Predictive Threat Models for Real-Time Decision Support

Jerry Scott Sheehan
Booz Allen Hamilton
El Segundo, California
sheehan_jerry@bah.com

Alec Gray Jr
Booz Allen Hamilton
Arlington, Virginia
grayjr_alec@bah.com

Dejan Neskovic
Booz Allen Hamilton
McLean, Virginia
neskovic_dejan@bah.com

ABSTRACT

In the current global threat environment, homeland security depends on domain and situational awareness. The evolving threat of illegal smuggling and entry along the U.S. borders requires efficient threat classification and resource allocation. The Department of Homeland Security (DHS) Science and Technology (S&T) and DHS Customs and Border Protection (CBP) Air and Marine Operations Center (AMOC) for National Air Domain Security are tasked with monitoring and interdicting threats in the air and maritime domains. Such a complex task requires sorting through large volumes of data to make timely and accurate decisions. This paper describes how leveraging advanced machine learning techniques can support this task.

The proposed approach involves developing predictive threat models (PTM), where multiple machine learning algorithms such as multilayer perceptron (MLP) classification, adaptive boosting, and artificial neural networks (ANN) are tested and evaluated. The top performing models are selected and compared to a hybrid ensemble approach, where the data is split into distinct groups before being used to train each of the models on the same classification and deep learning methods used before. By tailoring models using the hybrid approach and selecting the most applicable for each unique record, the new predictions outperform those of the single-model methodologies. By deploying the hybrid ensemble models using modern machine learning operations (MLOps) best practices such as automated pipelines, continuous integration/continuous deployment (CI/CD), and model performance evaluation, the delivery of these models is streamlined. This ensures our DHS operators can leverage the highest performing real-time predictive analytics to make informed decisions quickly and effectively in the face of evolving threats. Additionally, standalone time-series autoregressive models are continuously trained on live data and instantaneously produce accurate forecasted predictions, equipping our DHS operators with the ability to accurately monitor the future positionings of specific targets of interest on demand.

ABOUT THE AUTHORS

Scott Sheehan is a Senior Lead Data Scientist at Booz Allen Hamilton where he brings innovative data solutions to a variety of government projects. He is the Technical Lead for the PTM contract, assisting the decision-making process at AMOC through innovative, quality, and timely modeling solutions. Mr. Sheehan holds a bachelor’s degree in applied mathematics and statistics from Johns Hopkins University.

Alec “AJ” Gray is an experienced Machine Learning Engineer and Python subject matter expert (SME) at Booz Allen Hamilton who leads the data science efforts on the PTM contract. Applying advanced expertise in machine learning and deep learning, Mr. Gray manages and optimizes predictive models and builds interactive dashboards to continue model improvements. Mr. Gray holds a master’s degree in data analytics engineering from George Mason University.

Dejan Neskovic is a Senior Lead Scientist for Booz Allen Hamilton supporting the Director of Modeling and Simulation Technology Center (MS-TC). He provides strategic engagement advice and guidance to MS-TC on coordination with internal DHS components and other Department of Defense and civilian agencies on projects and technology development opportunities. His current focus is on successful practical application of predictive modeling such as PTM. Mr. Neskovic holds a bachelor’s degree in electrical engineering from the University of Tennessee.

Predictive Threat Models for Real-Time Decision Support

Jerry Scott Sheehan
Booz Allen Hamilton
El Segundo, California
sheehan_jerry@bah.com

Alec Gray Jr
Booz Allen Hamilton
Arlington, Virginia
grayjr_alec@bah.com

Dejan Neskovic
Booz Allen Hamilton
McLean, Virginia
neskovic_dejan@bah.com

INTRODUCTION

Background

Each year, the U.S. spends billions of dollars managing the illegal smuggling of drugs, weapons, and people across our nation's borders. In 2024 alone, the U.S. spent $4.5B for drug-related national security issues associated with border security, immigration enforcement, and countering illicit fentanyl (The White House, 2024). Illegal drugs can enter the U.S by land, maritime, and aviation, making it difficult to allocate the resources comprehensively to prevent drugs from entering the country.

The CBP AMOC Detection Enforcement Officers (DEO) are tasked with diminishing drugs, weapons, and people trafficking through maritime and aviation methods. DEOs continuously monitor air and marine vessels to identify those that pose a risk to the country, and they coordinate the appropriate law enforcement response. This threat classification helps prioritize and allocate resources to each threat effectively while providing law enforcement officials with critical information to intercept those vessels.

This is an incredibly challenging task for DEOs as thousands of vessels enter the U.S each day. DEOs must classify these vessels, identify the high-risks threats, and then determine where the vessel will land as quickly as possible to ensure local law enforcement has enough time to coordinate the interdiction. The accuracy of these often rapid decisions rely on the expertise and experience of the DEOs. However, given the high stress environment, there is a high turnover rate, resulting in a loss of critical subject matter expertise. To help provide DEOs with as much information as possible and assist with this difficult task, the AMOC has begun integrating machine learning models into their decision making for the threat and mission classification of air vessels and their final landing location.

Previous Methods

CBP AMOC DEOs combine domain and technical knowledge to classify aircraft based on the level of risk they pose to the nation and determine where high risk flights will be landing. These machine learning models and their ability to assist DEOs are discussed in detail in the paper by Neskovic et al. (2023). This previous paper describes the "Predictive Threat Models", detailing the methods used in their development, including data collection, Extract, Transform, Load (ETL), feature engineering and selection, model training, validation, and operational deployment. The current paper is a follow-up effort to the research done by Neskovic et al. (2023). This current paper investigates how to improve the previous methods to further assist DEOs by providing better performing models. To do so requires an understanding of the methods and models previously used which are described in the following sub-sections.

Preparing Data for the Model

Hundreds of flight features were captured to be used in the training dataset, but due to some being irrelevant, feature selection was used. Feature selection identifies the most relevant features or variables from a dataset that contributes significantly to the variability of the target variable. The target variable is the variable trying to be predicted. In this case the target variables are:

1. High-risk vs. Low-risk flight.
2. The mission the flight was on (crossing the U.S. border, landing just short and traveling by foot to cross the border, etc.).
3. The final landing location of the flight.

Both Analysis of Variance (ANOVA) and Spearman correlation were used to find the features that related most with the target variable. ANOVA can be used when both the target variable and the features are numeric, to assess the relationship between the target variable and each individual feature. It examines whether the variation in the target variable can be explained by the variability in the numeric features. While ANOVA performs well for linear relationships, Spearman performs well for nonlinear and non-normally distributed data and is therefore a good comparison for our models. To perform this analysis, Spearman uses statistical measures to assess the strength and direction of the monotonic relationship between two variables. It is therefore able to compare the target variable to each feature and determine those most strongly associated with each other. Multiple feature lists were selected for each of the three target variables based on the ANOVA and Spearman correlation tests. Each combination was then tested during the model creation phase and the best performer was used.

To create the machine learning models, the standard technique of splitting training data into a train dataset and a test dataset was used (Kohavi, R., 1995). A unique caveat is that instead of splitting the training data at the observation level, it was split at the group or track level, where each track is a unique flight. This data splitting technique is known as hierarchical data structuring (Farber et al, 2024). By ensuring that all data points from the same track are either in the training set or the test set, data leakage is prevented, and it ensures independence of the training and testing datasets (Farber et al, 2024). This means the model should function the same in the test dataset as it would with new live data. Additionally, the goal is to create a holistic model by providing it with as much data about individual flights' behaviors and characteristics as opposed to only providing it with fragmented flight data.

The following steps are used to process the feature engineered data:

1. All tracks within the training data are grouped together.
2. The tracks are then shuffled at random; randomization is key to improving generalization, reducing overfitting, and avoiding model bias (Abadi, M., 2016).
3. Based on a pre-determined test-size percentage, all observations pertaining to a percentage of the tracks get assigned to a training dataset and a test dataset respectively. For the deep learning model (discussed later), unique tracks were also included in a validation dataset, which is used to monitor the model's accuracy and to prevent overfitting throughout training (Oymak et al., 2021).

Additional enrichment methodology of the training data is devised to further improve performance of the single and course-based landing location models with additional track data from similar aircraft types that fly with comparable kinematic characteristics and origin – destination area parameters. Enriching the training data doubled the number of tracks for classification and deep landing location models and further improve their accuracy.

An important note here is that one classification or deep learning model is used to explain the variance of the entire dataset. Multiple models are created and compared to each other with the best performer being selected and used to provide a solution for a given problem. The top performing model is then used by itself to satisfy the problem, which receives zero influence from any other trained model.

Figure 1 shows a visual of this process by outlining the pipeline used to develop the model along with how the model is then deployed and used to predict on live flights.

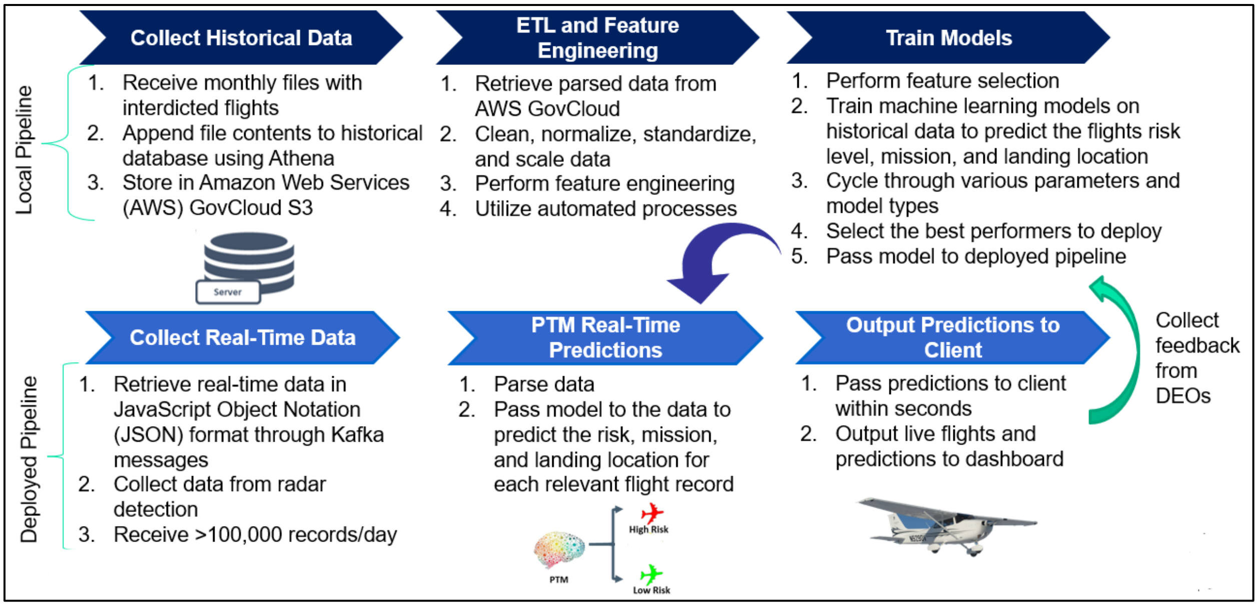

Figure 1: Model Architecture and Data Pipeline

The local pipeline collects the historical data which is then cleaned and transformed before being used in the feature engineering process. The data is then used to train multiple models with the best one for each target variable (risk, mission, landing location) being sent to the deployed pipeline. In the deployed pipeline the selected model for each target variable is used to predict on the live data with predictions being sent to the client within seconds. As feedback is received from enforcement officers, it is used to continue to refine and improve the models to ensure the best ones are being used.

Classification

The classification models leverage a suite of common modern machine learning algorithms: Naïve Bayes, Random Forest, Adaptive Boosting, Logistic Regression, Support Vector Machine, and a Multi-Layer Perceptron neural network (Chilyabanyama, et al. 2022). These models are used to predict the (a) level of risk a flight poses to the nation, (b) type of mission the flight is on, and (c) possible landing area for a particular flight within one degree. Several algorithms were fit to each of these classification target variables. For each problem, the selected algorithm was the one that produced the best model which was then deployed to an operational environment for live on-demand predictions. This is done iteratively each time new historical data is collected. The methodology used prevents reliance on any one algorithm and ensures flexibility and continued learning for the models.

Deep Model

Since predicting an actual coordinate is significantly more difficult than predicting a landing area, a neural network is used, referenced as the "deep model". The deep model, which leverages a subset of artificial intelligence called deep learning (Montezapour et Al, 2023), is designed to predict the actual landing coordinates of a particular flight more decisively. This model is comprised of layers of interconnected neurons (a structure similar to an animal's brain) where data is iteratively processed throughout a series of computations (Montezapour et al, 2023). There are a host of hyperparameter model properties that must work together to produce a generalized, highly explanatory, and qualifiable model.

Findings

The predictive threat models act as force multipliers and have given DEOs an additional decision support capability, making their jobs easier and more effective. These models can confidently classify flights that pose a high risk to the country and provide information on where those high-risk flight will be landing. However, there is room for improvement with the latter. Due to the complexity of the models and the time-consuming nature of neural networks, finding a tailored combination of hyperparameters that can produce accurate results proves challenging (see Neskovic et al. (2023) for a side-by-side results comparison between the modern classification models and deep learning models). The current paper looks at an updated approach by implementing a layered multi-stage ensembled methodology to provide the most accurate models.

METHODS

Multilevel Models

At their base level, the PTMs are trained on radar-captured track observations representing the kinematic characteristics of an aircraft. These characteristics describe the positioning of the aircraft, including latitudinal/longitudinal coordinates, speed, altitude, heading, and much more. A model built at this level would be known as a single-level model and would assume that all observations are independent and identically distributed. This ignores any hierarchical or nested structure in the data such as records from the same flight (Tranmer & Marsh, 2014). Instead, records are grouped by flights resulting in a type of modeling known as a simple multilevel model with one hierarchy (Peugh, 2010).

Multilevel models acknowledge the nested structure of the data, where observations are grouped at multiple levels (Tranmer & Marsh, 2014). To expand upon this, another layer is added to the deep model where flights are grouped by their dominant heading, the direction they predominantly travel across their path from origin to destination. This additional layering removes much of the random effects, a term which refers to the components of a model that capture the variability specific to different levels of the hierarchy within the data (Tranmer & Marsh, 2014). In other words, by splitting one model into several smaller models based on their dominant heading, there is an added potential to explain a higher percentage of variance within each group, collectively, as opposed to explaining data variance with one single model. The hypothesis is that flights heading northwest, for example, are likely going to share certain characteristics that may differ from flights heading south or east. Previously, one model was used to predict the landing location for flights headed in all directions, but this new methodology looks to enhance the performance through a hybrid approach with up to eight models, each predicting flights for a given heading (North, Northwest, Northeast, West, East, South, Southwest, and Southeast).

Another benefit of multilevel modeling is derived from the random effects. By accounting for the heading group-level variations through random effects, the model is less likely to overfit to the noise within each group. This means the model is not just learning the idiosyncrasies of the training data but is also learning the underlying patterns that can apply to new data (Lendave, 2024).

Hybrid-Ensemble Selection Process

Rather than just taking the heading-specific modeling approach, a combined method is used: the “Single-Deep” learning model trained on all data is compared to the heading-specific modeling approach, with the best performer for each heading selected. This method will be referred to as “Hybrid-Ensemble”. This Hybrid-Ensemble includes one to eight different models, depending on model performance. Each heading will compare the performance of the heading specific modeling approach to the performance of the Single-Deep model on that heading.

For example, the process would look like the following:

1. Train a model on all historical flight data (Single-Deep model).
2. Train a model on data for each heading that flights travel. This means taking the same data from step one, splitting it into eight different headings and training on each of the split datasets.
3. De-aggregate the Single-Deep model’s performance by the eight headings that it includes. This involves looking at how well the Single-Deep model performs for each of the given headings.
4. Compare performance of all 16 groups (eight heading models and eight heading groups from the Single-Deep model).
5. Select/package/send for deployment the models that performed the best for each heading. Here anywhere from one to eight models can be sent for deployment, depending on the performance for the eight headings.

This methodology should not only give significant flexibility, but also enable the deep model to recognize and adapt to the hierarchical structure within the data, which reduces the chances of overfitting leading to a more robust solution for the landing location regression problem (Watanabe & Yamana, 2022).

RESULTS

Note: *Due to our results being law enforcement sensitive, the results shown in this section are multiplied by a randomized coefficient, therefore keeping the results comparable to each other while maintaining security of the classified performance.*

Metric Used

As mentioned in the Methods section, the Single-Deep learning model is trained on all historical data available to determine the coordinates (latitude, longitude) of the landing location of a flight. The "Course-Binned" models are trained on data for flights with the respective heading of the eight buckets (North, Northeast, East, Southeast, South, Southwest, West, and Northwest) to determine the landing location coordinates of a flight. For all nine of these models, multiple parameters are tested in order to find the best performance. This includes the train test split percentages which determine the percent of data to train, validate, and test on. 80/15/5 (80% of the data is trained on, 15% is used to validate, and 5% of the data to test on), 90/5/5, and 95/3/2 were all tested. Additionally, the batch sizes of 128 and 256 were compared.

To determine the best model for each of these nine categories, a custom "Within-1" metric is used. The Within-1 metric will determine the percent of flights landing within one degree of the actual landing location of a flight. One degree is about 69 miles. As shown in Figure 2 below, the X marks the actual landing location of the flight. The circle around it represents one degree, or 69 miles away from the actual landing location. If a prediction is inside this circle it is considered "Within-1" and if it falls outside of this circle, then is considered "not Within-1". If there are 10 predictions and five fall within the circle, then Within-1 is 50% for the model. Each model uses this same metric, allowing the models to be compared to each other to determine the best performer.

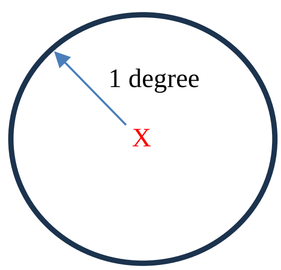

Figure 2: Within-1 Metric

Model Performance

To compare the Single-Deep model to the Course-Binned models, the Single-Deep model performance is broken down based on the performance for each heading category. This means, all predictions for the Single-Deep model are broken out into separate buckets for each heading. For each of these flight headings, the Within-1 metric is calculated and then compared to the performance of the respective Course-Binned model. These model comparisons are shown in Table 1 below.

Course-Binned Model Performance

Table 1: Single-Deep vs Course-Binned Model Performance

Heading	Single-Deep	Course-Binned
East	28.0%	**59.5%**
North	73.5%	73.5%
Northeast	30.3%	**35.0%**
Northwest	43.2%	**86.3%**
South	**65.3%**	25.7%
Southeast	29.2%	**71.2%**
Southwest	59.5%	**67.7%**
West	**51.3%**	39.7%

As shown in Table 1, the Course-Binned models generally outperform the Single-Deep model, but this is not always the case. As a result, it is not always advisable to choose either the Single-Deep model or the Course-Binned models. Instead, a combination of these models should be used to see if this allows for an increase in performance. For this, the Hybrid-Ensemble model methods are used to see how well the model can perform.

Hybrid-Ensemble Model Performance

Because the Single Deep model and the Course-Binned models perform better for different headings, it makes sense to select the top performing model for each heading. This way when a live flight is being tracked, the heading of it can be selected, and the best model will be selected to predict the landing location. Table 2 shows the Within-1 performance of the Single-Deep model for all flights based on the performance for the headings above. The percentage in Table 2 is calculated by multiplying the performance of the Single-Deep model for each heading by the number of flights moving in that direction. The same is done for the Course-Binned models being combined in row two of Table 2. This is then compared to the combination of them, the Hybrid-Ensemble method using the best performing model for each heading by using the following:

- East: Course-Binned
- North: Course-Binned
- Northeast: Course-Binned
- Northwest: Course-Binned
- South: Single-Deep
- Southeast: Course-Binned
- Southwest: Course-Binned
- West: Single-Deep

Table 2: Hybrid-Ensemble Deep Model Performance

Model	Within-1 of Actual
Single-Deep	45.5%
Course-Binned	60.7%
Hybrid-Ensemble	66.5%

Table 2 shows that the Course-Binned model outperforms the Single-Deep model by 15.2%, representing a large increase in the performance for the multilevel model as opposed to the previous model. This result confirms that by splitting the training datasets based on the course heading, the models are better able to predict their particular use case. The best performer though, is the Hybrid-Ensemble model, which is 5.8% more accurate in predicting the landing location for a flight Within-1. This shows the value of the Hybrid-Ensemble method in combining all models to find the best performer for each particular heading. Rather than selecting just the Course-Binned models, a combination of them with the Single-Deep model, for each particular heading, results in the best performer overall.

IMPACT, BENEFITS, AND OTHER APPLICATIONS

The original models were a significant step forward, allowing enforcement offers to condense operational/organizational knowledge/experience. This enabled them to make faster and better data-driven decisions. As shown in the previous section, the new models result in more accurate and precise landing location predictions. As a result, enforcement offers can now coordinate with the proper local law enforcement on where to interdict a flight with a higher rate of success and with more lead time. This results in more efficient use of resources and a greater likelihood of ascertaining criminals by being at the right place at the right time.

The adoption of multilevel data structuring in machine learning marks a significant advancement over traditional single-level models. By organizing data into hierarchical levels, this approach effectively captures the inherent complexities and relationships within the data that simpler models often overlook. This structured approach allows for more nuanced feature extraction and representation, leading to models that can learn and generalize better from the data. The enhanced data representation translates into improved accuracy and robustness in predictions, particularly in complex regression problems where capturing intricate patterns is crucial. The impact of multilevel data structuring is evident not only in aerospace, but also in finance, healthcare, and marketing – areas where predictive performance is a critical factor in decision-making.

Combining multilevel data structuring with ensemble modeling further elevates the performance of machine learning systems. Ensemble models, which integrate predictions from a suite of diverse models, capitalize on the strengths and mitigate the weaknesses of individual models. When these models are built upon a multilevel data foundation, the ensemble benefits from a richer, more comprehensive understanding of the data. This synergy enhances the model's ability to handle noise, outliers, and complex nonlinear relationships, leading to superior performance in regression tasks. For instance, in financial forecasting, combining the best models in an ensemble can lead to more accurate predictions of market trends, thereby aiding in better investment strategies. In the healthcare sector, such enhanced predictive capabilities can improve patient outcomes by enabling more precise diagnostic and treatment plans. Overall, the integration of multilevel data structuring and ensemble modeling not only pushes the boundaries of what machine learning can achieve but also broadens its applicability across various domains.

DISCUSSION/FUTURE DEVELOPMENTS

Deep Model Improvements

With the success of the current Hybrid-Ensemble modeling approach, we are beginning to look at other ways to layer our multilevel approach. Currently, we are grouping observations by track and tracks by heading. Next, we will group headings by region and run through a series of modeling tests to see if this level yields improved results. Here's what we envision.

Multilevel groupings:

- Flight-specific: Simple multi-stage model considering only flights.
- Heading-specific: Adds an additional grouping by headings.
- Region-specific: Further groups by regions, capturing even more variability.

By using region-specific modeling, which includes random effects for flights, headings, and regions, the models should generalize better because:

- Flight Level: Captures variability specific to each flight, helping to model individual flight characteristics.
- Heading Level: Accounts for variability in the direction of flights, which might affect factors like flight duration due to prevailing winds.
- Region Level: Considers regional differences, such as airport congestion or regional weather patterns, which could significantly impact flight timings.

Future Development

While we continue to optimize and improve our deep model, we recognize the importance of exploring other alternative solutions that may yield better results. For this reason, we are developing an online learning methodology, where instead of training regression models on a fixed dataset, we train models tailored to live incoming data, adjusting and updating the model as new data comes in. The approach is an ensemble architecture, which is comprised of the following components:

- Based on kinematic characteristics of a given flight, utilize a conditional rule-based logic that determines that flight's aircraft type: Single Prop, Twin Prop, High Performance Prop, Jet.
- Train and utilize a flight time remaining model to aid landing location airport forecasting.
- Based on kinematic characteristics of the aircraft, determine when it has begun its landing phase.
- Once the flight has begun its initial descent, utilize its aircraft type, current altitude, rate of descent and potentially other data points to predict the flight time remaining.
    - Once the flight time remaining has been determined, train an autoregressive forecasting model to determine the coordinates for where the flight will land.
    - Additionally, determine which airports are near the predicted coordinates where the flight is forecasted to land.

ACKNOWLEDGEMENTS

The research in this document was conducted with the U.S. Department of Homeland Security (DHS) Science and Technology Directorate (S&T) under contract 47QFCA22F0047. Any opinions contained herein are those of the authors and do not necessarily reflect those of DHS S&T.

REFERENCES

Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., & Zhang, L. (2016). Deep Learning with Differential Privacy. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. ACM.

Chilyabanyama, O. N., Chilengi, R., Simuyandi, M., Chisenga, C. C., Chirwa, M., Hamusonde, K., ... & Bosomprah, S. (2022). Performance of Machine Learning Classifiers in Classifying Stunting among Under-Five Children in Zambia. Children, 9(7), 1082.

Färber, I., Schwab, P., Horn, M., Moellers, C., & Karlen, W. (2024). Eliminating Information Leakage in Hard Concept Bottleneck Models with Supervised, Hierarchical Concept Learning. arXiv preprint. https://arxiv.org/abs/2402.05945

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI) (pp. 1137-1143).

Lendave, V. (2024, March 18). A guide to multilevel modeling in machine learning. AIM. https://analyticsindiamag.com/a-guide-to-multilevel-modeling-in-machine-learning/

Mortezapour Shiri, F., Perumal, T., Mustapha, N., & Mohamed, R. (2023). A Comprehensive Overview and Comparative Analysis on Deep Learning Models: CNN, RNN, LSTM, GRU. arXiv:2305.17473.

Neskovic, D., Sheehan, J., & Gray Jr, A. (2023). Real-Time Analytics to Support Operational Decision Making (Predictive Modeling). Interservice/Industry training, Simulation, and Education Conference (I/ITSEC).

Oymak, S., Li, M., & Soltanolkotabi, M. (2021). Generalization Guarantees for Neural Architecture Search with Train-Validation Split. In Proceedings of the 38th International Conference on Machine Learning (Vol. 139, pp. 8291-8301). Proceedings of Machine Learning Research. Retrieved from https://proceedings.mlr.press/v139/oymak21a.html

Peugh, J. L. (2010). A practical guide to multilevel modeling. Journal of school psychology, 48(1), 85-112.

The White House. (2024, March 11). Fact sheet: The President’s budget secures our border, combats fentanyl trafficking, and calls on Congress to enact critical immigration reform. The White House. https://www.whitehouse.gov/briefing-room/statements-releases/2024/03/11/fact-sheet-the-presidents-budget-secures-our-border-combats-fentanyl-trafficking-and-calls-on-congress-to-enact-critical-immigration-reform/

Tranmer, M., & Marsh, C. (2014). Countries and citizens: Linking International Macro and Micro Data. NCRM. https://www.restore.ac.uk/linking_micro_macro_data/

Watanabe, S., Yamana, H. Overfitting measurement of convolutional neural networks using trained network weights. Int J Data Sci Anal 14, 261–278 (2022). https://doi.org/10.1007/s41060-022-00332-1