Forecasting Philippine Trade Activity

using Multi-task Learning

**[Do not include author details in the initial submission to facilitate a double-blind review]**

**John Smith1 ** (ORCID hyperlink linked to the image)**, Rose Thompson2, Emma Baker3**

1Department, University, Country, 2Department of Economics, Universitat Politècnica de València, Spain, 3Department, University, Country.

How to cite: Smith, J.; Thompson R.; Baker E. 2025. Paper Title. In: 7th International Conference on Advanced Research Methods and Analytics (CARMA 2025). Rome, 2-4 July 2025. https://doi.org/10.4995/CARMA2025.2025.\* [To be completed by the publisher]

***Abstract***

*Between 100 and 150 words, briefly specifying the aims of the work, the main results obtained, and the conclusions drawn.*

***Keywords:*** *Keyword1; keyword2; keyword3 (maximum 6).*

# 1. Introduction

Nowcasting models is an important tool that help guide policy makers in recommending relevant, impactful and timely policies for the economy. Agencies such as the central banks continues to explore novel forecasting models to supportpolicy formulation. Philippine import/export is one of key economic indicator of interest to forecast as it is one of the components to the economic growth (GDP). Due to network effects of the supply chain and limited data sources of trade in the Philippines, developing an accurate model remains a challenge. Moreover, official trade statistics are published with a significant lag (two months) – making it challenging for policymakers to provide timely and data-driven policies.

In this study, we develop a forecasting model of Philippine Import and Export activity of the Philippines. We investigate the use of advanced machine learning models and compare its performance to a set of benchmark linear models. In particular, we explore the use of Multi-task Learning (MTL) paradigm and Convolutional Neural Network. MTL allows the model to learn non-linear patterns of import and export activity across ASEAN region. We also explore the addition of novel indicators on supply disruption (supply), prices (demand), and trade uncertainty to improve model performance and interpretablity.

# 2. Review of Related Literature

Multi-task learning (MTL) is a machine-learning (ML) approach aimed to improve performance of multiple tasks by training them together (Caruana 1997). The mechanism is inspired by human learning where knowledge from one tasks is applied to learn related tasks (Ruder 2017; Zhang 2021). The core mechanism is learning tasks in parallel using a shared representation. The key assumption of MTL is that tasks are related, i.e., tasks use the same features to make a decision.

## 2.1. Caruana (1997)

In a seminal work, Caruana (1997) demonstrated the application of MTL in ML models such as Back-propagating Neural Networks (NNs), K-Nearest Neighbor (KNN) regression, and Decision Trees. For Back-propagating Neural Networks, MTL is achieved by sharing hidden layers among tasks while maintaining task-specific output layers (many-to-many approach). For KNN regression, MTL learns a shared distance metric/attribute weights across multiple related tasks. The weighted sum of the main and extra tasks is used as a loss function. Model training for MTL follows similar train-test framework with single-task learning. However, one key aspect in MTL is that additional features that are cannot be used as inputs at run-time (e.g., future measurements) are made available and learned during training. These future measurements are extra MTL outputs to better inform the learning process. In the study, Caruana (1997) reported that with limited training set size, MTL yielded 5-10 % better than Single-Task Learning (STL).

## 2.2. Ruder (2017)

Ruder (2017) discussed the implementation and underlying mechanism of MTL with deep neural networks. The author discussed the two common approach to MTL namely, hard parameter and soft parameter sharing in NNs. Novel developments of parameter sharing was also discussed such as: cross-stich networks and sluice networks.

For hard parameter sharing (HPS), NN models have shared hidden layers while maintain separate, task-specific output layers. The main benefit of HPS is its regularization - ability to reduce overfitting by an order of the number of *N* tasks (Baxter, 1997). MTL encourages the model to simultaneously learn general representation (weights of hidden layers) that captures all tasks.

In contrast, soft parameter sharing (SPS) utilizes anindividual model – with its own parameters for each task. Then SPS regularizes the distance between parameters – encouraging them to be similar of across all models trained for specific tasks.

## 2.3. Chen et al (2020)

Chen et al (2020) proposed two self-attention based sharing scheme for MTL in Recurrent Neural Networks (RNN). These parameter sharing schemes are: General Global Shared Attention and Hybrid Local-global Shared Attention. The authors highlighted that RNNs struggle to model long-term dependency due to vanishing/explodings gradients during training. The study proposes the use of transformer architecture built on self-attention mechanism, MTL-Trans. The results of the study show that MTL models have lower RMSE compared to STL Long-Short Term Memory (LSTM) model, i.e., ~15% improvement over STL LSTM. In addition, MTL-Trans showed marginal improvement (~2%) compared to SSP-MTL (LSTM-based MTL used as benchmark).

# 3. Methodology

## 3.1. Data Sources

The goods import and export timeseries were collected from S&P Global. The import and export series (level, in Billion USD) are at monthly frequency covering period January 2010 to June 2025 (*N* = 186 observations per country). The series are activities from several countries such as US, Asia-Pacific region, and Middle-east.

## 3.2. Baseline Statistical and ML Models via Single-Task Learning

We will utilize several statistical/regression and machine learning models as benchmarks such as Naïve, Moving Average, ARIMA, and Elastic Net models. We also utilize standard machine learning models such as SVR, Random Forest, XGBoost, and Neural Nets (RNN, GRU, and LSTM). All the models will be trained on a single-task, i.e., to forecast up to 3-month ahead PH import/export activity.

The features will be the lags of the import/export and time variables. The target variable will be the 3-month ahead PH import/export activity.

## 3.3. ML Models via Multi-Task Learning

Focus on code repository

# 4. Discussion

What is the intuition behind MTL? MTL leverages on several unique ML and statistical properties to deliver better performance. Training from extra tasks promotes inductive bias - guiding the model towards more general solution. For low sample size (e.g., quarterly series), MTL increases the sample size of the training set by integrating other tasks leading to averaging out noise patterns in individual tasks. MTL inherently performs regularization reducing overfitting by constraining the model to generalize multiple tasks.

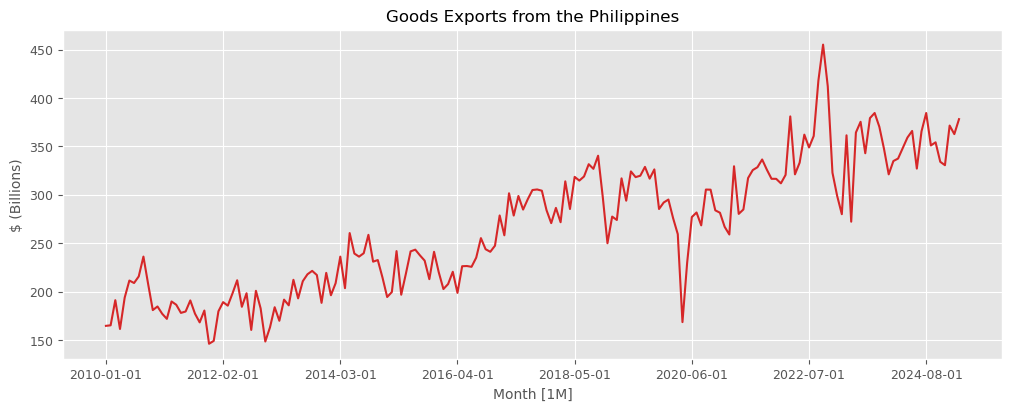
# References

Caruana, R. (1997). "Multitask Learning". Machine Learning 28. pp 41-75.

Ruder, S. (2017). "An Overview of Multi-Task Learning in Deep Neural Networks". ArXiv. https://doi.org/10.48550/arXiv.1706.05098

Chen, Z., Zhang, JEX, Sheng, H., and Cheng, X. (2020). "Multi-Task Time Series Forecasting With Shared Attention," 2020 International Conference on Data Mining Workshops (ICDMW), Sorrento, Italy, 2020, pp. 917-925, doi: 10.1109/ICDMW51313.2020.00132.

Annex. Time-series plots of PH goods imports and exports activity



A graph of a graph showing the growth of the philippines

AI-generated content may be incorrect.

Annex. Analysis roadmap

| EDA | TS features, PCA, STL decomposition  Box-Jenkins test for stationarity |
| --- | --- |
| Naïve model | Via STL decomposition  Trend – naïve  Seasonal – seasonal naïve |
| ARIMA | Auto-ARIMA (take difference if needed as pre-processing) |
| ETS |  |
| Dynamic Regression | Exog: Contemporaneous features of import/export of PH and other countries |
| NN | Include Time encoding as features |
| Transformers | Include Time encoding as features |

Annex. Exploratory Data Analysis

**Table.** Sample Statistics of the time-series (**Export**)

| Metrics | Countries | | | | | | | | | | | | Aggregate | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | PHL | USA | HKG | JPN | CHN | SGP | KOR | IND | THA | MMR | VNM | TWN | Mean | SD |
| Trend | 0.929 | 0.981 | 0.961 | 0.930 | 0.979 | 0.956 | 0.962 | 0.960 | 0.968 | 0.886 | 0.986 | 0.983 | 0.957 | 0.0293 |
| Seasonal strength | 0.461 | 0.591 | 0.790 | 0.811 | 0.832 | 0.407 | 0.528 | 0.463 | 0.611 | 0.340 | 0.652 | 0.697 | 0.598 | 0.1636 |
| x\_acf1 | 0.901 | 0.937 | 0.843 | 0.682 | 0.896 | 0.925 | 0.904 | 0.908 | 0.891 | 0.856 | 0.943 | 0.919 | 0.884 | 0.0699 |
| x\_acf10 | 6.162 | 6.854 | 5.183 | 3.486 | 6.200 | 5.896 | 6.178 | 6.553 | 6.139 | 5.713 | 7.474 | 6.737 | 6.048 | 0.9964 |
| e\_acf1 | 0.333 | 0.541 | 0.150 | 0.329 | 0.141 | 0.340 | 0.075 | 0.232 | 0.099 | 0.145 | 0.045 | -0.010 | 0.202 | 0.1579 |
| e\_acf10 | 0.302 | 0.476 | 0.164 | 0.332 | 0.229 | 0.576 | 0.072 | 0.237 | 0.232 | 0.208 | 0.093 | 0.064 | 0.249 | 0.1560 |
| Entropy | 0.433 | 0.446 | 0.500 | 0.533 | 0.460 | 0.485 | 0.489 | 0.459 | 0.490 | 0.429 | 0.402 | 0.425 | 0.463 | 0.0379 |
| Linearity | 12.049 | 11.751 | 11.878 | 7.207 | 12.174 | 9.972 | 11.502 | 11.406 | 12.004 | 11.069 | 13.078 | 11.928 | 11.335 | 1.4928 |
| Nonlinearity | 0.277 | 0.445 | 0.396 | 0.459 | 0.810 | 0.389 | 0.690 | 1.212 | 0.520 | 0.490 | 0.177 | 1.058 | 0.577 | 0.3108 |

Source: S&P Global; Author’s estimates

**Table.** Sample Statistics of the time-series (**Import**)

| Metrics | Countries | | | | | | | | | | | | Aggregate | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | PHL | USA | HKG | JPN | CHN | SGP | KOR | IND | THA | MMR | VNM | TWN | Mean | SD |
| Trend | 0.974 | NA | 0.965 | 0.944 | 0.973 | 0.955 | 0.975 | 0.959 | 0.953 | 0.923 | 0.988 | 0.957 | 0.960 | 0.0176 |
| Seasonal strength | 0.411 | NA | 0.812 | 0.518 | 0.676 | 0.419 | 0.379 | 0.303 | 0.327 | 0.404 | 0.589 | 0.499 | 0.485 | 0.1553 |
| x\_acf1 | 0.952 | NA | 0.837 | 0.888 | 0.910 | 0.925 | 0.948 | 0.931 | 0.901 | 0.892 | 0.952 | 0.872 | 0.910 | 0.0364 |
| x\_acf10 | 7.526 | NA | 5.090 | 5.221 | 6.872 | 5.849 | 7.161 | 6.588 | 6.151 | 5.605 | 7.619 | 6.199 | 6.353 | 0.8781 |
| e\_acf1 | 0.182 | NA | 0.066 | 0.141 | -0.056 | 0.402 | 0.204 | 0.225 | 0.058 | 0.317 | 0.044 | -0.174 | 0.128 | 0.1643 |
| e\_acf10 | 0.122 | NA | 0.078 | 0.206 | 0.071 | 0.586 | 0.117 | 0.195 | 0.205 | 0.241 | 0.121 | 0.317 | 0.205 | 0.1466 |
| Entropy | 0.358 | NA | 0.515 | 0.471 | 0.405 | 0.476 | 0.394 | 0.446 | 0.483 | 0.427 | 0.382 | 0.438 | 0.436 | 0.0481 |
| Linearity | 12.471 | NA | 11.788 | 7.425 | 12.116 | 9.897 | 10.984 | 10.802 | 11.118 | 9.042 | 13.098 | 11.131 | 10.898 | 1.6193 |
| Nonlinearity | 0.418 | NA | 0.412 | 0.007 | 0.650 | 0.276 | 0.230 | 1.520 | 0.311 | 1.779 | 0.118 | 1.225 | 0.632 | 0.5996 |

Source: S&P Global; Author’s estimates

**Figure.** Principal-Component Analysis of Statistical metrics

Outliers

Annex. Evaluation of model performance under Single-Task Learning (STL)

**Table.** Train/Test MAE of different models under Single-Task Learning paradigm

| Models | Train (with cross-validation) | | | Test (hold-out) | | |
| --- | --- | --- | --- | --- | --- | --- |
|  | *h* = 3 | *h* = 6 | *h* = 9 | *h* = 3 | *h* = 6 | *h* = 9 |
| Naïve |  |  |  |  |  |  |
| ARIMA |  |  |  |  |  |  |
| ETS |  |  |  |  |  |  |
| Dynamic Regression |  |  |  |  |  |  |
| RNN |  |  |  |  |  |  |
| GRU |  |  |  |  |  |  |
| LSTM |  |  |  |  |  |  |
| Transformer |  |  |  |  |  |  |

Annex. Residual Analysis for STL paradigm

*h* = 3 months

| Models | Train (with cross-validation) | | | | Test (hold-out) | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | bias | sd | D.A | is white noise | bias | sd | D.A | is white noise |
| Naïve |  |  |  |  |  |  |  |  |
| ARIMA |  |  |  |  |  |  |  |  |
| ETS |  |  |  |  |  |  |  |  |
| Dynamic Regression |  |  |  |  |  |  |  |  |
| RNN |  |  |  |  |  |  |  |  |
| GRU |  |  |  |  |  |  |  |  |
| LSTM |  |  |  |  |  |  |  |  |
| Transformer |  |  |  |  |  |  |  |  |

*h* = 9 months

| Models | Train (with cross-validation) | | | | Test (hold-out) | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | bias | sd | D.A | is white noise | bias | sd | D.A | is white noise |
| Naïve |  |  |  |  |  |  |  |  |
| ARIMA |  |  |  |  |  |  |  |  |
| ETS |  |  |  |  |  |  |  |  |
| Dynamic Regression |  |  |  |  |  |  |  |  |
| RNN |  |  |  |  |  |  |  |  |
| GRU |  |  |  |  |  |  |  |  |
| LSTM |  |  |  |  |  |  |  |  |
| Transformer |  |  |  |  |  |  |  |  |

Annex. Evaluation of model performance under Multi-Task Learning (STL)

**Table.** Train/Test MAE of different models under Multi-Task Learning paradigm

| Models | Train (with cross-validation) | | | Test (hold-out) | | |
| --- | --- | --- | --- | --- | --- | --- |
| Naïve |  |  |  |  |  |  |
| ARIMA |  |  |  |  |  |  |
| ETS |  |  |  |  |  |  |
| Dynamic Regression |  |  |  |  |  |  |
| RNN |  |  |  |  |  |  |
| GRU |  |  |  |  |  |  |
| LSTM |  |  |  |  |  |  |
| Transformer |  |  |  |  |  |  |

Annex. Residual Analysis for MTL paradigm

*h* = 3 months

| Models | Train (with cross-validation) | | | | Test (hold-out) | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | bias | sd | D.A | is white noise | bias | sd | D.A | is white noise |
| Naïve |  |  |  |  |  |  |  |  |
| ARIMA |  |  |  |  |  |  |  |  |
| ETS |  |  |  |  |  |  |  |  |
| Dynamic Regression |  |  |  |  |  |  |  |  |
| RNN |  |  |  |  |  |  |  |  |
| GRU |  |  |  |  |  |  |  |  |
| LSTM |  |  |  |  |  |  |  |  |
| Transformer |  |  |  |  |  |  |  |  |

*h* = 9 months

| Models | Train (with cross-validation) | | | | Test (hold-out) | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | bias | sd | D.A | is white noise | bias | sd | D.A | is white noise |
| Naïve |  |  |  |  |  |  |  |  |
| ARIMA |  |  |  |  |  |  |  |  |
| ETS |  |  |  |  |  |  |  |  |
| Dynamic Regression |  |  |  |  |  |  |  |  |
| RNN |  |  |  |  |  |  |  |  |
| GRU |  |  |  |  |  |  |  |  |
| LSTM |  |  |  |  |  |  |  |  |
| Transformer |  |  |  |  |  |  |  |  |