model performance before and after applying super-resolution and data augmentation. The combined techniques yielded substantial improvements, with mAP@50:95 increasing from 0.2067 to 0.5936 and mAP@75 rising from 0.0969 to 0.5864. These gains demonstrate that image enhancement and dataset expansion were more impactful for our small dataset than architectural changes and hyperparameter tuning alone. Thus, this further suggests that more structured data collection should be considered in the future to attain a higher-performing model.

**Table 2.6: Model performance with raw data versus enhanced and augmented data**

| Metric | Raw | Enhanced + Augmented |
|---|---|---|
| mAP@50:95 | 0.2067 | 0.5936 (+0.3869) |
| mAP@50 | 0.5973 | 0.7776 (+0.1803) |
| mAP@75 | 0.0969 | 0.5864 (+0.4895) |

The impact of this approach is evident in Figure 2.14. The validation loss curves are now significantly more stable and closely mirror the downward trend of the training loss curves when compared to the version without data augmentation (Figure 2.12). This stabilization, especially noticeable in the `val/box_loss` and the dramatic reduction of the initial spike in `val/cls_loss`, demonstrates that the model is no longer overfitting to the training data. By training on a more diverse set of images, the model has learned more robust features, therefore leading to better generalization. These training dynamics further explain the performance gains achieved after enhancing the original training dataset.

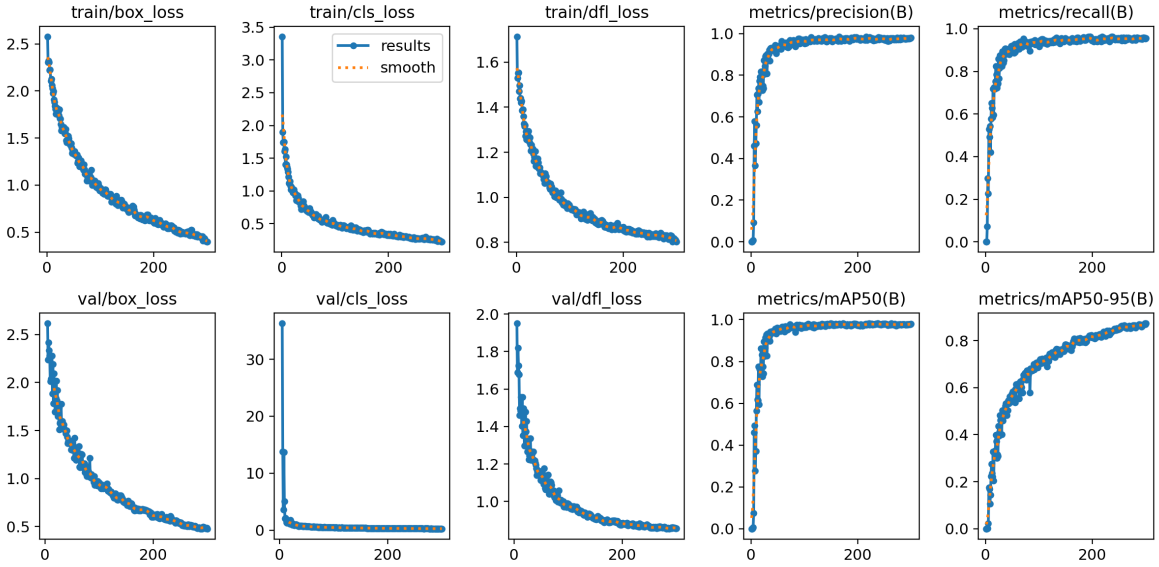



**Figure 2.14: YOLO training after applying the preprocessing pipeline**

In addition, the confusion matrix analysis (Figure 2.15) confirms high class-wise accuracy, with minimal confusion between “Lock Down” and “Lock Up” states and only a few false detections of background. These errors are relatively minor, as they can be effectively filtered out with simple post-processing rules. Taken together, these results show that with a larger dataset and a more structured data collection procedure, it is possible to obtain a high-performing system for this task.

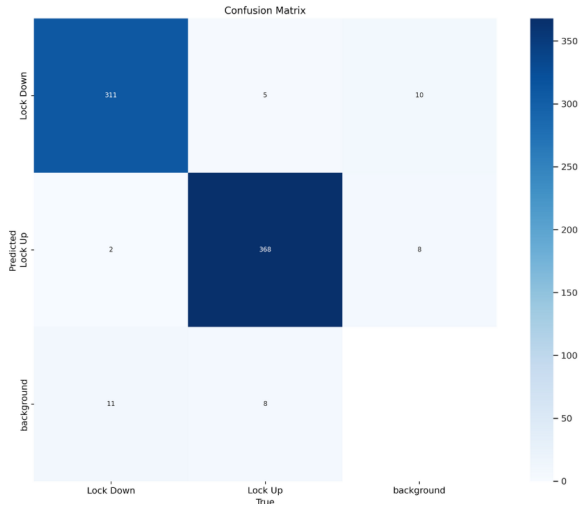



**Figure 2.15: Confusion matrix of the trained model. The results show minimal confusion between the "Lock Down" and "Lock Up" classes, with some background misclassifications**

### 2.6.4 Qualitative analysis

Figure 2.16 compares the model predictions on a raw input image versus a version processed with super-resolution. The raw image yields fewer detections and less accurate bounding boxes, while the enhanced version enables the model to identify more locks with better spatial precision. This highlights the positive impact of preprocessing techniques for super-resolution and the data augmentation procedure, which addresses challenging conditions that affect detection quality. In Figure 2.17, the model correctly identifies a lock located at the back of the cargo that was not annotated in the ground-truth image, and it also predicts the correct status for this lock. This result demonstrates the model's ability to generalize beyond the training labels and capture subtle visual cues.

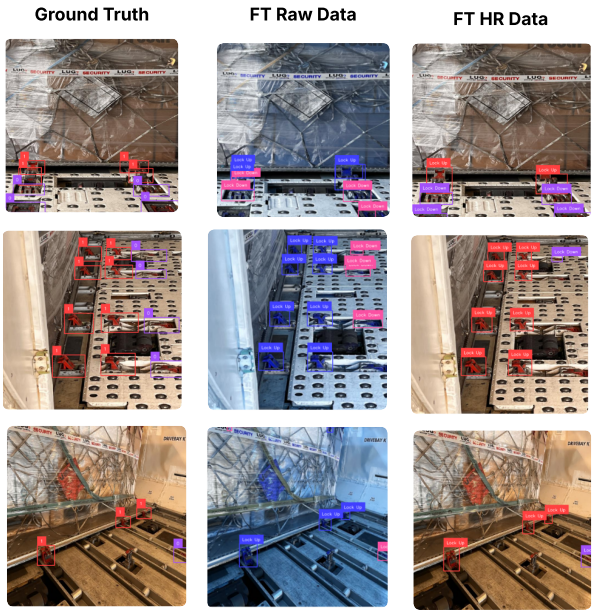



**Figure 2.16: Ground truth (left), detections using the raw images (middle), and detections after the super-resolution and data augmentation pipelines (right)**

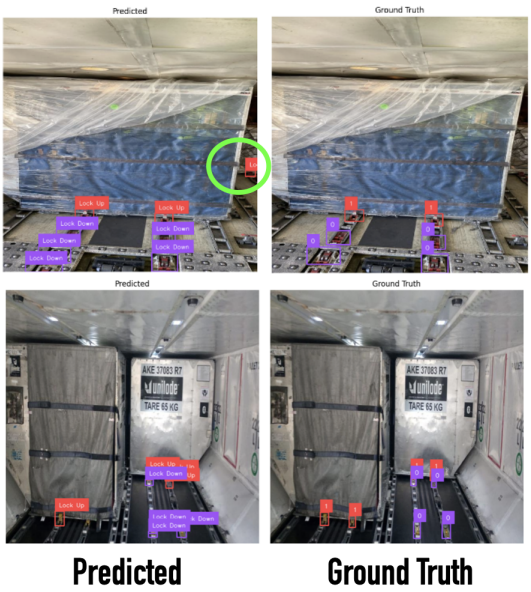



**Figure 2.17: Visual comparison of model predictions and actual labels**

These examples emphasize the importance of both increasing the dataset size and improving image capture conditions. A larger and more diverse dataset would help the model generalize better to new scenarios. At the same time, enhancements in image quality –either through automated super-resolution or improved operator guidelines for capturing clear and well-framed images– can significantly boost detection accuracy and robustness in deployment.

Moreover, it is worth mentioning that with a larger dataset and more time for training, employing medium or large models would be a logical next step to boost detection accuracy. In particular, since our application does not have a strict real-time requirement (e.g., processing 30 frames per second), considering larger models in the future can lead to a significant performance boost. Nevertheless, training such models will require more training data and processing power.

## 2.7 Conclusion and future work

In this project, we demonstrated that automated visual verification of cargo locks is achievable even with a limited dataset. By combining a robust object detection framework with targeted preprocessing steps, specifically data augmentation to simulate real-world variability and super-resolution to enhance image detail, we were able to overcome the limitations of low-quality inputs.

Our findings show that a straightforward supervised detector, when fed richer and clearer training examples, can generalize effectively to the task of identifying lock engagement. Moreover, the improvements from augmentation and enhancement suggest that defining a structured data collection protocol –specifically specifying camera viewpoints, lighting standards, and lock placement– and providing clear capture guidelines should further enhance system reliability. With the continued accumulation of diverse images and periodic fine-tuning, this approach can serve as a practical prototype for real-world deployment in cargo operations.

### 2.7.1 Future work

Future work should explore transformer-based detectors like DETR [4], which eliminate hand-crafted components such as anchor boxes and NMS. We also plan to test ensemble strategies using approaches such as Rank-of-Experts [1] or weighted box fusion [22] to combine YOLO and DETR outputs for

improved overall detection. Finally, leveraging domain-specific knowledge, such as constraining lock detections to ground-level regions as post-processing steps, will be investigated to reduce false positives.

## 2.8 Limitations

Although the proposed pipeline demonstrated strong performance on the provided dataset, several limitations must be acknowledged. The dataset consisted of only 117 labelled images, and although data augmentation significantly expanded this, the original sample was still very small and lacked diversity in terms of aircraft types, cargo configurations, and lighting conditions. Notably, certain lock types –especially damaged or non-standard variants– were underrepresented, which may affect detection reliability in edge cases.

Another limitation is the absence of metadata. We have no information about when or where the images were captured, nor the model of the aircraft from which they originated. This makes it difficult to assess whether our model would generalize to different settings, such as other cargo bays, lighting conditions, or aircraft types like the Boeing 787, which uses colour-coded locks. These variations could introduce domain shifts that significantly affect performance.

To address these concerns, future work should investigate domain adaptation techniques [15] such as domain-adversarial training [7], which aim to learn domain-invariant features. Additionally, domain generalization methods [11, 26] could help build models that perform well even on unseen environments, without access to target domain labels. These methods will be important to ensure reliability across varying real-world conditions.

## Bibliography

[1] Seung-Hwan Bae, Youngwan Lee, Youngjoo Jo, Yuseok Bae, and Joong-won Hwang. Rank of experts: Detection network ensemble. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017. Rank-based ensemble of detectors.

[2] Bernd Bischl, Martin Binder, Michel Lang, Tobias Pielok, Jakob Richter, Stefan Coors, Janek Thomas, Theresa Ullmann, Marc Becker, Anne-Laure Boulesteix, et al. Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 13(2):e1484, 2023.

[3] Alexander Buslaev, Vladimir I Iglovikov, Eugene Khvedchenya, Alex Parinov, Mikhail Druzhinin, and Alexandr A Kalinin. Albumentations: fast and flexible image augmentations. Information, 11(2):125, 2020.

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In European Conference on Computer Vision (ECCV), 2020.

[5] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05), volume 1, pages 886–893. IEEE, 2005.

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition, pages 248–255. IEEE, 2009.

[7] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In Proceedings of the 32nd International Conference on Machine Learning (ICML), 2015. Domain-adversarial training.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778, 2016.

[9] John H Holland. Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. MIT press, 1992.

[10] Glenn Jocher, Ayush Chaurasia, Alex Stoken, Jirka Borovec, Yonghye Kwon, Kalen Michael, Jiacong Fang, Zeng Yifu, Colin Wong, Diego Montes, et al. ultralytics/yolov5: v7. 0-yolov5 sota realtime instance segmentation. Zenodo, 2022.

[11] Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C Kot. Domain generalization with adversarial feature learning. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 5400–5409, 2018.

[12] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In Proceedings of the IEEE international conference on computer vision, pages 2980–2988, 2017.

[13] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In European conference on computer vision, pages 740–755. Springer, 2014.

[14] Matthias Minderer, Alexey Gritsenko, Austin Stone, Maxim Neumann, Dirk Weissenborn, Alexey Dosovitskiy, Aravindh Mahendran, Anurag Arnab, Mostafa Dehghani, Zhuoran Shen, et al. Simple open-vocabulary object detection. In European conference on computer vision, pages 728–755. Springer, 2022.

[15] Saeid Motiian, Marco Piccirilli, Donald A Adjeroh, and Gianfranco Doretto. Unified deep supervised domain adaptation and generalization. In Proceedings of the IEEE international conference on computer vision, pages 5715–5725, 2017.

[16] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In International conference on machine learning, pages 8748–8763. PmLR, 2021.

[17] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 779–788, June 2016.

[18] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems, 28, 2015.

[19] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 658–666, 2019.

[20] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention, pages 234–241. Springer, 2015.

[21] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.

[22] Roman Solovyev, Weimin Wang, and Tatiana Gabruseva. Weighted boxes fusion: Ensembling boxes from different object detection models. arXiv preprint, 2019. Method to fuse bounding boxes weighted by confidence.

[23] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001, volume 1, pages I–I. IEEE, 2001.

[24] Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In International Conference on Computer Vision Workshops (ICCVW), 2021.

[25] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In Proceedings of the European Conference on Computer Vision (ECCV) Workshops, September 2018.

[26] Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. IEEE transactions on pattern analysis and machine intelligence, 45(4):4396–4415, 2022.

# 3 Banque Nationale Courtage Direct: Forecasting the volume of Canadian investments in foreign equities

**Ting-Huei Chen, coordinator** [a]
**Golshid Aflaki** [b, d]
**Elvis Romarick Fotsi** [a]
**Zahra Hashemi** [b]
**Sébastien Jessup** [c]
**Dérek Laguë** [a]
**Anthony Larouche** [a]
**Alain Didier Noutchegueme** [e]
**Farid Rajabali** [d, f]

[a] Université Laval
[b] HEC Montréal
[c] University of Waterloo
[d] GERAD
[e] Université de Montréal
[f] Polytechnique Montréal

**October 2025**
**Les Cahiers du GERAD**

**Abstract:** *This report presents an analysis conducted during the Fifteenth Montréal Industrial Problem Solving Workshop in collaboration with Banque Nationale Courtage Direct (BNCD). Due to the unavailability of BNCD's proprietary data, we used Statistics Canada's International Transactions in Securities dataset as a proxy. The study includes qualitative exploration, quantitative time series modelling, and machine learning approaches. An exploratory sentiment analysis using a subset of the FNSPID dataset and a RoBERTa classifier is also included. All results rely on national-level proxy data and must be re-evaluated when BNCD's internal transaction data becomes available.*

## 3.1 Introduction

This report was prepared as part of the *Fifteenth Montréal Industrial Problem Solving Workshop*, organized by the Centre de Recherches Mathématiques (CRM), in collaboration with *Banque Nationale Courtage Direct* (BNCD). The project focused on estimating the volume of trading in U.S.-listed equities executed by Canadian investors, with particular interest in modelling the trading behaviour of BNCD's own brokerage clients.

Accurate forecasting of trading volume is essential for BNCD's internal planning processes, including budgeting and strategic decision-making. In the short term, understanding the drivers of sharp fluctuations in trading activity could help interpret anomalies and refine existing models. In the longer term, robust predictive models would allow BNCD to anticipate transaction volumes and better align resource allocation with market behaviour.

Ideally, the modelling effort would rely on BNCD's proprietary transaction data. However, due to confidentiality constraints, such data could not be accessed during the workshop. As a result, the objective was refocused toward estimating the national-level volume of Canadian investment in U.S. equities using publicly available data as a proxy.

To this end, we used Statistics Canada's *International Transactions in Securities* dataset [6], which reports monthly Canadian portfolio flows into foreign equities. While this dataset does not explicitly break out U.S. stocks, they constitute over 85% of the foreign equity transactions, making it a relevant and informative proxy. We also considered, but ultimately set aside, supplementary indicators such as exchange-traded fund (ETF) volumes and currency exchange rates due to either insufficient resolution or limited predictive gain.

The report presents our modelling strategy based on this proxy data, combining exploratory data analysis with time series forecasting and machine learning techniques. Our findings indicate that while time series models perform well in short-term forecasts, machine learning approaches may offer better performance over longer horizons and provide additional interpretability. However, the results should be interpreted with caution, given the mismatch between the proxy market and BNCD's actual client base. Ultimately, the models developed in this project form a foundation that can be further refined using internal data, once available.

## 3.2 Qualitative analysis

To address the objectives of BNCD, it is first essential to gain a thorough understanding of the response variable to be modelled. Figure 3.1 illustrates the purchases of foreign equity, in millions of Canadian dollars, over time. A marked increase in volatility can be observed in 2008, coinciding with the global financial crisis, followed by a period of relative stability lasting until 2016. From that point onward, a new phase of heightened volatility emerges and persists to the present day.

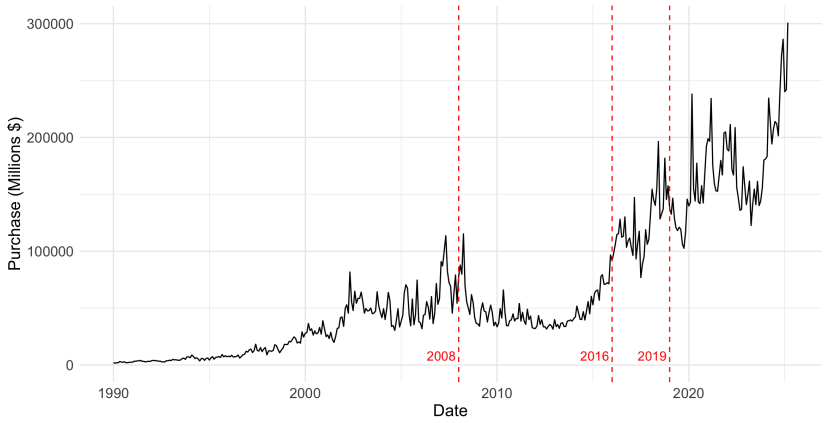



**Figure 3.1: Volume of purchases of foreign equity over time**

However, when a time series exhibits heterogeneous variability, it becomes necessary to stabilize it using an appropriate transformation. Such heterogeneity violates the assumptions of many statistical models, particularly those concerning homoscedasticity. A common approach to address this issue is to apply a logarithmic transformation to the response variable, as illustrated in Figure 3.2.

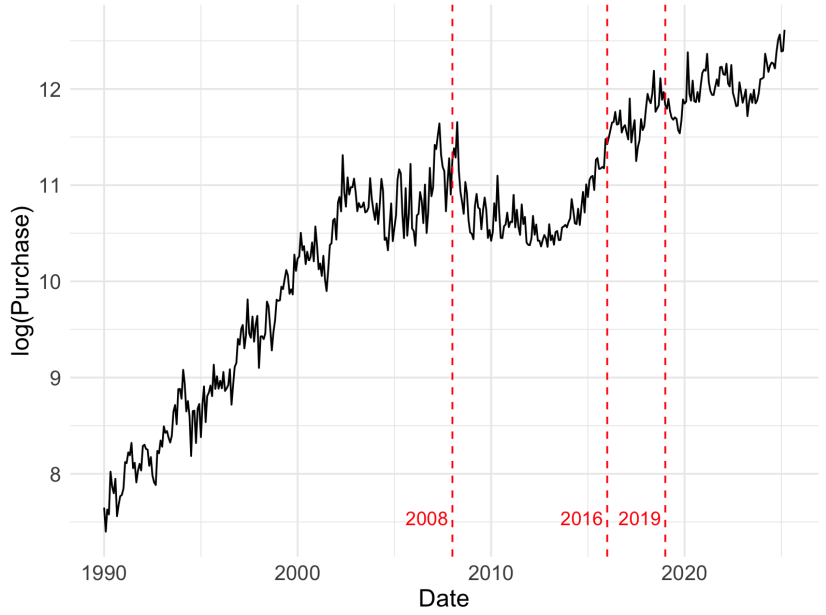



**Figure 3.2: Log-volume of purchases of foreign equity over time**

The logarithmic transformation highlights several shifts in trend and appears to contribute to variance stabilization. A potential seasonal component is also visible in the series, which could be leveraged for forecasting purposes.

The client also reported that their model became significantly less accurate starting in 2019. Analysis of the first two figures suggests that this decline in performance may be attributed to a sharp increase in variability, or more broadly, uncertainty during this period. These years were marked by several major events, including the COVID-19 pandemic, various geopolitical conflicts, and tariff threats under

President Trump. This heightened climate of uncertainty likely influenced the behaviour of individual investors. The following figure shows the relationship between transaction volume (in millions of Canadian dollars) and the logarithm of the Economic Policy Uncertainty Index.

The Economic Policy Uncertainty Index is a score constructed from newspaper coverage of policy-related economic uncertainty, as described in their official methodology (see [3]). Figure 3.3 shows a notable increase in this index during the years surrounding President Trump's first election, which aligns with the observed rise in volatility in the time series.

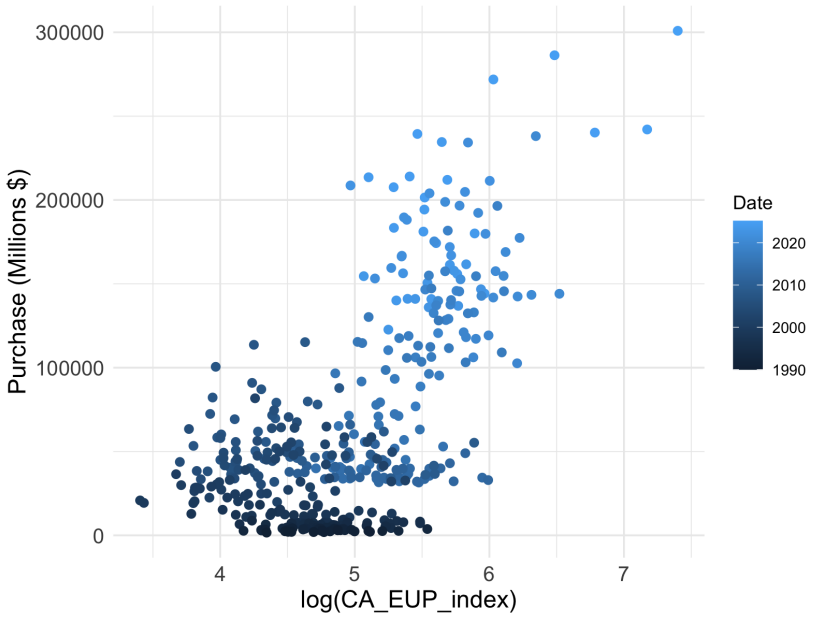



**Figure 3.3: Volume of foreign equity purchases versus the log of the Economic Policy Uncertainty Index. Here, "EUP" corresponds to the French abbreviation for the Economic Policy Uncertainty Index (EPU)**

It is therefore important to note that the high variability in transaction volume during these years reflects national-level aggregation, whereas BNCD's specific objective is to model the behaviour of its internal clientele. A more focused analysis of individual investor behaviour, excluding institutional actors, could significantly enhance the understanding of transaction volume dynamics.

The response variable plots, along with their relationship to the logarithm of the Economic Policy Uncertainty Index, suggest that a more in-depth analysis of internal clients is necessary to address the two main objectives more effectively. Nevertheless, the aggregated response variable still provides valuable insight, as it reveals a notable shift in the environment in terms of variability and uncertainty. Such elevated levels of variability and uncertainty pose significant challenges for any mathematical model, which, by nature, remains a simplification of the complex and evolving real-world environment.

### 3.2.1 Purpose and context

This section synthesizes the behavioural and macro-financial drivers behind Canadian purchases of foreign equities. It complements the quantitative forecasts presented elsewhere by unpacking the causal mechanisms—wealth effects, regulatory constraints, risk sentiment and institutional frictions—that give rise to the statistical regularities and transmit exogenous shocks through Canadian portfolios. Our qualitative framework unfolds in three interconnected stages:

1. **Correlation Analysis** (subsection 3.2.2): an interpretive discussion of the cross-sectional correlation heat-map between foreign-equity flows and key market variables.
2. **Narrative Rule-of-Thumb** (subsection 3.2.3): a story-driven account of the exceptionally tight linkage with the S&P 500, culminating in an empirically grounded "480-rule."
3. **Event Study** (subsection 3.2.4): a retrospective examination of six headline shocks and their structural break effects on investor behaviour.

By emphasizing transmission channels rather than point estimates, we trace how shocks propagate and interact with institutional features. All external data sources are cited parenthetically.

### 3.2.2 Correlation structure: What the heat-map reveals

The Pearson correlation matrix in Figure 3.4 maps the incentives faced by Canadian investors:

- *Equity Momentum* ($\rho \approx 0.9$): foreign equity purchases co-move almost one-for-one with the S&P 500. Rising U.S. prices both revalue existing foreign positions and reinforce momentum expectations, creating a self-reinforcing loop of capital chasing equity gains.
- *Yield-Curve Tilt* ($\rho \approx 0.5$): a steeper Canadian yield curve signals a relative pick-up in long-term bond yields, prompting portfolio rebalancing toward risk assets, of which foreign equities comprise the geographically diversified slice.
- *Exchange-Rate Effect* ($\rho \approx 0.3$): a stronger CAD marginally reduces the local cost of U.S. shares, but this valuation effect is overshadowed by return-driven incentives.
- *Volatility Timing* ($\rho \approx 0$ with VIX): investors respond to persistent bull or bear regimes rather than to transitory volatility spikes, which wash out at monthly frequency.

Together, these patterns imply that Canadian institutions are *return-seekers first, hedgers second.*

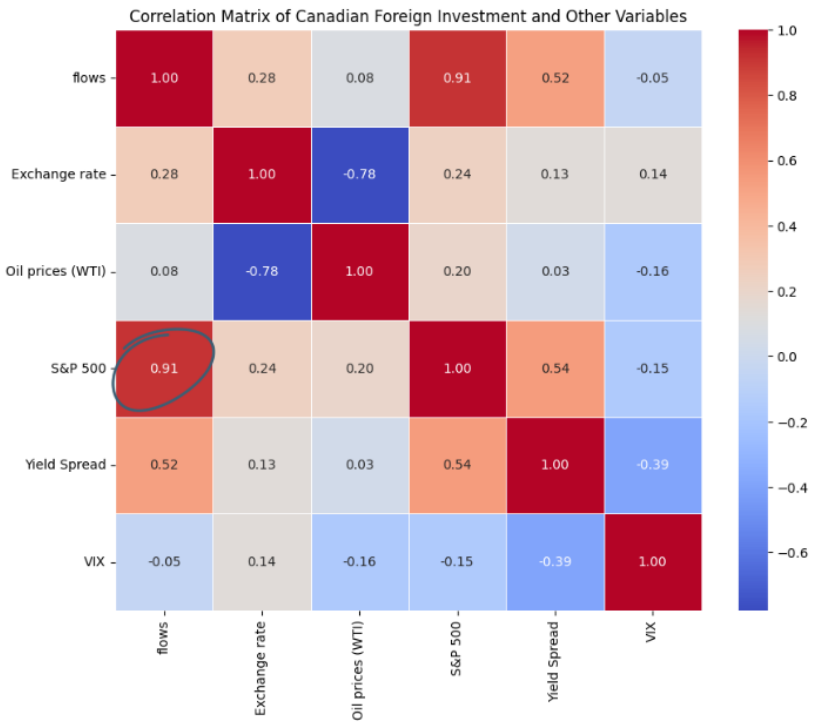



**Figure 3.4: Heat-map of Pearson correlations between foreign-equity flows and principal economic indicators**

### 3.2.3 Deepening the S&P 500 narrative

#### Visual co-movement

Plotting annual net purchases against the contemporaneous mean level of the S&P 500 (Figure 3.5) reveals three distinct phases:

1. **2000–2004**: modest flows tracking the post-dot-com recovery.
2. **2005–2007**: Canadian investment in foreign equities exhibits only minor fluctuations, following the removal of the 30% foreign-content limit in registered plans, and even its lowest point between 2005 and 2007 remains above the 2005 baseline.
3. **2016–2021**: synchronized climb interrupted by the 2022 inflation-geopolitical shock.

The visual alignment of the two series corroborates the high correlation and illustrates how U.S. equity levels act as a proxy for global risk appetite.

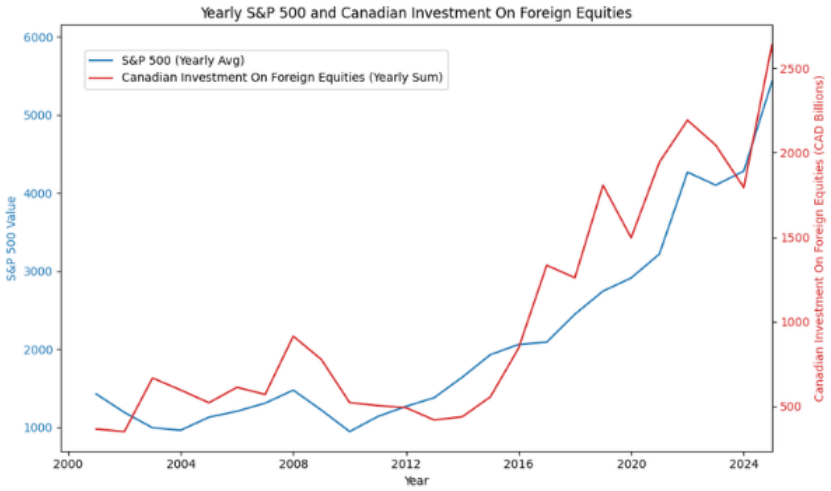



**Figure 3.5: Annual net purchases of foreign equities vs. S&P 500 annual mean**

#### Ratio stability and the "480-Rule"

Normalizing flows by the S&P 500 generates a ratio fluctuating between 300 and 700, centring on roughly 480 (Figure 3.6). Interpreted as an implicit leverage factor, it implies that each incremental point in the S&P 500 elicits about CAD 480 million in Canadian purchases. Using the 2024 average S&P 500 level of 5 500, the midpoint rule yields

$$5\,500 \times 480 \text{ million CAD} \approx 2.6 \times 10^{12} \text{ CAD}$$

for 2025, a useful scale estimate rather than a precise forecast. Sensitivity bounds (300–600) imply a plausible range of CAD 1.7–3.3 trillion.

#### Interpretive caveats

The linear "480-rule" abstracts from:

- *Regulatory shifts* (e.g., future pension reforms).
- *Valuation cycles* (P/E ratios, earnings momentum).
- *Global diversification* beyond U.S. equities.

Nonetheless, it captures the joint income and confidence channels driving cross-border equity flows.

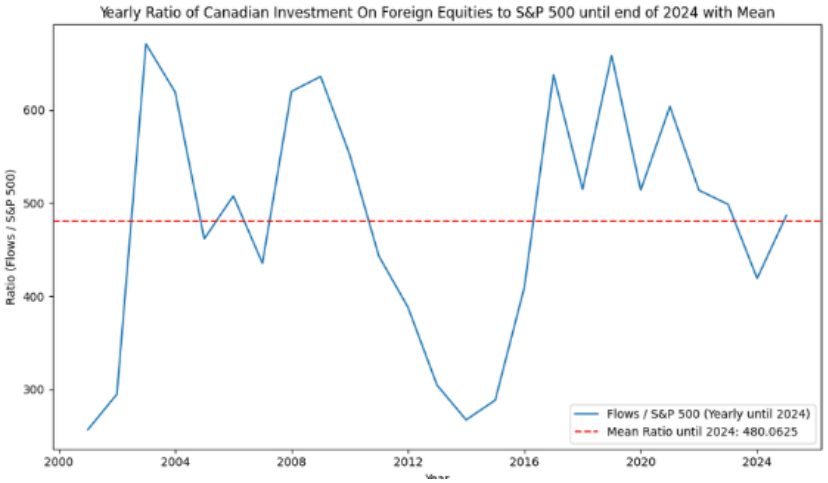



**Figure 3.6: Ratio of foreign-equity flows to S&P 500 level**

### 3.2.4 Major-event lens: Six structural shocks

Figure 3.7 overlays six headline shocks on the time series of purchases, highlighting their structural impacts.

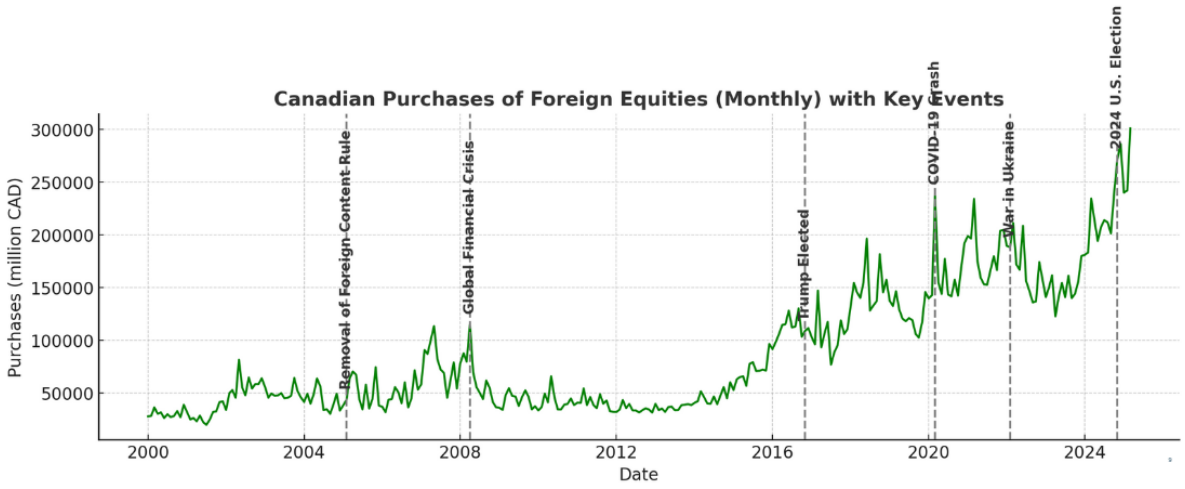



**Figure 3.7: Net purchases of foreign equities with annotated structural shocks**

**Foreign-Content Rule Removal (February 2005)** A one-off upward level shift in baseline outflows, illustrating the primacy of policy in resetting capital-flow norms.

**Global Financial Crisis (May 2008)** A rapid reversal to net negative flows within four months, driven by liquidity needs, USD-funding stress, and broad risk aversion.

**U.S. Election (November 2016)** A brief pause in October–November, followed by accelerated inflows as the post-election policy outlook clarified.

**COVID-19 Crash (March 2020)** An unprecedented CAD 300 billion sell-off in one month, offset by a swift buying rebound once liquidity backstops were deployed.

**Ukraine War & Inflation (February 2022)** A partial rebound after CAD 60 billion Q1 divestment, signalling greater persistence when shocks originate in supply constraints.

**2024 U.S. Election (November)** A shallow, short-lived dip in October, reaffirming that elections generate timing noise rather than permanent breaks absent non-consensus policy shifts.

### 3.2.5 Synthesis and Forecasting Implications

The qualitative evidence converges on three propositions:

1. *Return-chasing dominates*: U.S. equity performance serves as the global risk barometer.
2. *Policy shocks reset the baseline*: regulatory changes shift the long-run mean of outflows.
3. *Crisis episodes are asymmetric*: sell-offs are sharp but quickly reverse once backstops are in place; supply-driven shocks have more persistent effects.

A robust flow-projection model should integrate:

- A *structural term* for regulatory regime and demographic saving trends;
- A *cyclical term* tied to realized and expected U.S. equity returns, modulated by risk-sentiment indices;
- A *shock term* activating under tail-risk conditions to capture transient overshoots and rapid mean reversion.

Calibrating this tripartite framework around the "480-rule" provides a disciplined yet flexible template for scenario planning.

## 3.3 Quantitative analysis

### 3.3.1 Machine learning

In response to the dual challenge posed by BNCD, our objective is twofold: to predict the volume of foreign equity purchases by Canadians and to explain discrepancies between predicted and observed volumes. Machine learning offers a promising framework for addressing both tasks. Techniques such as generalized additive mixed models (GAMMs; see [8]), neural networks [5], and random forests [4] are well-suited for both regression and classification tasks. Meanwhile, support vector regression (SVR; see [2]) is particularly effective for predictive modelling.

These methods typically involve a model $\mathcal{M}$ that maps a vector of input features $\mathbf{x}$ to a target variable $y$, expressed as $y = \mathcal{M}(\mathbf{x})$. Despite their potential, machine learning models present several challenges in this context. First, most models assume independence between observations, which is violated here due to temporal dependencies. Second, our goal of forecasting a full year ahead introduces substantial complexity. While machine learning models can effectively predict one month ahead using monthly indicators, extending this to a full year would require forecasting all input variables over that horizon. Such an approach would introduce significant uncertainty and could render the model impractical. A more feasible alternative is to use lagged monthly indicators from the previous year to capture seasonal patterns and market variability.

Another limitation is the lack of interpretability often associated with machine learning models, which can hinder transparency and trust in the results. Nevertheless, when temporal structure is properly incorporated, these models can identify key drivers of transaction volume and provide valuable insights into client behaviour and market dynamics.

In fact, using machine learning techniques, we can investigate the relevance of diverse market predictors in explaining the volume of transactions. In particular, starting from approximately 100 different variables related to macroeconomic factors (interest rates, inflation, GDP, unemployment rate, etc.), market factors (volatility, stock indices, exchange rates, etc.), and sectoral factors (commodity

prices and other trends and risks specific to the company's industry), we can use stepwise AIC to identify the variables with the most predictive power concerning volume of transactions.

We find variables such as the S&P 500 index, front-month futures contract for gold, and the Canadian Economic Policy Uncertainty Index. All of these are intuitive in explaining market movement and uncertainty, which can explain purchases. Figure 3.8 illustrates the link between gold commodity prices compared to the volume of transactions, further highlighting the intuitive nature of the variables identified by stepwise AIC.

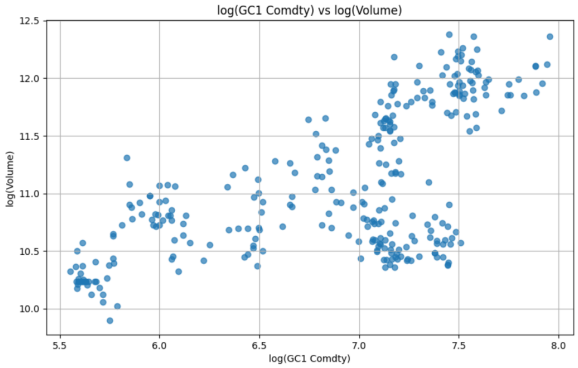



**Figure 3.8: Log-volume of transactions compared to log of gold commodity price**

We can further observe the indices with strong explanatory power through the Shapley additive explanations in Figure 3.9.

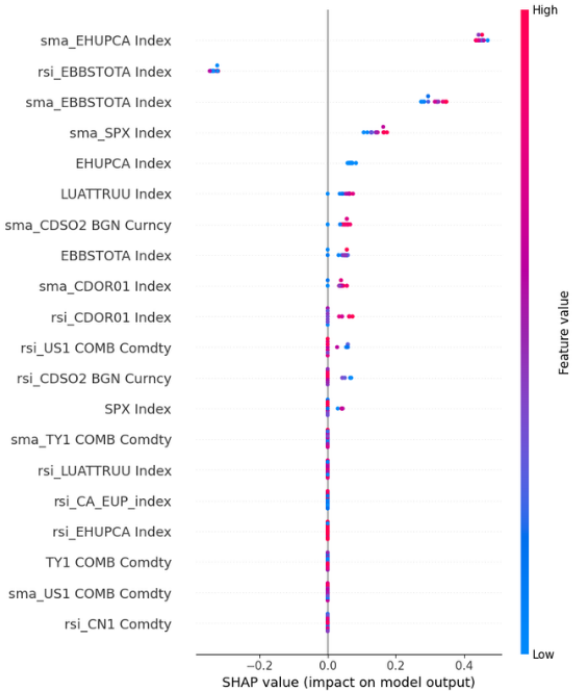



**Figure 3.9: SHAP values of selected predictors**

To explore the relevance of temporal modelling, we test a simple GAMM for next-month forecasting using a single predictor, namely the S&P 500 index, while incorporating smooth terms for month and year, allowing us to account for the autoregressive nature of the data. More specifically, letting

$y_t = \log(\text{volume}_t)$

$$y_t = \beta_0 + \beta_1 \text{SPX}_{t-1} + f_1(\text{year}_t) + f_2(\text{month}_t) + \epsilon_t,$$

where

$$\epsilon_t = \phi \epsilon_{t-1} + \eta_t,$$

where $\phi$ is the autocorrelation parameter, and $\eta_t$ represents white noise.

The results, summarized in Table 3.1, indicate that the temporal trend is more significant than the S&P 500 index in predicting next-month activity, supporting the intuition that time series considerations are crucial in this setting.

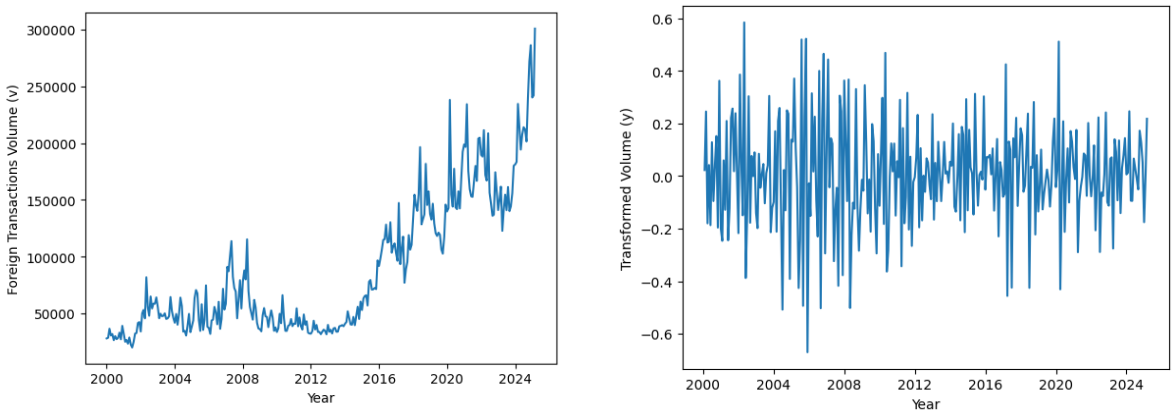
**Table 3.1: Significance levels of GAMM covariates**

| Covariate | p-value |
|---|---|
| Intercept | $< 2 \times 10^{-16}$ |
| SPX | 0.442 |
| s(year) | 0.0039 |
| s(month) | 0.00577 |

### 3.3.2 Time series

In this section, various time series models are implemented to forecast international trade volumes, and their performance is compared. Prior to modelling, necessary preprocessing steps and statistical tests are conducted to ensure data stationarity and suitability for forecasting.

#### Data transformation and stationarity check




**(a) Original Trade Volume** $V_t$

**(b) Transformed Volume** $Y_t = \Delta \log(V_t)$

**Figure 3.10: Visual comparison of trade volume before and after transformation**

Canadian international trade data were analyzed, initially revealing non-stationarity in raw trade volumes. To address this, a logarithmic transformation followed by first differencing was applied. This transformation stabilizes variance and removes trend components from the series. The resulting transformed variable is denoted by $Y$, and defined as:

$$Y_t = \log(V_t) - \log(V_{t-1}) = \Delta \log(V_t)$$

where $V_t$ represents the original trade volume at time $t$. This transformation is critical for achieving stationarity, which is a necessary condition for reliable time series modelling. Based on the Augmented Dickey-Fuller (ADF) and KPSS tests, the transformed series $Y_t$ was confirmed to be stationary:

- **ADF Test:** stat = -5.1696, p-value = 0.0000
- **KPSS Test:** stat = 0.0400, p-value = 0.1000

### Seasonality and lag analysis

Seasonality was first examined through a monthly distribution of trade volume data, revealing recurring fluctuations across calendar months. This observation suggested the presence of seasonal effects. To further investigate, autocorrelation (ACF) and partial autocorrelation (PACF) plots were analyzed. These diagnostics confirmed significant seasonal dependencies, with notable lags at [1, 2, 4, 12] for autoregressive (AR) terms and [1, 4, 12] for moving average (MA) terms.

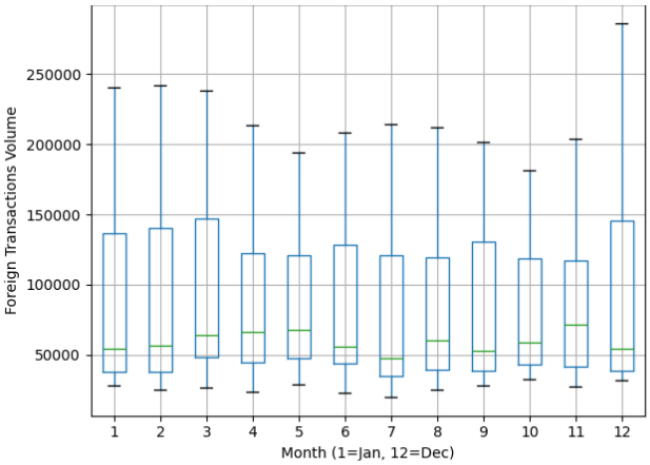



**Figure 3.11: Monthly boxplot of foreign transaction volumes**

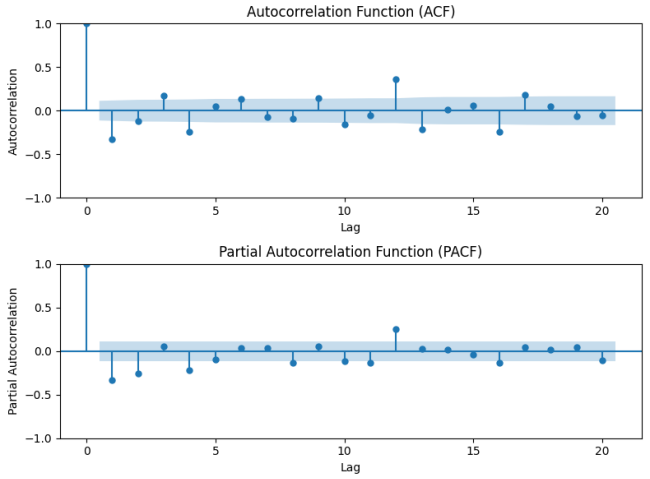



**Figure 3.12: ACF and PACF plots**

### ARIMA

ARIMA serves as the foundational benchmark in this analysis. A grid search was conducted over different combinations of autoregressive (AR) lags (p) and moving average (MA) lags (q) to determine the most appropriate model configuration. Model selection was based on the minimization of the Akaike Information Criterion (AIC) and the Bayesian Information Criterion (BIC). Based on this optimization, the ARIMA(4,1,5) configuration was selected as the best-performing model:

**Table 3.2: Top ARIMA models based on AIC and BIC scores**

| Model | AIC | BIC |
|---|---|---|
| ARIMA(4, 1, 5) | -93.03 | -56.16 |
| ARIMA(5, 1, 2) | -91.42 | -61.25 |
| ARIMA(2, 1, 4) | -91.16 | -64.34 |
| ARIMA(4, 1, 3) | -90.39 | -60.23 |
| ARIMA(3, 1, 3) | -90.26 | -63.44 |

The ARIMA(4,1,5) model was trained on the training subset and evaluated on the test data. The resulting Root Mean Square Error (RMSE) was 0.1457.

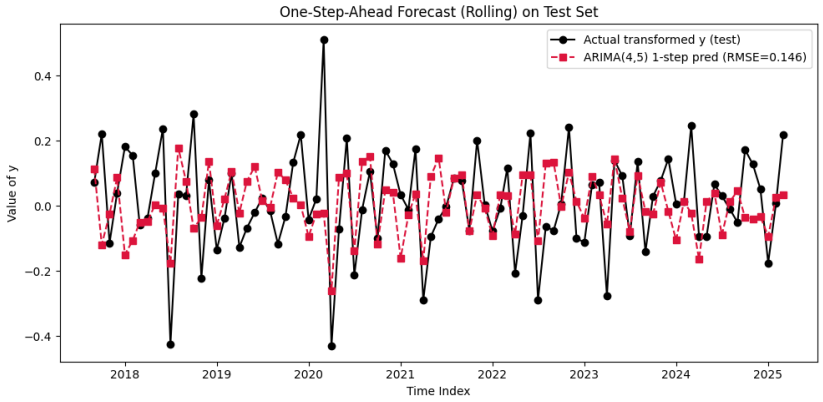



**Figure 3.13: Rolling one-step-ahead forecast comparison: ARIMA(4,1,5) vs. actual $y$**

## SARIMAX

To build on the ARIMA model, a SARIMAX (Seasonal ARIMA) model was implemented using only the statistically significant lags identified from ACF and PACF analyses. Specifically, AR lags of [1, 2, 4] and MA lags of [1, 4] were included, along with a seasonal component at lag 12 to capture annual effects.

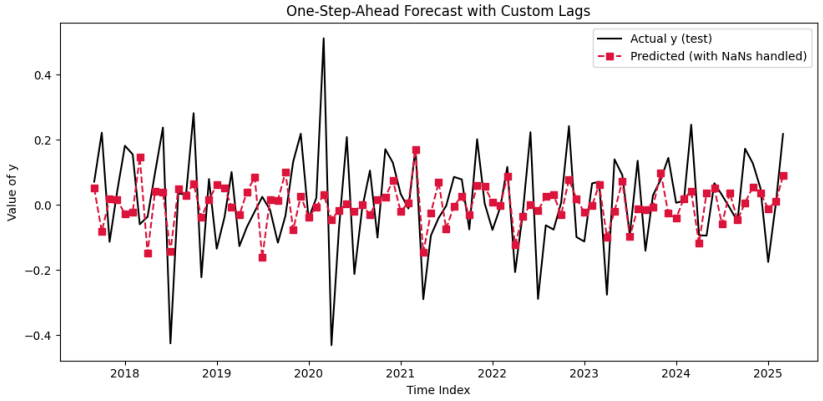



**Figure 3.14: Rolling one-step-ahead forecast comparison: SARIMAX vs. actual $y$**

The SARIMAX model was trained on the same training dataset and evaluated on the test set using one-step-ahead forecasting. The resulting Root Mean Square Error (RMSE) was 0.0961, demonstrating a notable improvement in predictive accuracy over the baseline ARIMA model.

### SARIMAX with control variables

To extend the SARIMAX model, control variables were introduced, including the VIX, USD/CAD exchange rate, oil prices (WTI), S&P 500 index, yield spread, and volume of foreign transactions. Each variable's stationarity was tested using the Augmented Dickey-Fuller (ADF) and KPSS tests, and first differences were taken to ensure stationarity.

Polynomial features of degree 3 were constructed for the differenced variables, capturing linear, squared, and cubic relationships. Lasso regression was then applied to select the most informative predictors. The selected features and their coefficients were as follows:

- VIX × S&P 500 × Yield Spread: 0.0166
- S&P $500^2$: 0.0159
- VIX: 0.0021
- Oil prices (WTI) × Yield Spread: 0.0010
- $\text{VIX}^2$ × Yield Spread: -0.0027
- Oil prices (WTI): -0.0045
- VIX × Oil prices (WTI): -0.0061

These selected variables were added to the SARIMAX model with the previously determined custom lags. After training and testing the updated model, the RMSE was found to be 0.1532. This result indicates that including control variables did not improve predictive performance relative to the SARIMAX model without them.

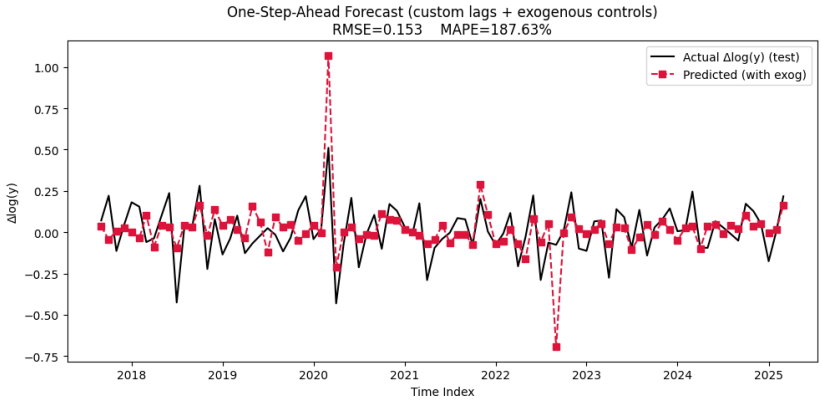



**Figure 3.15: One-step-ahead Forecast: SARIMAX with control variables vs actual $\Delta \log(y)$**

### Volatility modelling with GARCH

To account for volatility clustering and time-varying conditional variance observed in the trade volume data, a GARCH (Generalized Autoregressive Conditional Heteroskedasticity) model was implemented. While ARIMA and SARIMAX models effectively captured the conditional mean, they were limited in modelling the conditional variance, particularly during periods of heightened market turbulence.

A GARCH(1,1) model was fitted to the residuals obtained from the best-performing ARIMA model, assuming normally distributed errors. The specification was chosen based on standard information criteria and diagnostics on residuals and squared residuals, which indicated the presence of autoregressive conditional heteroskedasticity (ARCH effects).

Forecast evaluation of the ARIMA-GARCH model on the test set yielded an RMSE of 0.1450, which did not outperform the SARIMAX model. It is worth noting that only a basic GARCH configuration was examined in this study. Future research could explore advanced GARCH specifications (e.g.,

EGARCH, TGARCH, multivariate GARCH) and parameter tuning to potentially improve forecast performance.

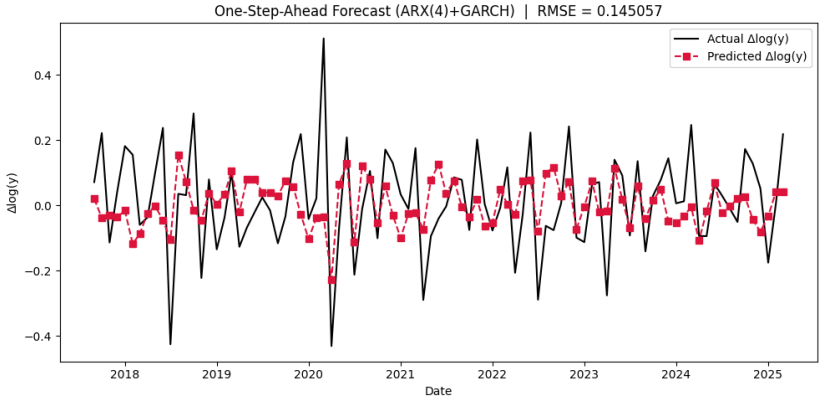



**Figure 3.16: Forecast and conditional variance from ARIMA-GARCH model**

Based on RMSE, the SARIMAX model outperformed the other time series models considered in this study. However, it is important to note that RMSE may not always be the optimal criterion for selecting the best forecasting model. For instance, GARCH models are particularly well-suited for capturing dynamics during periods of market distress or regime shifts, where volatility plays a significant role. Conversely, SARIMAX models offer an advantage when the goal is to identify the most influential explanatory variables affecting trade volume. Table 3.3 briefly compares the models; here, ARIMA(4,5) is shorthand for ARIMA(4,1,5).

| Model | RMSE (Lower Better) | Pros | Cons |
|---|---|---|---|
| ARIMA(4,5) | 0.1457 | Simple; works well when stable | Struggles during volatility spikes |
| SARIMAX | 0.0961 | Simple, works well when stable, captures seasonality | Struggles during volatility spikes |
| SARIMAX with control variables | 0.1532 | Incorporates external factors; improve accuracy in large datasets | Requires additional data; more complex |
| ARIMA-GARCH | 0.1450 | Captures changing volatility; best accuracy in turbulent periods | Higher complexity; greater computational cost |

**Table 3.3: Model comparison**

The SARIMAX model that achieved the lowest RMSE in our comparison is formally represented as:

$$Y_t = \mu + \sum_{i\in\{1,2,4\}} \phi_i Y_{t-i} + \sum_{j\in\{1,4\}} \theta_j \varepsilon_{t-j} + \Phi_{12} Y_{t-12} + \Theta_{12}\varepsilon_{t-12} + \varepsilon_t$$

where:

- $Y_t$ denotes the differenced log of trade volume at time $t$,
- $\mu$ is a constant intercept term,
- $\phi_i$ are autoregressive coefficients for lags 1, 2, and 4,
- $\theta_j$ are moving average coefficients for lags 1 and 4,
- $\Phi_{12}$ is the seasonal autoregressive term (lag 12),

- $\Theta_{12}$ is the seasonal moving average term,
- $\varepsilon_t$ is a white noise error term.

To retrieve the original trade volume $V_t$ from the differenced log-transformed series $Y_t$, the following inverse transformation is applied:

$$\log(V_t) = \log(V_{t-1}) + Y_t \quad \Rightarrow \quad V_t = V_{t-1} \cdot e^{Y_t}$$

This allows us to convert forecasted $Y_t$ values back into the original trade volume scale for interpretation and practical use.

### Additive model

The analysis of the graph of the volume[1] of foreign equity purchases by Canadians shows that we have a seasonal time series with a trend component that is increasing, a seasonal component, and an irregular component.

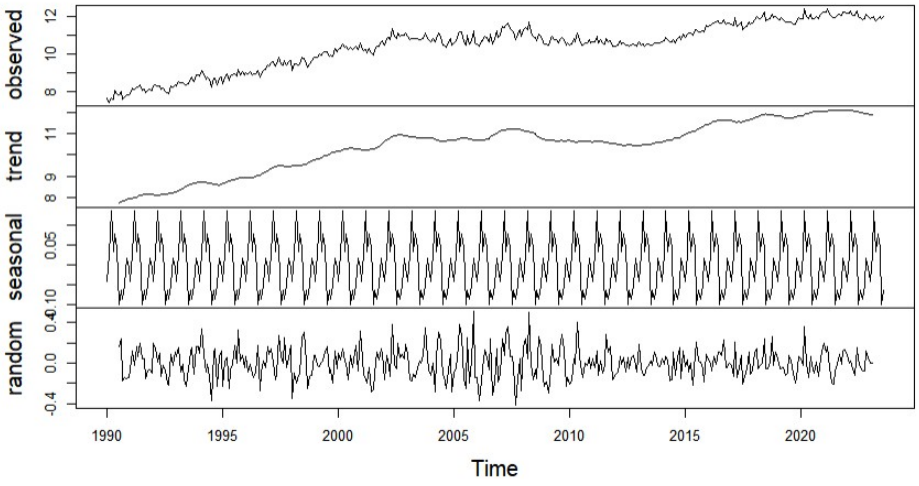



**Figure 3.17: Decomposition of the series**

The time series can be described using an additive model since the seasonal and random fluctuations seem to be roughly constant in size over time. Therefore, the method Holt-Winters exponential smoothing has been used to estimate the trend (level and slope) and seasonal component at the current time point. The model estimated by Holt-Winters is useful for short-term forecasting and makes no assumptions about the correlations between successive values of the time series. The general formula of the model is:

$$\hat{y}_{t+h|t} = \ell_t + h\, b_t + s_{\,t+h-m\,(k+1)}, \tag{3.1}$$

$$\ell_t = \alpha \left(y_t - s_{t-m}\right) + (1-\alpha)\left(\ell_{t-1} + b_{t-1}\right) = level, \tag{3.2}$$

$$b_t = \beta^* \left(\ell_t - \ell_{t-1}\right) + (1-\beta^*)\, b_{t-1} = trend, \tag{3.3}$$

$$s_t = \gamma \left(y_t - \ell_{t-1} - b_{t-1}\right) + (1-\gamma)\, s_{t-m} = seasonality, \tag{3.4}$$

where

$$k = \left\lfloor \frac{h-1}{m} \right\rfloor,$$

and the seasonal component can equivalently be written as

$$s_t = \gamma^* \left(y_t - \ell_t\right) + (1-\gamma^*)\, s_{t-m}, \quad \text{with} \quad \gamma = \gamma^*\,(1-\alpha).$$

[1] after the logarithm transformation

**Update equations:** The level and trend follow Holt's linear method, while the seasonal component blends the current seasonally adjusted observation $\big(y_t - \ell_{t-1} - b_{t-1}\big)$ with the seasonal index from $m$ periods ago.

**Forecasting:** At horizon $h$,

$$\hat{y}_{t+h|t} = \ell_t + h\, b_t + s_{\,t+h-m(k+1)}$$

so that the correct seasonal index is reused for multi-period ahead forecasts.

**Smoothing parameters:** $\alpha, \beta^*, \gamma$ control the weights given to past observations, with $0 \le \alpha, \beta^*, \gamma \le 1$. The alternative seasonal form uses $\gamma^* = \gamma/(1-\alpha)$.

The model has been estimated on data from January 1990 to December 2023, and the result at time $t$ is:

$$\ell_t = 12.06437, \quad b_t = 0.007910076,$$

and the seasonal indices for the 12 months:

$$\begin{aligned}(s_1, \ldots, s_{12}) = (\, &-0.040043388,\ 0.028017169,\ 0.137250573,\ 0.033892591,\\ &0.078308981,\ 0.045882936,\ -0.097762043,\ -0.063904559,\\ &-0.086141684,\ -0.049727303,\ 0.017271037,\ -0.003044309)\,.\end{aligned}$$

Then the $h$-step-ahead forecast is

$$\hat{y}_{t+h|t} = \ell_t \ + \ h\, b_t \ + \ s_{\,t+h-12\left(\lfloor (h-1)/12 \rfloor + 1\right)}.$$

In particular, for $1 \le h \le 12$ (i.e. within the next year), $\lfloor (h-1)/12 \rfloor = 0$ and

$$\hat{y}_{t+h|t} = 12.06437 \ + \ 0.007910076\, h \ + \ s_{\,h},$$

where $s_h$ is the seasonal index of the $h$-th upcoming month (January $= s_1$, February $= s_2$, …, December $= s_{12}$).

In terms of results, the model shows that for the first six months of 2024, the volume of predicted purchases is 1 050 892 (Millions $) and the real volume for this period is 1 214 531 (Millions $), around +13%.

## 3.4 Results

Figure 3.18 presents a comparison of the SVR model alongside the additive and SARIMAX time series models, compared to the actual purchase volume. For the first six months, the time series models outperform SVR, offering more accurate short-term predictions. However, in the latter half of the year, particularly in the last two months, SVR shows better performance, successfully capturing the observed increase in transaction volume, which the time series models failed to anticipate.

These results suggest that a hybrid approach, combining different models depending on the forecast horizon, could be optimal. Time series models are well-suited for short-term predictions, while machine learning methods like SVR may be better equipped to capture longer-term patterns and structural changes in the data.

## 3.5 Sentiment analysis

In this section, a different perspective is studied, focusing on the human and less predictable side of the stock market, which is equally important. This analysis supports the study of the impact of news on the stock market by examining the sentiment of financial news. The importance of studying financial

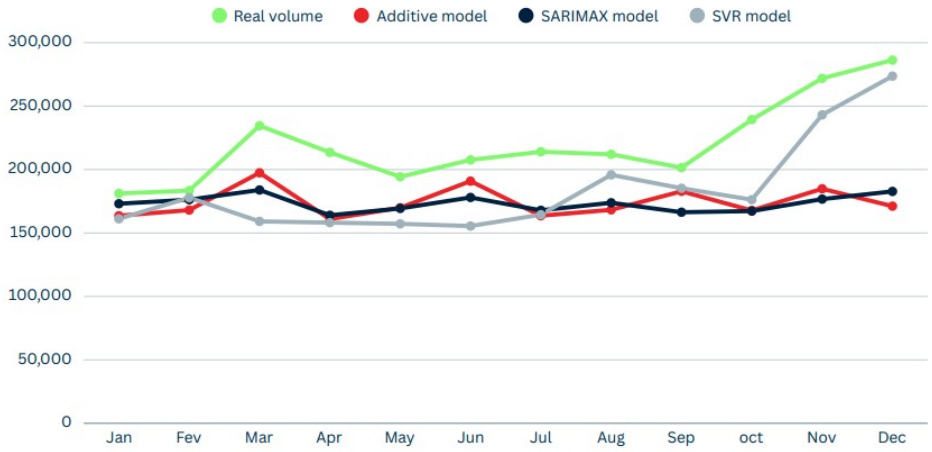



**Figure 3.18: Model performance comparison for 2024**

news lies in investors' behaviour, which is often influenced by more than just quantitative indicators. Market sentiment, or how investors feel about current events, plays a crucial role in shaping buying and selling decisions. Headlines about interest rates, economic outlooks, corporate performance, and geopolitical developments can quickly alter perceptions of risk and opportunity. Some day-to-day news inspires confidence, while other news sparks panic in investors' behaviour.

To investigate the relationship between financial news sentiment and the volume of Canadian investments in foreign equities, we performed a sentiment analysis on a large-scale financial news dataset. This analysis helps uncover patterns in how media tone may align with, or even predict, investor behaviour over time.

For this analysis, we used a subset of the publicly available FNSPID (Financial News and Stock Price Integration Dataset), a comprehensive dataset designed to enhance stock market prediction by integrating both quantitative and qualitative data sources. The dataset is available on GitHub and hosted via Hugging Face due to its large size. FNSPID contains 15.7 million financial news records and 29.7 million stock price entries. Each article is tagged with a company from the S&P 500 index, offering a detailed and high-frequency view of market-related news. Recorded news spans 1999 to 2023 and was aggregated from four major stock market news websites.

Due to the time and resource constraints of the workshop setting, we limited our scope to a sample from 2015 to 2020, using only the financial news portion of the dataset. We applied a transformer-based (RoBERTa-base) natural language processing model to classify the sentiment of each article into one of three categories: Positive, Negative, or Neutral. From this, we computed monthly sentiment counts, which were used as input features in our exploratory analysis.

Following an exploratory analysis, Figure 3.19 shows the most important and most repeated words in the 2020 financial news. This could also be done on a monthly basis to identify the buzzwords for each month.

To observe how sentiment behaves over time and how it might relate to investment activity, the top plot in Figure 3.20 shows the monthly volume of Canadian investments in foreign equities from 2015 to 2020. We can see a general upward trend, with noticeable spikes, particularly around 2020. In the bottom plot, the monthly counts of financial news articles, broken down by sentiment, green for positive, red for negative, and pink for neutral, are shown. Focusing a bit more around early 2020, we can see that the negative news (red) suddenly spikes, neutral news drops, and positive news eventually rises again.


**Figure 3.19: World cloud of cleaned news articles (2020)**

This aligns closely with the early months of the COVID-19 pandemic. This sharp shift in sentiment corresponds with a period of high market uncertainty, and interestingly, it also matches major fluctuations in investment volume. While we have not modelled causality here, this pattern gives us a strong reason to believe that news sentiment has a real influence on investor behaviour.

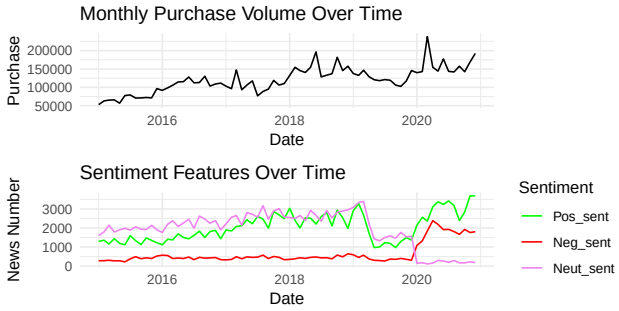



**Figure 3.20: Monthly volume of Canadian investments in foreign equities and monthly counts of financial news articles**

Furthermore, to explore the relationship between sentiment and investment behaviour more closely, Figure 3.21 shows the purchase volume time series, coloured based on the number of sentiment-tagged news articles each month. In the top row plot of Figure 3.21, the colour represents the number of positive news articles. You can see that during times when the green gets brighter—meaning more positive news—we often see a rise or sustained level in investment volume. A similar pattern is shown in the middle row plot, but coloured by negative sentiment. You can immediately spot the bright red spike around early 2020—that's the COVID outbreak. Interestingly, we also see a drop followed by a sharp rise in investment, possibly due to panic followed by the rapid spread of news. The bottom row plot is coloured by neutral sentiment. There is less of a clear pattern here, which is expected, as neutral news tends to have a weaker emotional impact and might not drive investor behaviour in the same way.

To further investigate the relationship between financial news sentiment and market behaviour, the S&P 500 index was visualized over time, colour-coded by the monthly volume of Positive, Negative, and Neutral financial news articles. This allows us to explore how sentiment levels correspond to broad market movements. The top row plot in Figure 3.22 shows that the periods of higher positive sentiment (brighter green) often coincide with rising or stable market levels as expected. Notably, after

the COVID-19 dip in early 2020, the recovery period aligns with an increase in positive sentiment, suggesting that optimistic coverage may support investor confidence during rebounds. The middle row plot shows that spikes in negative sentiment (intense red) are most visible during major downturns, especially the market crash in early 2020 due to the pandemic. The alignment of a sharp rise in negative news with the market drop highlights the potential of using sentiment indicators to detect early signs of market stress. The bottom row plot shows that the neutral sentiment appears more muted and less correlated with index movement. A decline in neutral news begins around 2020, perhaps reflecting a shift toward more emotionally polarized or impactful reporting during volatile periods.

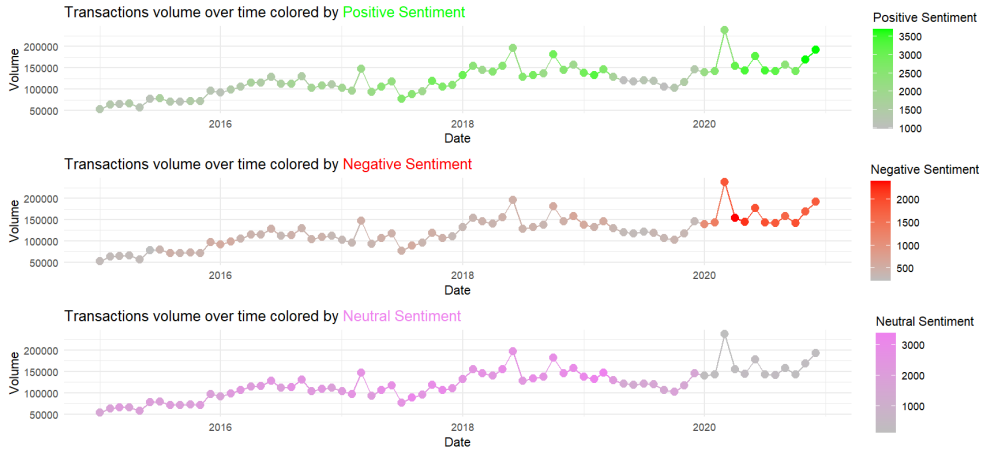



**Figure 3.21: Purchase volume time series, coloured by monthly counts of sentiment-tagged financial news articles: positive (top), negative (middle), and neutral (bottom)**

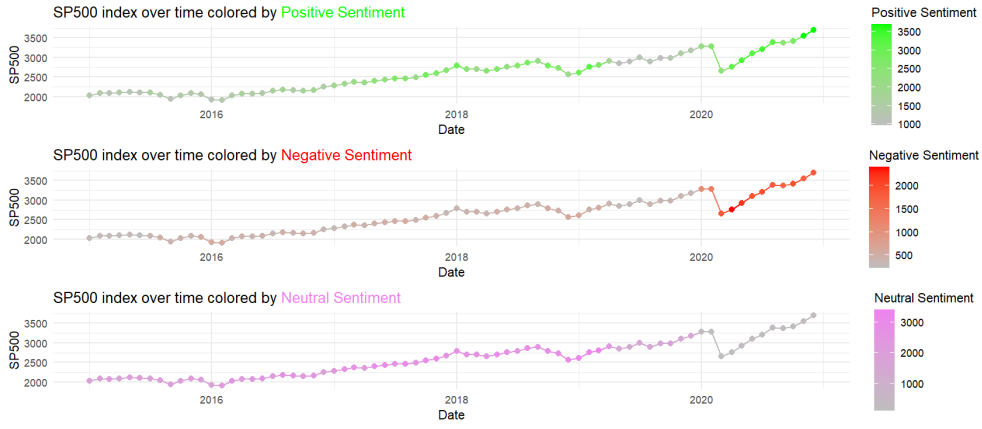



**Figure 3.22: S&P 500 index over time, coloured by monthly counts of sentiment-tagged financial news articles: positive (top), negative (middle), and neutral (bottom)**

While this analysis is exploratory, it illustrates how integrating qualitative news sentiment with quantitative market indicators can provide valuable context for understanding investor behaviour. This type of visual analysis gives us valuable clues; however, with more time and resources, if we could model these relationships properly, we could dig deeper using machine learning models to quantify the predictive power of these sentiment features. Sentiment could be a very useful predictor in forecasting investment trends.

## 3.6 Next steps

As we have emphasized on several occasions, the shortage of genuine, high-quality observations makes it difficult for our models to generalize. This limitation is by no means a justification for abandoning the project.

The aim of the present section is therefore twofold: (i) to provide a concrete, step-by-step protocol for deploying our models in an operational environment, and (ii) to draw up the functional specifications for software that can automate both the pricing task and the detection of influential covariates.

When practitioners seek to exploit the outcome of a comparative study, the reflex is often to copy-and-paste the model that achieved the "best" score. Yet the comparative exercise that we did was never intended to uncover a one-size-fits-all "magic" model. Its role was merely to demonstrate feasibility by experimenting with several, sometimes approximate, modelling approaches with approximate data.

Adopting a copy-and-paste strategy is risky for at least three fundamental reasons:

- Functional transformations are not universally appropriate (see the example involving genuinely linear data).
- A model's notion of "validity" is never absolute.
- Automating the selection of the "right model" is intrinsically difficult.

### 3.6.1 Limits of data transformation

In each of our candidate models, we began by applying a logarithmic transformation. This was not done for aesthetic reasons: empirical evidence suggests that the raw series exhibit exponential growth. The purpose of the log is to smooth the variance, standardize magnitudes, and reveal underlying trends—in short, to stabilize the time series prior to estimation. This step is particularly crucial for ARIMA-type models, which require the series to be stationary before estimation. A time series is deemed *stationary* when its global level (mean) and dispersion around that level (variance) remain constant across time.

Without such preliminary processing, an ARIMA specification will perform poorly.

Conversely, applying a log transform mechanically to data that are already stationary invariably degrades performance.

#### Illustrative experiment.

Consider a well-known stationary dataset: the average (over the year) weekly earnings of employees in public administration (in CAD) from 2001 to 2023.

Since we seek to apply our models to BNCD's real data, it makes sense to perform this illustrative experiment on a dataset that is completely different from the one we used for the comparative analysis of this report.

Let us compare two forecasting strategies:

**Model 1:** simple linear regression on the raw data, trained on the 2001–2015 window and extrapolated to 2023:

$$Y_t = a\,t + b, \quad t \in [2001,\, 2015].$$

**Model 2:** linear regression on log-transformed data, back-transformed to the original scale, trained on the same window:

$$Y_t = K\,\exp(Lt), \quad t \in [2001,\, 2015].$$

A visual inspection shows that the basic linear regression adheres far more closely to the observed values than its log-based counterpart.

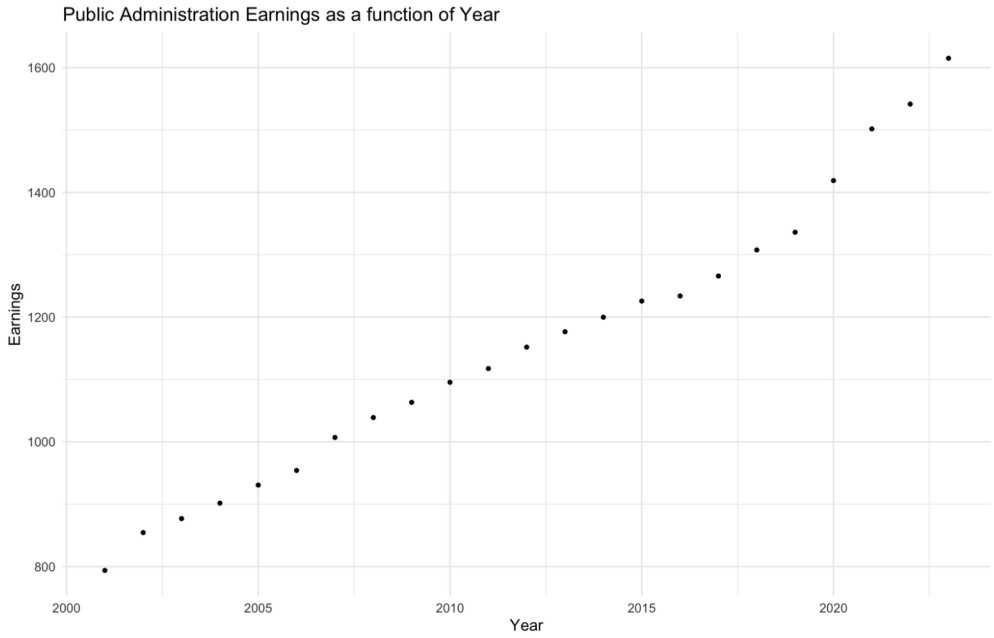



**Figure 3.23: Public administration earnings as a function of year (2001–2023)**

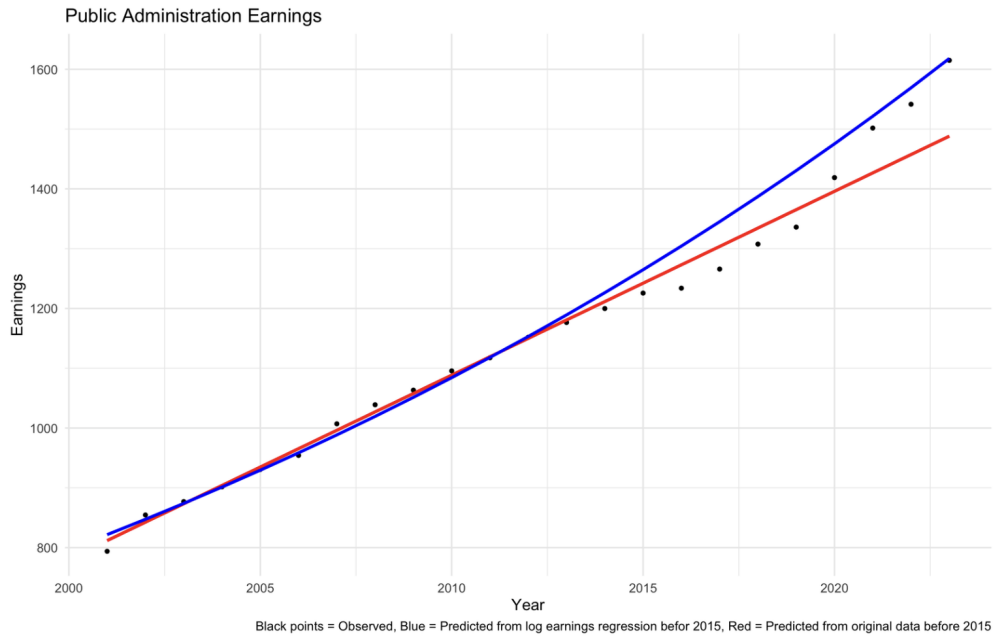



**Figure 3.24: Models 1 and 2: linear vs. log-transformed regression fittings**

### Moral.

Before applying any transformation, one must assess its relevance.

Should BNCD's operational data turn out to be stationary—a plausible scenario if, for instance, the bank's clientele consists of prudently risk-averse investors—the forecasts produced on log-scaled data would be misleading.

### 3.6.2 Model validation

Before any model is used, its underlying validity assumptions must be verified in order to establish reliability. Only after each model has passed its own diagnostic tests should models be compared against one another.

**Assumption checking.**

For the three main models placed in competition, the validity criteria can be summarized as follows (see [1] for SARIMAX and additive Holt–Winters assumptions, and [7] for SVR):

**Forecast models**

| Test \ Model | SARIMAX | Additive | SVR |
|---|---|---|---|
| Mean Square error | ✓ | ✓ | ✓ |
| Stationarity | ✓ | | |
| Seasonality | ✓ | ✓ | |
| White noise | ✓ | ✓ | ✓ |

**Features selection**

| Test \ Method | AIC | Sentiment Analysis |
|---|---|---|
| Mean Square error | ✓ | |
| Visualisation | | ✓ |

**Figure 3.25: Summary of forecast models and feature selection**

SARIMAX

- The series must be stationary or, at a minimum, must admit a transformation that renders it stationary.
- Seasonality must be present; otherwise, ARIMAX is more auspicious.
- Residuals must be free of autocorrelation.

Additive Holt–Winters

- No stationarity requirement.
- Seasonality must be present since it also estimates the seasonal part.
- Residuals must be free of autocorrelation.

Support Vector Regression (SVR)

- A single requirement: uncorrelated residuals.

For each model, the root-mean-square error (RMSE) is the natural metric that measures predictive discrepancy relative to actual data. If a single quantitative criterion is needed for model ranking, RMSE is an appropriate choice.

**Identification of influential variables.**

Traditional methods remain the most effective:

- A correlation matrix, produced with minimal effort, offers an immediate visual map of covariates that exhibit strong linear relationships with transaction volume.

- The Akaike Information Criterion (AIC) supplies a hierarchical ranking of variables according to impact.
- A descriptive cross-analysis of transaction volume versus press-derived sentiment scores enables the rapid identification–at a glance–of the sentiments that materially affect the market.

### 3.6.3 Functional specifications

A prototype software solution tailored to BNCD's needs could adopt the following workflow:

Dashboard initialisation

- The user is prompted to specify, *a priori*, a tolerable average forecast error—for instance, an RMSE ceiling of 20 %.

Data ingestion

- An upload button accepts files that have been pre-formatted according to the system schema.

Automated modelling pipeline

- The software determines the most appropriate transformations for the uploaded data;
- It selects the pricing model that satisfies its own validity assumptions;
- It identifies the features with the strongest influence on the target variable.

Output generation

- The predicted value(s), the chosen model (optionally revealed), and the most influential covariates are displayed;
- An additional button offers access to the full comparative analysis.

Exception handling

- If no candidate model can promise an RMSE below the user-defined threshold, an error message is returned.

Selecting an appropriate model is, quite simply, essential.

## 3.7 Conclusion

This project addressed two challenges proposed by Banque Nationale Courtage Direct (BNCD): modelling their transaction volume and explaining discrepancies between their current model and the observed volume. However, we did not have access to BNCD's actual transaction data or their existing model, which posed significant limitations.

To work around the absence of transaction data, we used a proxy: the total volume of foreign equity purchases by Canadian investors, as reported by Statistics Canada. It is important to note that this proxy represents a much larger market compared to BNCD's clients. Moreover, the broader market is dominated by institutional investors, whereas BNCD primarily serves individuals, whose investment behaviour differs substantially. Additionally, the proxy data was not directly adjusted for inflation, which could potentially affect the interpretation of transaction volume trends, particularly in time series analysis. However, a preliminary verification suggested that inflationary effects are not a major concern over the study period. Consequently, our findings should be interpreted with caution, and the models developed in this study should be re-evaluated using BNCD's proprietary data.

Despite these limitations, our analysis showed that both machine learning and time series models were effective in capturing market-wide trends in purchase volumes. Time series models performed better for short-term forecasts (up to six months), while machine learning methods offered improved accuracy over longer horizons. In addition, machine learning models provided greater interpretability,

offering insights into market movements and investor behaviour, thereby supporting BNCD's second objective.

In summary, while the proposed methods show promise, their applicability should be reassessed using BNCD's actual data. We recommend a follow-up analysis with the true transaction volumes to validate model performance and ensure alignment with BNCD's operational context.

## Bibliography

[1] Yves Aragon. Séries temporelles avec R méthodes et cas. Université de Toulouse 1–Capitole, 2011.

[2] Mariette Awad and Rahul Khanna. Support vector regression. In Efficient learning machines: Theories, concepts, and applications for engineers and system designers, pages 67–80. Springer, 2015.

[3] Scott R. Baker, Nicholas Bloom, and Steven J. Davis. Economic policy uncertainty index: Methodology. https://www.policyuncertainty.com/methodology.html, 2016. Accessed: 2025-09-25.

[4] Leo Breiman. Random forests. Machine learning, 45:5–32, 2001.

[5] Kevin Gurney. An introduction to neural networks. CRC press, 2018.

[6] Statistics Canada. Table 36-10-0028-01: International transactions in securities, portfolio transactions in canadian and foreign securities, by type of instrument and issuer, monthly. https://www150.statcan.gc.ca/t1/tbl1/en/tv.action?pid=3610002801, 2025. Accessed: 2025-06-02.

[7] Vladimir Vapnik. The nature of statistical learning theory. Springer science & business media, 2013.

[8] Simon N Wood. Generalized additive models: an introduction with R. Chapman and Hall/CRC, 2017.

# 4 IVADO: Characterization of the research community in data valorization through collaboration analysis

**Faramarz Farhangian** [a]

**Gilles Caporossi, coordinator** [b, c]

**Helen Samara Dos Santos** [d]

**Junya Wang** [e]

**Mariam Tagmouti** [f]

**Ovide Kuichua** [g]

**Thierno Mamadou Baldé** [h]

[a] ETS

[b] HEC Montréal

[c] GERAD

[d] Memorial University of Newfoundland

[e] Université de Montréal

[f] IVADO

[g] Polytechnique Montréal

[h] Université de Bretagne Occidentale

**October 2025**
**Les Cahiers du GERAD**

**Abstract:** *This report presents a structured, data-driven approach for expanding and characterizing IVADO's research network through the analysis of collaborations and researcher expertise. The project addresses two core challenges: (1) identifying new researchers and integrating them into IVADO's network, and (2) recommending the most suitable researchers for a given industrial project. To tackle the first challenge, we developed a web scraping pipeline targeting a specific academic conference (e.g., NeurIPS) and combined it with affiliation matching. To infer researcher expertise, we explored web scraping and evaluated several natural language processing (NLP) techniques, including keyword extraction (KeyBERT), skill tagging (SkillNER), and prompt-based large language models. For the second challenge, we embedded both researcher profiles and project descriptions into a shared semantic space using Sentence-BERT and measured cosine similarity to rank potential matches. Our results demonstrate the feasibility of integrating new academic researchers into IVADO's network and matching them to relevant projects.*

**Keywords:** Researcher profiling; natural language processing; web scraping; academic network analysis; skill extraction; semantic similarity; project-researcher matching

## 4.1 Introduction

The Institute for Data Valorization (IVADO, *Institut de valorisation des données*) is an interdisciplinary, cross-sectoral consortium dedicated to research, training, and knowledge mobilization in the service of sustainable industrial transformation. IVADO's mission is to support the development of a robust, ethical, and collaborative ecosystem for digital intelligence through interdisciplinary research, training, and innovation. This includes major initiatives such as $\mathrm{R}^{\mathbf{3}}$AI (Robust, Reasoning, and Responsible AI), which aim to advance trustworthy artificial intelligence across sectors. Based in Quebec and supported by the Canada First Research Excellence Fund, IVADO brings together academic institutions, industry partners, and public organizations to foster innovation in data science, artificial intelligence, and operations research.

There is an ongoing need for IVADO to grow and strengthen its researcher network in a way that aligns with its interdisciplinary mission. This includes both identifying new academic researchers whose expertise is relevant to IVADO's focus areas and effectively connecting these researchers to industrial or institutional projects.

This report addresses two interrelated challenges in this context. First, how can IVADO proactively discover and assess new potential contributors using publicly available research data? Second, how can it algorithmically recommend the most relevant experts for a given project, based on research interests, technical skills, and prior outputs?

To address the dual challenge of expanding IVADO's researcher network and improving project-researcher alignment, we adopted a stepwise approach combining different techniques. Our first focus was researcher discovery: we began by identifying conferences that would attract researchers working in relevant fields. Among these, NeurIPS 2024 was selected as a first test case to develop a proof of concept. We built a web scraping tool to gather author names, affiliations, and publication metadata. These data were then processed through name normalization and institutional matching to identify researchers with potential ties to the IVADO network. After that, to characterize the expertise of each new researcher, we tested several NLP-based strategies. KeyBERT was used to extract relevant keyphrases from abstracts, while SkillNER applied named entity recognition to identify predefined technical terms. We also explored web scraping and prompt-based extraction with large language models to generate structured skill outputs based on article content. These methods were compared, and their outputs were stored in a central database of researcher profiles.

For project-to-researcher matching, we embedded both project descriptions and researcher expertise into a shared semantic space using a transformer-based model (Sentence-BERT). We then measured cosine similarity to assess relevance and generate ranked recommendations. This embedding-based approach enables multilingual understanding and captures thematic connections beyond exact keyword matches.

Figure 4.1 illustrates a practical implementation of the workflow of both challenges, with the prompt-based approach for identifying expertise, which yielded complete and clear results with minimal tuning.

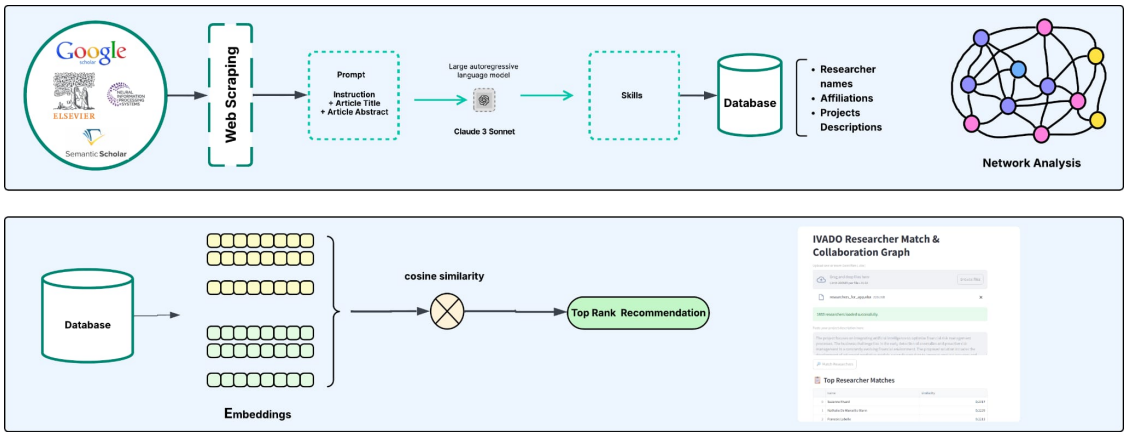



**Figure 4.1: Overview of the researcher network expansion and matching pipeline developed for this project**

Our approach demonstrates the feasibility of expanding IVADO's researcher database and matching academic expertise to industrial needs in a scalable and interpretable way.

## 4.2 Datasets and data sources

Our project drew on both internal datasets provided by IVADO and publicly available academic data. The primary data sources are as follows:

**IVADO Internal Researcher Datasets** (two Excel files):

- The first dataset includes researcher names, institutional affiliations, personal websites, and a list of expertise (generated using ChatGPT).
  *Columns:* Nom, Prénom, Université, Site web, Expertises (généré par ChatGPT).
- The second dataset includes names, institutions, links to Google Scholar and Microsoft Academic profiles, and extracted skills from each source, along with a column of internally curated skills provided by IVADO.
  *Columns:* Last name, First name, Institution, Google Scholar profile, Microsoft Academic profile, Skills from Google Scholar, Skills from Microsoft Academic, Skills from IVADO.

**Conference Publications**: We scraped metadata and paper content from the NeurIPS conference website. Extracted data included paper titles, abstracts, author lists, and full-text PDFs, which were parsed to extract affiliations and research topics.

**Web-Sourced Academic Data**: To improve the completeness of the researcher profiles, we scraped additional data from the web from OpenAlex [2] to expand profiles with expertise keywords and research areas.

**IVADO Project Description**: We were provided with a sample project description, which served as an input for testing and evaluating the researcher–project matching process.

## 4.3 Identifying new researchers

One of the core objectives of the project was to identify new researchers who could be integrated into IVADO's network. To achieve this, we developed a pipeline using NeurIPS 2024 as a primary test case.

The NeurIPS proceedings were accessed from https://papers.nips.cc/paper_files/paper/2024. This approach can be generalized to other years.

The pipeline proceeds as follows:

1. **Scrape metadata from NeurIPS proceedings**:
   - Extract paper titles, URLs, and author names.

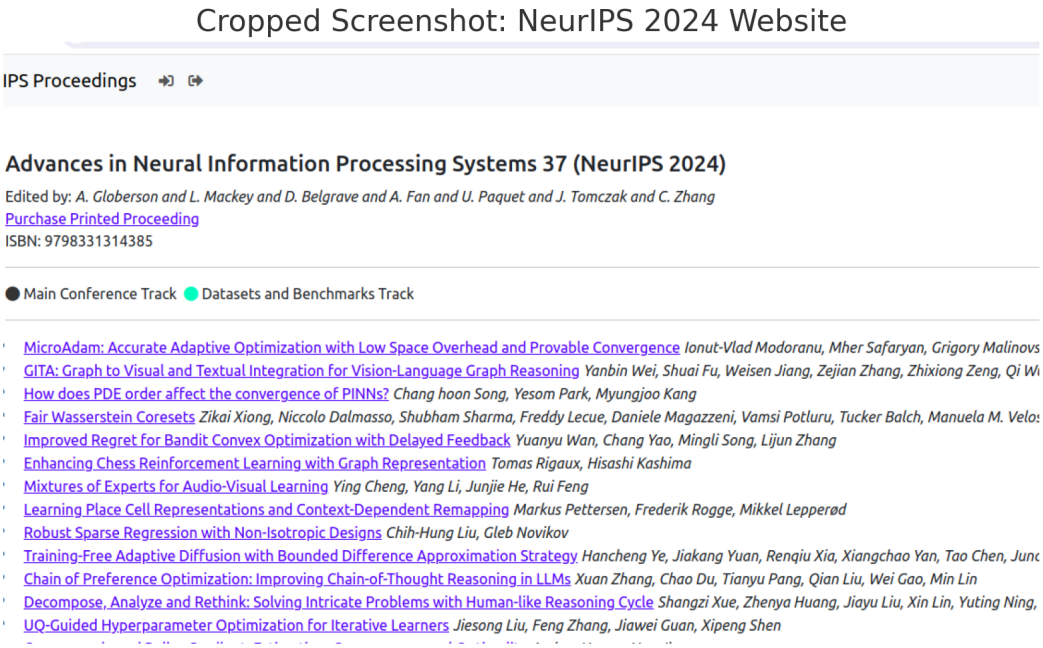



**Figure 4.2: Example of the NeurIPS 2024 proceedings page used in our web scraping pipeline**

2. **Extract details from each paper page**:
   - Scrape abstracts.
   - Download and parse paper PDFs to extract author affiliations.
3. **Match institutions**:
   - Identify whether any affiliations correspond to one of the five IVADO member universities.
4. **Integrate researchers into the IVADO network**:
   - **Option A** (inclusive):
     - For each paper with at least one author from one of the IVADO-affiliated universities:
       * Add all authors from IVADO-affiliated universities to the IVADO research network.
       * Optionally include co-authors from selected external institutions (e.g., University of Toronto), regardless of existing network connections.
   - **Option B** (targeted):
     - For each paper with at least one author already in the IVADO researcher database:
       * Add only the co-authors who are not yet in the IVADO network.

* Optionally include co-authors from external institutions who have a collaborative link to IVADO researchers.

These methods offer both broad and targeted approaches to expanding the network, focusing on recent high-impact research and leveraging institutional affiliations to identify researchers aligned with IVADO's areas of interest. Examples of researchers identified using these methods are presented in Section 4.6.

## 4.4 Extracting and representing expertise

Once a new researcher is identified, the next step is to infer their research expertise in a structured and comparable way.

To identify both broad and specific skills for newly identified researchers, we explored several strategies: web scraping, prompt-based skill extraction, and topic modelling. An additional approach (skill matching and ranking) is proposed as a direction for future work.

**Web Scraping:** As a proof of concept, we retrieved academic metadata from OpenAlex to illustrate how publicly available data can be used to characterize a researcher's interests.

**Prompt-Based Skill Extraction:** We applied prompt engineering techniques using a large language model (Claude 3 Sonnet) to infer topics and skills based on the title and abstract of a research paper. This method produced structured outputs listing both broad research areas and specific technical skills.

**Topic Modelling:** We extracted granular skills from publications using keyword extraction techniques. In particular, we used KeyBERT, which leverages sentence embeddings from a pretrained transformer model (specifically, `all-mpnet-base-v2`) to identify meaningful keyphrases from research abstracts.

We also tested SkillNER, a named entity recognition tool designed to extract predefined skill mentions from text. However, using the default skill taxonomy, it often returned few or no results. A custom taxonomy tailored to the data science and AI domains would likely yield better performance.

**Skill Matching and Ranking (Future Work):** Although not implemented, we considered using sentence embeddings and cosine similarity to compare new researchers to existing IVADO members. This could help infer relevant skills for the new researchers. We discuss this approach further in Section 4.7.

### 4.4.1 KeyBERT-based pipeline

KeyBERT is a keyword extraction tool that uses pretrained transformer models to identify the most representative keyphrases in a document [1]. It works by embedding both the full text and candidate phrases into the same semantic vector space and ranking the candidates based on their cosine similarity to the overall document embedding.

In our project, we used the `all-mpnet-base-v2` model, and KeyBERT was configured to extract phrases containing one to three words (`keyphrase_ngram_range=(1, 3)`), a range chosen to capture both general topics and more specific technical terms. We also limited the candidate pool to 20 phrases (`nr_candidates=20`) to reduce processing time.

Furthermore, we used the MaxSum ranking strategy (`use_maxsum=True`), which is intended to select keyphrases that are both relevant to the document and relatively distinct from one another. However, in practice, we observed substantial repetition in the extracted outputs. For example, for a paper on privacy-preserving data release, KeyBERT returned phrases such as "data release strategies," "data release paper," "preserving data release," and "data release." This redundancy suggests that the MaxSum setting did not sufficiently diversify the selected phrases in our case. Additional post-processing or tuning would be needed to improve the uniqueness and clarity of the extracted keywords.

Our KeyBERT-based skill extraction pipeline included the following steps:

- **Data Preprocessing**: Combine the title and abstract into a single text field, convert to lowercase, and apply manual formatting corrections.
- **Extract Keyphrases**: Use KeyBERT with `all-mpnet-base-v2` to extract 1- to 3-word phrases. Apply the MaxSum similarity method to rank candidates.
- **Filter Results**: Remove phrases containing digits and overly generic terms, and manually inspect for repetitions.
- **Save Results**: Store the cleaned keyphrases in a structured dataframe for further use in expertise profiling.

### 4.4.2 SkillNER-based pipeline

SkillNER is a named entity recognition (NER) tool designed to extract mentions of technical or professional skills from text [3]. It uses a predefined taxonomy of skill terms and applies rule-based or machine learning–based techniques to identify them within a document.

In our project, we applied SkillNER to the title and abstract of each research paper to extract explicit skill mentions. The default skill taxonomy was used without modification. While straightforward to implement, the results were limited. In many cases, the output was either empty or contained only very general terms, indicating that the default taxonomy was not well-suited to academic writing in the domains of data science and artificial intelligence. This highlights the need to customize or extend the taxonomy to better reflect the vocabulary used in research contexts.

Our SkillNER-based pipeline included the following steps:

- **Data Preprocessing**: Combine the title and abstract into a single column, convert to lowercase, and apply formatting corrections.
- **SkillNER Initialization**: Run SkillNER using its default taxonomy to identify skill mentions in the text.
- **Filter Results**: Remove duplicates, digits, and generic or irrelevant terms.
- **Save Results**: Store the extracted skills in a structured dataframe for comparison with outputs from other methods.

## 4.5 Matching projects to researchers

The second objective of this project was to develop a method for recommending researchers in the IVADO network who are best suited to contribute to or lead new research projects. This required a way to compare the content of project descriptions to the expertise profiles of individual researchers in a semantically meaningful way. To this end, we built a matching pipeline based on text embedding and similarity ranking.

The first step involved preprocessing the researcher database to prepare it for semantic comparison. This included normalizing column names, converting all text fields to lowercase, removing excess whitespace and punctuation, and dropping rows with missing or empty expertise fields. Duplicate entries were also removed to ensure clean inputs and accurate matching.

Once the dataset was cleaned, we used a pretrained sentence embedding model to convert both researcher expertise and project descriptions into vector representations. Specifically, we employed the `all-MiniLM-L6-v2` model, a lightweight sentence embedding model from the SentenceTransformers library, trained using the MiniLM architecture. For each researcher, the text representing their expertise was encoded into a compact, high-dimensional vector that preserves semantic meaning. The same process was applied to the project description provided by IVADO, which was encoded into a single

vector representation by averaging token embeddings within the model's architecture. This allowed us to compare the semantic content of researcher profiles and project needs beyond simple keyword matching, capturing contextual meaning and related concepts across disciplines.

This embedding-based approach supports multilingual understanding, which is particularly valuable in IVADO's bilingual (French and English) research context. It also allows for broader generalization across disciplines and vocabulary, making it possible to identify similar areas of expertise even when they are described using different terms.

To compare researcher and project embeddings, we computed the cosine similarity between each researcher's vector and the project vector. Cosine similarity measures the angle between two vectors in high-dimensional space, yielding a value between 0 (no similarity) and 1 (perfect match). Because this metric focuses on the direction rather than the magnitude of the vectors, it is robust to variations in text length and formatting, making it well-suited for comparing researcher profiles and project descriptions that differ in length and level of detail.

Using these similarity scores, we ranked all researchers by their degree of alignment with the project description. The output of this process is a sorted list of candidates, where higher scores indicate closer semantic alignment. This ranked list enables IVADO to quickly identify researchers whose expertise is most relevant to a given project, even when exact keywords or field labels are not shared between the project and profile descriptions.

Overall, this matching framework provides a scalable, language-agnostic, and semantically rich approach to connecting projects with suitable researchers. It extends IVADO's ability to mobilize expertise across its network by recommending collaborators who may not be obviously linked to a project through keywords alone, but who are semantically well-aligned based on their research profile.

Our matching pipeline included the following steps:

- **Data Preprocessing**: Normalize researcher profile data by converting text to lowercase and removing duplicates, empty rows, and unnecessary punctuation.
- **Embedding Generation**: Use the `all-MiniLM-L6-v2` model from the SentenceTransformers library to convert researcher expertise and project descriptions into numerical vectors that represent their semantic content.
- **Similarity Computation**: Compute cosine similarity between the project vector and each researcher's expertise vector.
- **Ranking**: Sort researchers by similarity score to produce a ranked list of recommended matches.
- **Save Results**: Store the ranked matches in a structured output for interpretation or further processing.

## 4.6 Results and evaluation

### 4.6.1 Identifying new researchers from NeurIPS 2024

Initial runs of the pipeline successfully identified multiple researchers not previously listed in the IVADO researcher database. These individuals are candidates for integration based on co-authorship with known members and/or confirmed institutional affiliation, depending on whether Option A or Option B is selected.

**Option A** involves checking whether at least one author of a paper is affiliated with an IVADO member university. If so, all co-authors from that paper are considered for inclusion in the IVADO network. This requires downloading the full PDF of each paper and parsing it to extract author affiliations from the text.

As an example of Option A, consider the paper *"Geometry of Naturalistic Object Representations in Recurrent Neural Network Models of Working Memory"* authored by Xiaoxuan Lei, Takuya Ito, and Pouya Bashivan. None of the authors is currently listed in the IVADO researcher database. However, both Lei and Bashivan are affiliated with IVADO member institutions — McGill University and Université de Montréal — and would therefore be candidates for inclusion under the Option A strategy. Their co-author, Author affiliations are shown in Table 4.1.

**Table 4.1: Example application of Option A: authors from a NeurIPS 2024 paper with affiliations to IVADO member institutions**

| Author | Affiliation(s) |
|---|---|
| Xiaoxuan Lei | Department of Physiology, McGill University<br>Mila, Université de Montréal |
| Takuya Ito | IBM Research, Yorktown Heights, NY, USA |
| Pouya Bashivan | Department of Physiology, McGill University<br>Mila, Université de Montréal |

**Option B** by contrast, focuses on direct name-based matching with known IVADO researchers. Instead of extracting affiliations from PDFs, we compare author names scraped from the NeurIPS 2024 website with those listed in the IVADO-provided datasets. With this approach, we only need to download the PDFs for which we know at least one author's name is in IVADO's network. This approach relies on normalization (e.g., converting to lowercase, stripping accents) to reduce formatting inconsistencies.

It is important to note that name-based matching alone can lead to false positives due to homonyms. For instance, while "Jian Tang" is a known IVADO researcher, the Jian Tang listed as an author of the paper *"EDT: An Efficient Diffusion Transformer Framework Inspired by Human-like Sketching"* — alongside Xinwang Chen, Ning Liu, Yichen Zhu, and Feifei Feng — is affiliated with an institution outside the IVADO network. This highlights the need to combine name matching with affiliation checks to ensure accurate identification of researchers. Consequently, downloading the PDF of the paper to extract and verify institutional affiliations remains a valuable step in the pipeline, even when using a more efficient name-based approach.

In total, we found **46 researchers already present in the IVADO database** who authored one or more papers at NeurIPS 2024. This confirms the utility of our name-matching approach.

For example, in the paper *"SR-CACO-2: A Dataset for Confocal Fluorescence Microscopy Image Super-Resolution"*, the presence of Eric Granger — already listed in the IVADO database — signals the potential inclusion of his co-authors in the extended network. As shown in Table 4.2, Soufiane Belharbi, Mara KM Whitford, Phuong Hoang, Shakeeb Murtaza, and Luke McCaffrey are candidates for addition based on institutional affiliation. Granger is the only author in this list currently affiliated with IVADO.

Among the co-authors, Luke McCaffrey is a professor at McGill University and represents a strong candidate for inclusion in IVADO's network expansion. Further inspection revealed that some of the other candidates are postdoctoral researchers rather than permanent faculty members. This highlights a limitation of the current approach: while co-authorship and institutional affiliation are helpful signals, they may not accurately represent long-term professional roles. Incorporating a final step for career-stage filtering could improve the quality and stability of the expanded network.

Although only selected examples are presented here, the pipeline generated additional candidates for the IVADO network expansion. Option B is more efficient and aligns closely with IVADO's current researcher database structure. Option A, although broader in scope, requires more processing and assumes relevance based solely on institutional affiliation.

**Table 4.2: Example application of Option B: authors from a NeurIPS 2024 paper with an IVADO-affiliated co-author (Eric Grange)**

| Author | |
|---|---|
| Soufiane Belharbi | LIVIA, ILLS, Dept. of Systems Engineering, ETS Montreal |
| Mara KM Whitford | Goodman Cancer Institute, McGill University, Montreal |
| | Dept. of Biochemistry, McGill University, Montreal |
| Phuong Hoang | Goodman Cancer Institute, McGill University, Montreal |
| Shakeeb Murtaza | LIVIA, ILLS, Dept. of Systems Engineering, ETS Montreal |
| Luke McCaffrey | Goodman Cancer Institute, McGill University, Montreal |
| | Dept. of Biochemistry, McGill University, Montreal |
| | Gerald Bronfman Dept. of Oncology, McGill University, Montreal |
| Eric Granger | LIVIA, ILLS, Dept. of Systems Engineering, ETS Montreal |

### 4.6.2 Extracting and representing expertise

The two examples below illustrate how the topic modelling techniques described in Section 4.4 performed in practice, alongside external benchmark labels from Google Scholar, Microsoft Academic, and IVADO's own skills list.

#### Example 1

**Paper Title:** *Measuring privacy/utility tradeoffs of format-preserving strategies for data release*

**Abstract:** *In this paper, we introduce a novel approach to evaluate the risk of re-identification of individuals associated with format-preserving data release strategies, focusing on three strategies: data minimization (i.e. through data removal using random sampling and data Shapley values), data anonymization (i.e. through -anonymity), and data synthesis (i.e. through CTGAN and TVAE generative models). More precisely, our approach consists in simulating a security game in which (1) an attacker performs singling-out attacks as outlined in data protection regulations and (2) an evaluator scores attacks based on the linkability of records and the information gain obtained by the attacker. In addition, we further enhance our approach by simulating attacks as a cooperative game, in which the value of the attackers' information resources is determined using the Shapley value borrowed from game theory.*

- ***KeyBERT****: data release strategies, privacy utility, data release paper, preserving data release, data release*
- ***SkillNER****: game theory*
- ***Google Scholar****: Graph theory, computer systems*
- ***IVADO****: Complex networks, operation of data, algorithms, combinatorial optimization, artificial intelligence*
- ***Microsoft Academic****: Computer science, mathematics, theoretical computer science, heuristic, algorithm, centrality, graph, graph theory, mathematical optimization, artificial intelligence, variable neighborhood search, discrete mathematics, vertex, combinatorics, metaheuristic, enumeration*

KeyBERT extracted phrases closely tied to the paper's language, but it lacked semantic diversity, as evidenced by the redundant appearance of multiple variants of "data release". SkillNER returned only one match ("game theory"), despite the technical nature of the abstract. In contrast, the benchmark sources produced a broader set of concepts, including algorithmic and mathematical themes that were not captured by the NLP models without additional tuning.

#### Example 2

**Paper Title:** *Random trees have height O(n)*

**Abstract:** *We obtain new nonasymptotic tail bounds for the height of uniformly random trees with a given degree sequence, simply generated trees and conditioned Bienaymé trees (the family trees of*

*branching processes) in the process settling three conjectures of Janson (Probab. Surv. 9 (2012) 103–252) and answering several other questions from the literature. Moreover, we define a partial ordering on degree sequences and show that it induces a stochastic ordering on the heights of uniformly random trees with given degree sequences. The latter result can also be used to show that sub-binary random trees are stochastically the tallest trees with a given number of vertices and leaves (and thus that random binary trees are the stochastically tallest random homeomorphically irreducible trees (Acta Math. 101 (1959) 141–162) with a given number of vertices).*

- ***KeyBERT****: stochastic tallest, binary trees, heights uniformly random, generated trees, simply generated trees*
- ***SkillNER****: family tree, binary tree*
- ***Google Scholar****: Probability, combinatorics*
- ***IVADO****: Applied mathematics, discrete mathematics, probability, statistics*
- ***Microsoft Academic****: Zero, combinatorics, bounded function, limit, mathematics, random graph, minimum spanning tree, complete graph, discrete mathematics, degree, graph, conjecture, random tree, tree, random walk, vertex, brownian motion, upper and lower bounds, sequence*

In this case, KeyBERT returned more specific technical phrases, including tree structures and stochastic descriptors. SkillNER again produced only a few domain-relevant terms. Compared to the vocabulary in external sources, both KeyBERT and SkillNER were limited by their reliance on surface-level text and default parameters.

Overall, these examples demonstrate that methods like KeyBERT and SkillNER offer a useful starting point for automated expertise extraction. Future work should focus on improving keyword quality through model selection and post-processing techniques, as well as adapting SkillNER's taxonomy to better reflect the academic language used in domains such as data science, artificial intelligence, and related areas.

#### Prompt-based skill extraction

We also explored the use of large language models (LLMs) for skill extraction via prompt engineering. The idea is to leverage the general knowledge and reasoning capacity of these models to infer relevant research topics and skills directly from an article's title and abstract.

We designed a prompt that instructs the model to act as a domain expert tasked with identifying both high-level topics and the technical skills required to carry out the research described in the article. The prompt was structured to produce a JSON-style output containing two fields: `"topics"` and `"skills"`.

- **Sample Prompt:**

  > You are an expert in the field, responsible for determining the topic and skills required to carry out the research presented in the article, based on the title of the abstract and your own knowledge of the field.
  > Output Format: { "topics": [...], "skills": [...] }
  > Here is the input: {title and abstract}
  > Write your output without any additional comments.

- **Sample Output (for a paper on random tree height):**

```
{
"topics": ["Random Tree Height Analysis"],
"skills": [
"Probability Theory",
"Combinatorics",
"Graph Theory",
```

```
"Stochastic Processes",
"Mathematical Conjectures",
"Nonasymptotic Analysis",
"Degree Sequences",
"Branching Processes",
"Tree Structures",
"Mathematical Proofs"
]
}
```


This method produced rich and coherent outputs with both broad topics and fine-grained technical skills. While more costly to run at scale, it offers a flexible and interpretable alternative to unsupervised methods, such as KeyBERT and SkillNER. Further evaluation is needed to compare its accuracy and domain alignment with benchmark taxonomies.

## 4.7 Discussion and future work

As observed in the results, some researchers identified through co-authorship and affiliation may not align with IVADO's long-term strategic goals. A potential improvement to the researcher expansion pipeline involves a post-processing step to filter out collaborators who are students or postdoctoral researchers. While co-authorship and institutional affiliation provide useful signals, many authors in large conferences such as NeurIPS are temporary collaborators whose inclusion may be less aligned with IVADO's long-term strategic goals. Future work could explore methods for identifying and excluding such cases, either by scraping publicly available information (e.g., institutional webpages or academic profiles) or by querying language models to assess whether a given author is likely to hold a faculty or permanent research position. Incorporating this step could help improve the overall quality and long-term relevance of the expanded network.

While KeyBERT provided a way for extracting keyphrases based on semantic similarity, the quality of the results reflected our specific design choices. Future work could explore the use of alternative transformer-based models to assess whether they produce more diverse or accurate keyphrases. Also, additional post-processing could help reduce redundancy and filter out overly generic terms in the extracted output.

One key limitation of our current implementation of SkillNER is its reliance on a generic, predefined skill taxonomy, which was not well aligned with the specialized vocabulary found in academic research papers from these specific proceedings. To improve performance, future work should focus on developing a domain-specific taxonomy tailored to data science and artificial intelligence. A promising direction would be to leverage IVADO's own curated list of research skills as a customized taxonomy for SkillNER. This could increase both the coverage and relevance of the extracted skills when profiling researchers.

Although not implemented in this project, another future direction is to infer a new researcher's skills based on their co-authorship links and shared publications with researchers already in the IVADO network. For example, suppose Researcher B is added to the network because they co-authored a paper (P) with Researcher A, whose skill profile is already known. Since both authors contributed to the same work, it is reasonable to assume that at least some of their expertise overlaps. To formalize this, we propose embedding the title and abstract of paper P into a semantic vector space using a pretrained Sentence-BERT model. We then compute the cosine similarity between the paper embedding and the embeddings of Researcher A's known skills. The skills most semantically aligned with the paper content are likely to represent shared expertise between A and B. These overlapping skills could then be assigned to Researcher B with a reasonable degree of confidence, thereby seeding their profile with inferred competencies based on collaborative evidence. This method could serve as an efficient way to bootstrap skill profiles for new researchers who were brought into IVADO's network.

## Bibliography

[1] Maarten Grootendorst. Keybert: Minimal keyword extraction with bert. https://github.com/MaartenGr/KeyBERT, 2020. Accessed: 2025-06-30.

[2] OurResearch. Openalex: The open catalog of the global research system. https://openalex.org, 2022. Accessed: 2025-06-30.

[3] skillNer contributors. skillner python package. https://pypi.org/project/skillNer, 2020. Accessed: 2025-06-30.